Computer Vision and Image Understanding 146 (2016) 40–50

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

# Randomized low-rank Dynamic Mode Decomposition for motion detection☆

N. Benjamin Erichson*, Carl Donovan

*School of Mathematics and Statistics, University of St Andrews, UK*

ARTICLE INFO

*Article history:*
Received 25 July 2015
Accepted 6 February 2016
Available online 12 February 2016

*Keywords:*
Dynamic Mode Decomposition
Robust principal component analysis
Randomized singular value decomposition
Motion detection
Background subtraction
Video surveillance

ABSTRACT

This paper introduces a fast algorithm for randomized computation of a low-rank Dynamic Mode Decomposition (DMD) of a matrix. Here we consider this matrix to represent the development of a spatial grid through time e.g. data from a static video source. DMD was originally introduced in the fluid mechanics community, but is also suitable for motion detection in video streams and its use for background subtraction has received little previous investigation. In this study we present a comprehensive evaluation of background subtraction, using the randomized DMD and compare the results with leading robust principal component analysis algorithms. The results are convincing and show the random DMD is an efficient and powerful approach for background modeling, allowing processing of high resolution videos in real-time. Supplementary materials include implementations of the algorithms in *Python*.



## 1. Introduction

The demand for video processing is rapidly increasing, driven by greater numbers of sensors with greater resolution, new types of sensors, new collection methods and an ever wider range of applications. For example, video surveillance, vehicle automation or wild-life monitoring, with data gathered in visual/infra-red spectra or SONAR, from multiple sensors being fixed or vehicle/drone-mounted etc. The overall result is an explosion in the quantity of high dimensional sensor data. Motion detection is often the fundamental building block for more complex video processing and computer vision applications, e.g. object tracking or human behavior analysis. In practice, there are many different types of sensors giving data suitable for object extraction, however we focus here on video data provided by static optical cameras, noting the findings generalize to other data types. In this case, the change in position of an object relative to its surrounding environment can be detected by intensity changes over time in a sequence of video frames. The challenge therefore is to separate intensity changes corresponding to moving objects from those generated by background noise i.e. dynamic and complex backgrounds. From a statistical point of view this can be formulated as a density estimation problem, aiming to find a suitable model describing the background. Moving objects can then be identified by differences from the reconstructed background from the video frames, via some thresholding, as illustrated in Fig. 1. In practice, the problem of finding a suitable model is difficult and often ill-posed due to the many challenges arising in real videos, e.g., dynamic backgrounds, camouflage effects, camera jitter or noisy images, to name only a few. One framework for tackling these challenges is provided by subspace learning techniques. Recently, robust principal component analysis (RPCA) has been very successful in separating video frames into background and foreground components [1]. However, RPCA comes with relatively high computational costs and it is of limited utility for real-time analysis of high resolution video. Hence, in light of increasing sensor resolutions there is a need for algorithms to be more rapid, perhaps by approximating existing techniques.

A competitive alternative is Dynamic Mode Decomposition (DMD) — a data-driven method allowing decomposition of a matrix representing both time and space [2]. Due to the unique properties of videos (equally spaced time with high temporal correlation), DMD is well suited for motion detection, as first demonstrated by Grosek and Kutz [3].

### 1.1. Related work

Bouwmans [4] or Sobral and Vacavant [5] provide recent and comprehensive reviews of methods for background modeling and related challenges. Among the many different techniques, the class of (robust) subspace models are prominent. PCA can be considered a traditional technique for describing the probability distribution of a static background. However, PCA has some essential

☆ This paper has been recommended for acceptance by Leonid Sigal.
* Corresponding author.
*E-mail address:* nbe@st-andrews.ac.uk (N.B. Erichson).

http://dx.doi.org/10.1016/j.cviu.2016.02.005
1077-3142/

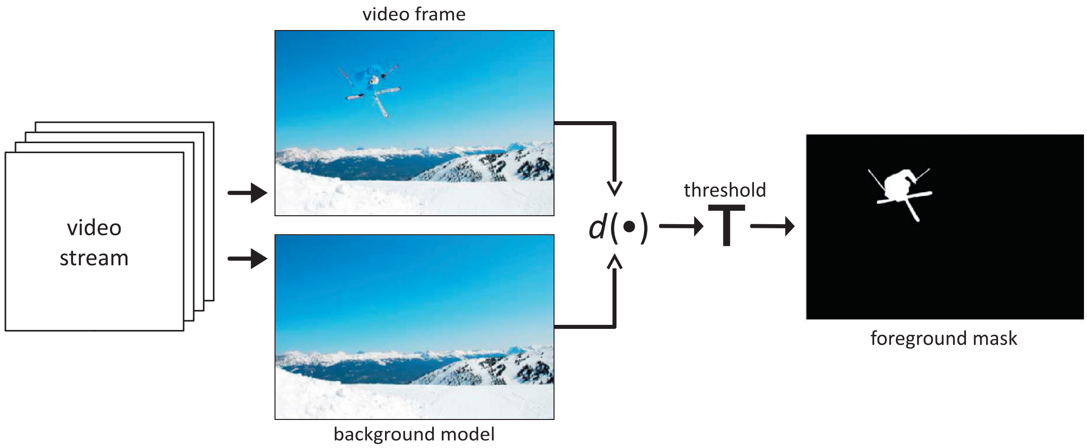


**Fig. 1.** Illustration of background subtraction.

shortcomings and many enhancements have been proposed since the method was first proposed for background subtraction by Oliver et al. [6], e.g. adaptive, incremental or independent PCA. A review of those traditional subspace models and related issues is provided by Bouwman [7]. While DMD is related to PCA and shares some of the same limitations, it can overcome others to greatly improve the performance. Grosek and Kutz [3] have shown that DMD can be seen in fact as an approximation to robust PCA (see also [8]). The idea of RPCA is to separate a matrix $\mathbf{A}$ into a low-rank $\mathbf{L}$ and sparse component $\mathbf{S}$

$$\mathbf{A} = \mathbf{L} + \mathbf{S} \tag{1}$$

This can be formulated as a convex optimization problem that minimizes a combination of the $l_2$ and $l_1$ norm. Applied to video data, the low-rank component describes the relatively static background environment, which is allowed to gradually change over time, while the second component captures the moving objects. This approach has gathered substantive attention for foreground detection since the idea was first introduced by Candès [9] — further extended by Zhou [10] for also capturing entry-wise noise. Bouwmans and Zahzah [1] recently provided a comparative evaluation of the most prominent RPCA implementations, whose results show LSADM [11] and TFOCS [12] algorithms perform best in extracting moving objects in terms of the F-measure. Guyon et al. [13] show in detail how the former algorithm can be used for moving object detection.

The problem formulation via RPCA leads to iterative algorithms with high computational costs. Most of the algorithms require repeated computation of the Singular Value Decomposition (SVD), so clearly the algorithms may be accelerated by using faster approximate SVD, aiming to find only the $k$ dominant singular values. Liu et al. [14] present a Krylov subspace-based algorithm for computing the first $k$ singular values with high precision. They showed that their LMSVD algorithm can reduce the computational time of RPCA substantially. Later they showed even greater computational savings with their Gauss–Newton method based SVD algorithm [15]. If high precision is not the main concern then approximate Monte-Carlo based SVD algorithms can be interesting alternatives [16,17]. A different approach is via randomized matrix algorithms, which are surprisingly robust and provide significant speed-ups, while being simple to implement [18]. Halko et al. [19] and Gu [20] provide comprehensive surveys of randomized algorithms for constructing approximate matrix decompositions, while Mahoney [21] gives a more general overview. One successful approximate robust PCA algorithm using a randomized matrix algorithms is given in GoDec [22].

### 1.2. Motivation and contributions

A core building block of the DMD algorithm, as for RPCA, is the SVD. As noted, traditional deterministic SVD algorithms are expensive to compute and with increasing data they often pose a computational bottleneck. We propose the use of a fast, probabilistic SVD algorithm, exploiting the rapidly decaying singular values of video data. Randomized SVD is a lean and easy to implement technique for computing a robust approximate low-rank SVD [19]. Compared to deterministic truncated or partial SVD algorithms, we gain computational savings in the order of 10 to 30 times. The next effect is to increase speed of about 2 to 3 times with randomized DMD, rather than deterministic SVD based DMD. Hence, randomized DMD may facilitate real-time processing of videos. Moreover, randomized SVD and DMD are embarrassingly parallel and we show that the computational performance can benefit from a Graphics Processing Unit (GPU) implementation. To demonstrate the applicability for motion detection, we have evaluated and compared dynamic mode decomposition on a comprehensive set of synthetic and real videos with other leading algorithms in the field.

The rest of this paper is organized as follows. Section 2 presents randomized SVD as an approximation to the deterministic algorithms. Section 3 first introduces DMD and then shows how a low-rank DMD approximation using randomized SVD can be used for background modeling. Finally a detailed evaluation of DMD is presented in Section 4. Concluding remarks and further research directions are given in Section 5.

## 2. Singular Value Decomposition (SVD)

Matrix factorizations are fundamental tools for many practical applications in signal processing, statistical computing and machine learning. SVD is one such technique, used for data analysis, dimensionality reduction or data compression. Given an arbitrary real matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ we seek a decomposition, such that

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^* \tag{2}$$

where $\mathbf{U} \in \mathbb{R}^{m \times m}$ and $\mathbf{V} \in \mathbb{R}^{n \times n}$ are orthogonal matrices, and $\mathbf{\Sigma} \in \mathbb{R}^{m \times n}$ is a diagonal matrix with the same dimensions as $\mathbf{A}$ [23]. The columns of $\mathbf{U}$ and $\mathbf{V}$ are both orthonormal, called right and left singular vectors, respectively. The singular values denoted as $\sigma_i$ are the diagonal elements of $\mathbf{\Sigma}$ sorted in decreasing order. While we assume a real matrix here, for generality we use the Hermitian transpose denoted as $^*$.

In practice we may be interested in a low-rank approximation of $\mathbf{A}$ with target rank $k \ll m, n$. Choosing the optimal target rank $k$ is highly dependent on the task, i.e. whether one is interested in a