Computer Vision and Image Understanding 199 (2020) 103022

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

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

Dynamic mode decomposition via dictionary learning for foreground modeling in videos

Israr Ul Haq [a,*], Keisuke Fujii [b], Yoshinobu Kawahara [a,c]

[a] Center for Advanced Intelligence Project, RIKEN, 744 Motooka, Fukuoka 819-0395, Japan
[b] Graduate School of Informatics, Nagoya University, Nagoya, Aichi 464-8601, Japan
[c] Institute of Mathematics for Industry, Kyushu University, Fukuoka 819-0395, Japan

ARTICLE INFO

Communicated by Smith John

Keywords:
Dynamic mode decomposition
Nonlinear dynamical system
Dictionary learning
Object extraction
Background modeling
Foreground modeling

ABSTRACT

Accurate extraction of foregrounds in videos is one of the challenging problems in computer vision. In this study, we propose dynamic mode decomposition via dictionary learning (dl-DMD), which is applied to extract moving objects by separating the sequence of video frames into foreground and background information with a dictionary learned using block patches on the video frames. Dynamic mode decomposition (DMD) decomposes spatiotemporal data into spatial modes, each of whose temporal behavior is characterized by a single frequency and growth/decay rate and is applicable to split a video into foregrounds and the background when applying it to a video. And, in dl-DMD, DMD is applied on coefficient matrices estimated over a learned dictionary, which enables accurate estimation of dynamical information in videos. Due to this scheme, dl-DMD can analyze the dynamics of respective regions in a video based on estimated amplitudes and temporal evolution over patches. The results on synthetic data exhibit that dl-DMD outperforms the standard DMD and compressed DMD (cDMD) based methods. Also, the results of an empirical performance evaluation in the case of foreground extraction from videos using publicly available dataset demonstrates the effectiveness of the proposed dl-DMD algorithm and achieves a performance that is comparable to that of the state-of-the-art techniques in foreground extraction tasks.

1. Introduction

The problem of extracting dynamics to detect moving objects in videos is a fundamental task in computer vision. The basic approach to achieve this task is to separate the video stream into foreground and background information which is still considered to be a challenging task in practice because the true background is often difficult to estimate. In real-life scenarios, several challenges, such as gradual or sudden changes in illumination, dynamic background, camera-jitter, sleeping foreground objects, and so on, can be encountered. To address these difficulties, various methods have been proposed over the last decade. For detailed overview of some of the traditional and state-of-the-art methods, we recommend (Bouwmans et al., 2017; Sobral and Vacavant, 2014). One of the most extensively used frameworks to separate a video into foreground and background information is decomposing the video frames into a low-rank matrix (background) and a sparse-matrix (foreground) by principal component analysis (PCA) (Oliver et al., 1999). Variants of this method, such as robust principal component analysis (RPCA), are further discussed by Candès et al. (2011). The decomposition of a matrix into low-rank and sparse matrices can be alternatively solved by dynamic mode decomposition (DMD), which accurately separates a matrix into the stationary background and foreground motions by differentiating between the near-zero frequency modes and the remaining modes away from the origin (Kutz et al., 2015). However, there are some limitations in the standard DMD method that often causes inaccurate extraction of dynamics from the video. In standard DMD method, image sequences ordered in time as column vectors are considered as input, such arrangement of image sequences is unable to extract complex dynamics in videos. Also a modified version of standard DMD; compressed DMD have been proposed by Erichson et al. (2016). The compressed DMD achieves almost the same result as the standard DMD method but at low computation cost. Additionally, the standard DMD approach is, in principle, not applicable to analyze the local level dynamic information in spatiotemporal data.

2. Related work

In literature number of methods have been proposed for background and foreground modeling. Some of the methods are PCA based and the basic idea behind these methods is to split the video into

* Corresponding author.
E-mail address: israr.haq@riken.jp (I. Ul Haq).

https://doi.org/10.1016/j.cviu.2020.103022
Received 4 December 2019; Received in revised form 7 May 2020; Accepted 17 June 2020
Available online 23 June 2020

low-rank (background) and a sparse matrix (foreground) (Vaswani et al., 2018; Guo et al., 2014). Ebadi et al. (2016) proposed a dynamic tree structured RPCA based approach via column subset selection and they considered image sequences as a sum of low-rank and dynamic tree-structured sparse matrix and solved the decomposition using approximated RPCA. They also introduced the low-rank background modeling via column subset selection that reduces the order of complexity and decreases the computation time to process large videos. Incremental Principle Component Pursuit (PCP) method has been proposed by Rodriguez and Wohlberg (2016), they proposed a modified PCP and claim that their method achieves the same results as standard batch PCP algorithms with low computational complexity and allows real time processing compared to PCP algorithm. Javed et al. (2017b, 2018) proposed a Structured-Sparse based RPCA method for moving object detection. They modified the traditional approach of decomposing the input image sequences into low-rank matrix which is considered as background and sparse matrix that contains the foreground information and additionally incorporating two regularization terms (spatial and temporal graph sparsity) on sparse components.

In addition to RPCA based methods deep learning based methods have also gained attention and a comparative overview of recent methods has been presented by Bouwmans et al. (2019). Lim and Keles (2018) proposed a supervised deep learning method under a triplet framework in the encoder part to embed an image in multiple scales into the feature space and use a transposed convolutional network in the decoder part to learn a mapping from feature space to image space by utilizing a pre-trained VGG-16 Network. Minematsu et al. (2018) also proposed a deep learning approach, they modified the approach proposed by Braham and Van Droogenbroeck (2016) by observing full resolution feature maps in all the network layers. García González et al. (2019) proposed a model that divides each video frame into patches and then those are fed to a stacked denoising autoencoder, which is responsible for the extraction of significant features from each image patch. After that, a probabilistic model that is composed of a mixture of Gaussian distributions decides whether the given feature vector describes a patch belonging to the background or the foreground. A unified method based on Generative Adversarial Network (GAN) and image in-painting has been proposed by Sultana et al. (2019). It is an unsupervised visual feature learning hybrid GAN based on context prediction and followed by a semantic in-painting network for texture optimization. Zheng et al. (2019) also proposed a GANs based background subtraction algorithm, they used median filtering algorithm for background extraction then they built a background subtraction by using Bayesian GANs to classify background and foreground pixels.

Seth D. Pendergrass and Brunton (2016) developed a parallelized algorithm to compute the dynamic mode decomposition (DMD) on a graphics processing unit using the streaming method of snapshots singular value decomposition. This method avoids the redundant inner-product as the new data becomes available. Tirunagari et al. (2016) proposed a method to obtain a color background from video frames. In their work they concatenated RGB channels vertically and fed to the DMD algorithm to obtain the RGB background. DMD based methods for background modeling are further discussed by Nathan Kutz et al. (2017) and Hirsh et al. (2019).

David et al. (2009) proposed a dictionary learning approach to split sequence of frames into background and foreground. They learned the dictionary using K-mean classifier and estimated the coefficients using matching pursuit algorithm. In their method they initiated the algorithm with estimated background obtained by patch-wise averaging on frames. However, this method fails to estimate complex background if the background is dynamic or foreground objects are static for short interval of time. Zhao et al. (2011) also proposed a dictionary learning based background modeling algorithm. They learned a dictionary so that it approximates the training data with minimum amount of outlying foreground pixels and produces the sparsest representations of the backgrounds. However, it becomes difficult in real world challenging videos if no single complete background frame is present in the video.

In this study, we advocate the use of DMD via dictionary learning (dl-DMD) for accurate extraction of dynamics in videos. To achieve this task, a dictionary is learned using random patches of input image sequences for better approximation of the input signals. Then coefficient matrices are obtained over this learned dictionary, which contains the better representation of underlying dynamics in videos. Therefore, this is expected to result in a sharp extraction of moving objects from the background, which is achieved by separating modes into foreground and background based on their corresponding eigenvalues.

The remainder of this study can be organized as follows. First, we provide an overview of the dynamic mode decomposition in Section 3. Then, in Section 4, we describe a problem formulation and procedure to perform dl-DMD. Further, in Section 5, we briefly explain dictionary learning. Extraction of local level dynamics is discussed in Section 6. Experiments are presented in Section 7 along with performance evaluations on the basis of benchmark datasets. Finally, Section 8 summarizes and concludes the study.

3. Dynamic mode decomposition

DMD spatiotemporally decomposes the sequential data via data-driven realization of the spectral decomposition of the Koopman operator (Koopman, 1931). Spectral analysis of the Koopman operator lifts the analyses of nonlinear dynamical systems to those of linear systems in function spaces. Further, we briefly review the underlying theory.

Consider a (possibly nonlinear) dynamical system:

$$\boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t), \qquad \boldsymbol{x} \in \mathcal{M},$$

where $\boldsymbol{f}: \mathcal{M} \to \mathcal{M}$, $\mathcal{M}$ is the state space, and t is the time index. In this system, the Koopman operator $\mathcal{K}$ for $\forall \boldsymbol{x} \in \mathcal{M}$ can be defined as follows:

$$\mathcal{K}g(\boldsymbol{x}) = g(\boldsymbol{f}(\boldsymbol{x})),$$

where $g: \mathcal{M} \to \mathbb{C} (\in \mathcal{F})$ denotes an observable in function space $\mathcal{F}$. By definition, $\mathcal{K}$ is a linear operator in $\mathcal{F}$. Assume that there exists a subspace of $\mathcal{F}$ invariant to $\mathcal{K}$, which can be denoted by $\mathcal{G} \subset \mathcal{F}$. Additionally, assume that $\mathcal{G}$ is finite-dimensional and that a set of observables $\{g_1, \dots, g_n\}$ that span over $\mathcal{G}$ are observed to exist. If $\boldsymbol{g} = [g_1, \dots, g_n]^\top : \mathcal{M} \to \mathbb{C}^n$, the one-step evolution of $\boldsymbol{g}$ for $\forall \boldsymbol{x} \in \mathcal{M}$ can be expressed as follows:

$$\boldsymbol{K}\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{g}(\boldsymbol{f}(\boldsymbol{x})),$$

where the finite dimensional $\boldsymbol{K}$ is the restriction of $\mathcal{K}$ to $\mathcal{G}$. An eigenfunction of $\boldsymbol{K}$ can be expressed as $\boldsymbol{\varphi}: \mathcal{M} \to \mathbb{C}^n$, and the corresponding eigenvalue can be expressed as $\lambda \in \mathbb{C}$, i.e., $\boldsymbol{K}\boldsymbol{\varphi}(\boldsymbol{x}) = \lambda\boldsymbol{\varphi}(\boldsymbol{x})$. If all eigenvalues are distinct, any value of $\boldsymbol{g}$ can be expressed as follows:

$$\boldsymbol{g}(\boldsymbol{x}) = \sum_{i=1}^{n} \boldsymbol{\varphi}(\boldsymbol{x})\xi_i$$

with some coefficients ξ_i. Thus, we obtain

$$\boldsymbol{g}(\boldsymbol{x}_t) = \sum_{i=1}^{n} \lambda_i^t \boldsymbol{c}_i, \qquad \boldsymbol{c}_i = \boldsymbol{\varphi}_i(\boldsymbol{x}_0)\xi_i,$$

where $\boldsymbol{g}$ is decomposed into modes $\{\boldsymbol{c}_i\}$, and the modulus and argument of λ_i express the decay rate and frequency of $\boldsymbol{c}_i$, respectively. Differing from classical modal decomposition of linear systems, this decomposition can be applied to nonlinear systems. DMD computes such decomposition using the numerical data. Assume the following data matrices of sizes $\mathbb{C}^{n\times T}$:

$$\boldsymbol{Y}_1 = [\boldsymbol{g}(\boldsymbol{x}_0), \dots, \boldsymbol{g}(\boldsymbol{x}_{T-1})] \quad \text{and} \quad \boldsymbol{Y}_2 = [\boldsymbol{g}(\boldsymbol{x}_1), \dots, \boldsymbol{g}(\boldsymbol{x}_T)]. \tag{1}$$

The two data matrices $\boldsymbol{Y}_1$ and $\boldsymbol{Y}_2$ are connected by a time-independent operator $\boldsymbol{A}$, which approximates the linear evolution and can be written in the matrix notation as follows:

$$\boldsymbol{Y}_2 = \boldsymbol{A}\boldsymbol{Y}_1. \tag{2}$$

Algorithm 1 Dynamic Mode Decomposition Schmid (2010)

Require: $\boldsymbol{Y}_1$ and $\boldsymbol{Y}_2$ defined in Eq. (1)
Ensure: Dynamic modes $\boldsymbol{\Phi}$ and eigenvalues $\boldsymbol{\Delta}$
1: $\boldsymbol{U}_r, \boldsymbol{S}_r, \boldsymbol{V}_r \leftarrow$ compact SVD of $\boldsymbol{Y}_1$.
2: $\tilde{\boldsymbol{A}} \leftarrow \boldsymbol{U}_r^* \boldsymbol{Y}_2 \boldsymbol{V}_r \boldsymbol{S}_r^{-1}$.
3: $\tilde{\boldsymbol{W}}, \boldsymbol{\Delta} \leftarrow$ eigenvectors and eigenvalues of $\tilde{\boldsymbol{A}}$;
4: $\boldsymbol{\Phi} \leftarrow \boldsymbol{Y}_2 \boldsymbol{V}_r \boldsymbol{S}_r^{-1} \tilde{\boldsymbol{W}}_r$
5: **return:** $\boldsymbol{\Phi}, \boldsymbol{\Delta}$;

Algorithm 2 Compressed Dynamic Mode Decomposition Erichson et al. (2016).

Require: flattened video frames $\boldsymbol{Y}_1, \boldsymbol{Y}_2$
1: $\boldsymbol{R} = rand(p_c, m)$ ▷ Generate sensing matrix
2: $\boldsymbol{Y}_c = \boldsymbol{R}^* \boldsymbol{Y}_1$, $\boldsymbol{Y}'_c = \boldsymbol{R}^* \boldsymbol{Y}_2$ ▷ Compress input matrix
3: $\boldsymbol{U}_c, \boldsymbol{S}_c, \boldsymbol{V}_c = \text{svd}(\boldsymbol{Y}_c)$ ▷ SVD
4: $\boldsymbol{A}_c = \boldsymbol{U}_c * \boldsymbol{Y}'_c * \boldsymbol{V}_c * \boldsymbol{S}_c^{-1}$ ▷ Least squares fit
5: $\boldsymbol{W}_c, \boldsymbol{\Delta}_c = \text{eig}(\boldsymbol{A}_c)$ ▷ Eigenvalue decomposition
6: $\boldsymbol{\Phi}_c = \boldsymbol{Y}_2 * \boldsymbol{V}_c * \boldsymbol{S}_c^{-1} * \boldsymbol{W}_c$ ▷ Compute DMD modes
7: $\boldsymbol{b} = \text{lstsq}(\boldsymbol{\Phi}, \boldsymbol{Y}_1)$ ▷ Compute amplitudes by least square method

Algorithm 3 dl-DMD for foreground extraction in videos

Require: video frames $\mathbf{V}$, patch size d, dictionary size k
1: Learn a dictionary $\mathbf{D}$ as in Eq. (5).
2: Calculate the coefficient matrices $\mathbf{B}_1$ and $\mathbf{B}_2$ as in Eqs. (6) and (7), respectively.
3: Perform DMD over coefficient matrices $\mathbf{B}_1$ and $\mathbf{B}_2$ (Section 4.3.3).
4: Threshold zero-frequency modes based on the eigenvalues obtained by Step 3.
5: Reconstruct foregrounds from the approximated coefficient matrix and dictionary as in Eq. (12).

The estimation of matrix $\boldsymbol{A}$ can be found by solving the following least square problem,

$$\tilde{\boldsymbol{A}} = \min_{\boldsymbol{A}} \left\| \boldsymbol{Y}_2 - \boldsymbol{A}\boldsymbol{Y}_1 \right\|^2, \tag{3}$$

and DMD modes that contain the spatial information are the eigenvectors that are obtained through the eigen-decomposition of $\tilde{\boldsymbol{A}}$,

$$\tilde{\boldsymbol{A}}\boldsymbol{W} = \boldsymbol{W}\boldsymbol{\Delta}. \tag{4}$$

Thus, when the dimension of input data is large, it is computationally expensive to analyze matrix $\boldsymbol{A}$. Instead, a rank reduced representation is considered, which is summarized in Algorithm 1 (see Schmid, 2010). In compressed DMD method, data matrices are first compressed by a random sensing matrix and then modes are reconstructed using the original data matrix. The algorithm is further described in Algorithm 2.

4. Proposed method

We propose dl-DMD by extending DMD to employ the dictionary atoms that have been learned using random patches in video frames. The dictionary learning step allows the reconstruction of input video frames using a small subset of dictionary atoms. Then, DMD is performed over the coefficient matrices those are obtained over the dictionary atoms (explained in Section 4.2) which contain the better representation of underlying dynamics of input video and expected to cause accurate foreground/background separation based on the obtained eigenvalues and spatial modes (explained in Section 4.3). The overall procedure of dl-DMD is summarized in Algorithm 3, and the proposed method is further illustrated in Fig. 1. The details of the main steps are described as follows:

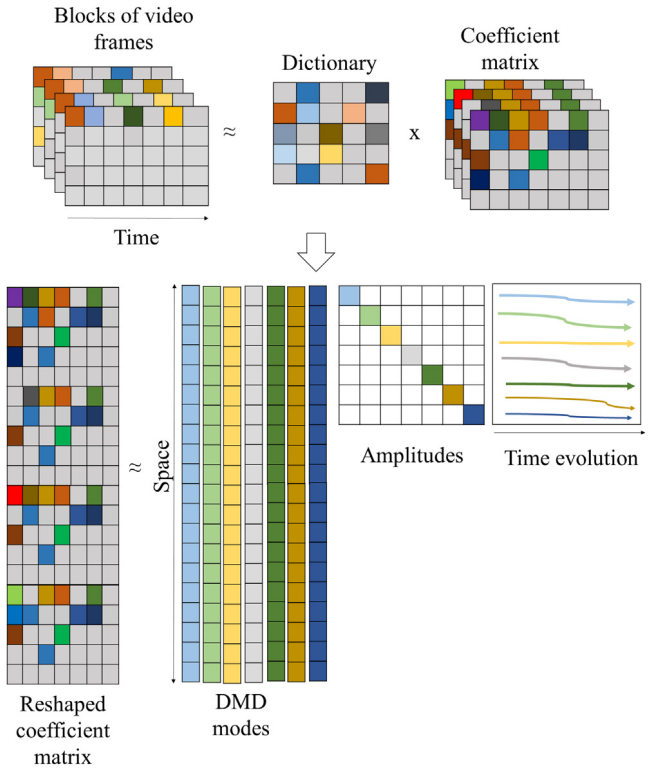

Fig. 1. Illustration of dl-DMD for background/foreground separation in videos.

4.1. Dictionary learning

First, in case of the video frames $\mathbf{V} \in \mathbb{R}^{n_1 \times n_2 \times T}$, each frame $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_T\}$ is converted to a set of overlapping patches, and l patches from the entire set are selected randomly to train a dictionary $\mathbf{D} \in \mathbb{R}^{d \times k}$, where d is the size of a patch and k is the number of elements in the dictionary. The dictionary can be learned by optimizing the coefficient matrix $\mathbf{Z} \in \mathbb{R}^{k \times l}$ and the dictionary in an iterative manner. The dictionary and coefficient matrix are estimated to approximate $\mathbf{X} \in \mathbb{R}^{d \times l}$, which contains the randomly selected patches $\{\mathbf{x}_j\}_{j=1}^{l}$ in the columns. This can be performed by solving the following minimization problem:

$$\min_{\mathbf{D},\mathbf{Z}} \left\{ \|\mathbf{X} - \mathbf{D}\mathbf{Z}\|_F^2 \right\} \quad \text{subject to} \quad \forall_i, \|\boldsymbol{z}_i\|_0 \le T_0, \tag{5}$$

where the coefficient matrix $\mathbf{Z} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_l\}$ contains coefficients that represent each patch of $\mathbf{X}$ and T_0 is the maximum number of non-zero coefficients used to represent those patches.

4.2. Coefficient matrix estimation

The coefficient matrices $\mathbf{B}_1 = \{\tilde{\boldsymbol{\beta}}_{i,1}^1, \tilde{\boldsymbol{\beta}}_{i,2}^1, \ldots, \tilde{\boldsymbol{\beta}}_{i,(T-1)}^1\}_{i=1}^P$ and $\mathbf{B}_2 = \{\tilde{\boldsymbol{\beta}}_{i,1}^2, \tilde{\boldsymbol{\beta}}_{i,2}^2, \ldots, \tilde{\boldsymbol{\beta}}_{i,(T-1)}^2\}_{i=1}^P$ of sizes $\mathbb{R}^{K \times (T-1)}$ are learned over the trained dictionary to approximate the patches of image sequences $\mathbf{Q}_1 = \{\boldsymbol{q}_{i,1}, \boldsymbol{q}_{i,2}, \ldots, \boldsymbol{q}_{i,(T-1)}\}_{i=1}^P$ and $\mathbf{Q}_2 = \{\boldsymbol{q}_{i,2}, \boldsymbol{q}_{i,3}, \ldots, \boldsymbol{q}_{i,T}\}_{i=1}^P$ of sizes $\mathbb{R}^{N \times (T-1)}$. Here, $\{\cdot\}_{i=1}^P$ is the vectorized column with the total number of overlapping patches, P; further, N and K represent the total number of rows in the aligned frames and coefficient matrices, respectively. The patches along all the aligned frames are represented as $\mathbf{Q} = \{\boldsymbol{q}_{i,j}\}_{i=1}^P \in \mathbb{R}^{N \times T}$ for $j = 1, \ldots, T$, and those approximations can be obtained by solving the following minimization problems:

$$\tilde{\boldsymbol{\beta}}_{i,j}^1 = \arg\min_{\boldsymbol{\beta}_{i,j}^1} \left\| \boldsymbol{q}_{i,j} - \mathbf{D}\boldsymbol{\beta}_{i,j}^1 \right\|^2 + \lambda_1 \left\| \boldsymbol{\beta}_{i,j}^1 \right\|_1$$
$$(i = 1, 2, \ldots, P, \ j = 1, 2, \ldots, T-1), \tag{6}$$

$$\tilde{\beta}^2_{i,j} = \underset{\beta^2_{i,(j-1)}}{\arg\min} \left\| q_{i,j} - \mathbf{D}\beta^2_{i,(j-1)} \right\|^2 + \lambda_2 \left\| \beta^2_{i,(j-1)} \right\|_1$$
$$(i = 1, 2, \ldots, P,\ j = 2, \ldots, T), \tag{7}$$

where λ_1 and λ_2 in Eqs. (6) and (7) denote the regularization parameters to control the sparsity in the coefficient matrices $\mathbf{B}_1$ and $\mathbf{B}_2$, respectively.

4.3. Dynamic mode decomposition

The dynamic modes are computed by applying Algorithm 1 to the coefficient matrices $\mathbf{B}_1$ and $\mathbf{B}_2$. That is, we regard these coefficient matrices as the realizations of basis functions whose linear combination gives an observable in $\mathcal{G}$. This is the same analogy with Extended DMD (Williams et al., 2015).

Now, a set of dynamic modes $\mathbf{\Phi} := \{\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_r\}$ and the corresponding eigenvalues $\mathbf{\Delta} := \{\Lambda_1, \ldots, \Lambda_r\}$ are used to reconstruct these image sequences. Here, r is the number of adopted eigenvectors. These modes represent the slowly varying or rapidly moving objects at time points $t \in \{0, 1, 2 \ldots, T-1\}$ in the video frames with associated continuous-time frequencies and can be expressed as follows:

$$\boldsymbol{\omega}_j = \frac{\log(\Lambda_j)}{\Delta t}. \tag{8}$$

Further, the approximated video frames for low- and high-frequency modes at any time point can be reconstructed as

$$\boldsymbol{B}(t) \approx \sum_{j=1}^{r} \boldsymbol{\phi}_j \exp(\boldsymbol{\omega}_j t)\boldsymbol{\alpha}_j = \mathbf{\Phi}\exp(\mathbf{\Omega} t)\boldsymbol{\alpha}, \tag{9}$$

where $\boldsymbol{\phi}_j$ is a column vector of the ith dynamic mode that contains the spatial structure information and $\boldsymbol{\alpha}_j$ is the initial amplitude of the corresponding DMD mode. The vector of the initial amplitudes $\boldsymbol{\alpha}$ can be obtained by taking the initial video frame at time $t = 0$, which reduces Eq. (9) to $\{\tilde{\beta}^1_{i,1}\}^P_{i=1} = \mathbf{\Phi}\boldsymbol{\alpha}$. Note that the matrix of eigenvectors is not square; thus, the initial amplitudes can be observed using the following pseudoinverse process:

$$\boldsymbol{\alpha} = \mathbf{\Phi}^{\dagger}\{\tilde{\beta}^1_{i,1}\}^P_{i=1}. \tag{10}$$

4.4. Foreground/background separation

The key principle to separate the video frames into foregrounds and the background is the thresholding of low frequency modes based on the corresponding eigenvalues. Generally, the portion that represents the background is constant among the frames and satisfies $|\boldsymbol{\omega}_p| \approx 0$, where $p \in \{1, 2, \ldots, r\}$. Typically, a single mode represents the background, which is located near the origin in the complex space, whereas $|\boldsymbol{\omega}_j|, \forall j \neq p$ are the eigenvalues that represent the foreground structures bounding away from the origin. Therefore, the reconstructed video frames can be separated into the background and foreground structures as follows:

$$\tilde{\mathbf{B}} = \underbrace{\boldsymbol{\phi}_p \exp(\boldsymbol{\omega}_p t)\boldsymbol{\alpha}_p}_{Background} + \underbrace{\sum_{j \neq p} \boldsymbol{\phi}_j \exp(\boldsymbol{\omega}_j t)\boldsymbol{\alpha}_j}_{Foreground}, \tag{11}$$

where $\tilde{\mathbf{B}} = \{\tilde{\boldsymbol{\beta}}_{i,1}, \tilde{\boldsymbol{\beta}}_{i,2}, \ldots, \tilde{\boldsymbol{\beta}}^1_{i,T}\}^P_{i=1}$ is the reconstructed coefficient matrix and $t = \{0, \ldots, T-1\}$ is the time indices up to $(T-1)$ frames. Note that the initial amplitude $\boldsymbol{\alpha}_p = \boldsymbol{\phi}^{\dagger}_p\{\tilde{\beta}^1_{i,1}\}^P_{i=1}$ of the stationary background is constant for all the future time points, whereas $\boldsymbol{\alpha}_j = \boldsymbol{\phi}^{\dagger}_j\{\tilde{\beta}^1_{i,1}\}^P_{i=1}, \forall j \neq p$ are the initial amplitudes of varying foreground structures. However, full flattened approximated image sequences $\tilde{\mathbf{Y}}$ are reconstructed with dictionary by the following equation:

$$\{\tilde{y}_{i,j}\}^{P,T}_{i=1,j=1} = \mathbf{D}\{\tilde{\boldsymbol{\beta}}_{i,j}\}^{P,T}_{i=1,j=1} \tag{12}$$

The basic idea behind splitting foregrounds and the background in a video sequence is to separate the low and high eigenvalues, as illustrated in Fig. 2. Fig. 2 depicts the continuous time eigenvalues and temporal evolution of amplitudes. Subplot (a) shows a set of video frames of a moving boat.[1] It can be observed that the boat is absent during the initial and last frames (left to right), whereas the middle frame exhibits a full moving boat. The representation of these frames into modes that describe dynamics by applying dl-DMD which provides an interesting insight related to the moving objects in the foreground, that can be achieved by decomposing these frames into spatial modes, amplitudes, and temporal evolutions. Subplot (b) exhibits the different eigenvalues that are based on the information present in the frames. The background is usually static in videos, which corresponds to the zero eigenvalue that is located near the origin, whereas the eigenvalues that are located away from the origin confirm the presence of other dynamics. Further, subplot(c) depicts the amplitude evolution and dictates that the zero-frequency mode which is constant over time, is the background, and that the remaining modes, which correspond to different frequencies, depict the foreground structures. Additionally, we note that the amplitude that describes the moving boat is negative in the initial frames and begins to increase, eventually reaching its maximum at a frame index of 40 when the boat is almost at the center of the video, capturing majority of the foreground information. The amplitude begins to decrease when the boat moves away from the center. The remaining amplitudes with different frequencies describe the other dynamics of the moving objects in the video.

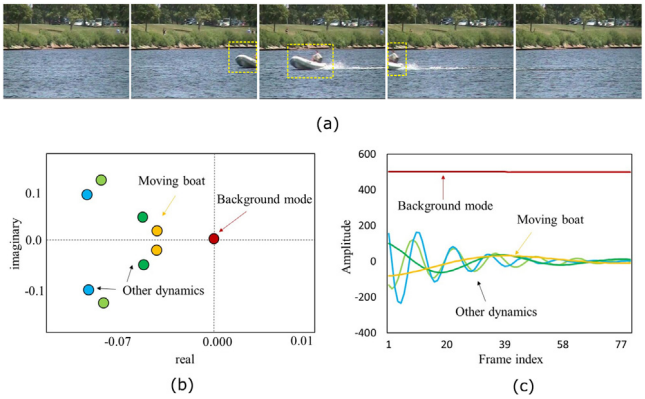

Fig. 2. Splitting foreground and the background (Changedetection.net Goyette et al., 2012 video sequence "boats") . (a) five samples of a moving boat. (b) the near zero eigenvalue corresponds to the estimated background (other eigenvalues correspond to the estimated foreground moving objects). (c) temporal evolutions of amplitudes: the constant amplitude corresponds to the background, while rest of the amplitudes correspond to the foreground moving objects.

5. Dictionary learning and signal approximation

Among several methods that have been proposed for dictionary learning, K-SVD (Aharon et al., 2006) and online dictionary learning (ODL) (Mairal et al., 2009) are popular and practical dictionary learning methods. K-SVD estimates the coefficient matrix using orthogonal matching pursuit (OMP), and a dictionary is updated using SVD. A common approach to perform this calculation is the alternative optimization of the dictionary and coefficient matrix, i.e., minimizing one while maintaining the other constant (Lee et al., 2007; Ouzir et al., 2017). Note that K-SVD is one of the most popular algorithms for image denoising (Elad and Aharon, 2006). Here, we learn a noise free version of the dictionary for better approximation of spatiotemporal signals, $\mathbf{Y}_1$ and $\mathbf{Y}_2$. Typically, while learning the dictionary, the number of overlapping patches, P, is set to a large value that is relative to the signal dimension, d. Generally, the number of dictionary atoms

[1] http://changedetection.net/.

Algorithm 4 Dictionary learning on random input block patches of video

Require: input $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_l\}$, patch size d, dictionary size k

Initialization: Set $\mathbf{D} := \mathbf{D}_0$ ▷ $\mathbf{D}_0$ initialized with random patches.
for $n = 1, \ldots, L$ **do** ▷ Total number of iterations
  $\text{argmin}_{\boldsymbol{b}_i} \|\mathbf{x_i} - \mathbf{D}\boldsymbol{z}_i\|_F^2$ subject to $\forall_i, \|\boldsymbol{z}_i\|_0 \leq T_0$,
  for $j = 1, \ldots, k$ **do**
    $D_j := 0$ ▷ jth Dictionary atom to be updated
    I:=indices of signals in $\mathbf{X}$ whose representations use d
    $\boldsymbol{E} := \boldsymbol{X}_I - \sum_{l \neq j} D_l \boldsymbol{Z}_{l,I}$ ▷ Error matrix without jth atom.
    $\min_D \|\boldsymbol{E} - D_j \boldsymbol{g}_j^{\mathcal{T}}\|_F^2$ ▷ $\boldsymbol{g}_j^{\mathcal{T}} = \boldsymbol{Z}_{j,I}$ is the jth row of nonzeros in $\boldsymbol{Z}_I$.
    $D_j = \boldsymbol{u}_1, \boldsymbol{g}_j = \sigma_1 \boldsymbol{v}_1$ ▷ Update the dictionary atom via SVD of $\boldsymbol{E}$.
  end for
end for

is considerably less than the input signals, i.e., $k \ll P$. Thus, each input signal only uses a few atoms in the dictionary. Note that we only consider an over-complete dictionary with $k > d$, so that dictionary can hold enough information for accurate signal approximation. The dictionary learning procedure is further summarized in Algorithm 4.

Learning strategy: There are two strategies to learn a dictionary; it can either be fixed, i.e., trained offline using a large dataset, or it can be used online in an adaptive manner based on the current estimation. This study formulates the approximation of spatiotemporal signals using an offline dictionary. The training process is performed only once, which exhibits an advantage that is computationally less expensive.

Training data for dictionary learning: To construct training data for dictionary learning, random patches are selected from the randomly selected frames. In some cases of real-life videos, majority of the frames contain background information when there is no moving object or present for a short duration. In such videos, we only choose a set of frames where both foreground and background information is present to approximate the input signals accurately. The training data contains block patches of size 8×8 pixels, and the total number of these patches were set to 30% to 40% of total input patches to train a dictionary. The size of the learned dictionary is set to 64×128 for all videos, where 128 is the total number of atoms in the dictionary. The training set to train a dictionary is depicted in Fig. 3(a), and a learned dictionary on the changedetection.net dataset (Wang et al., 2014), is shown in Fig. 3(b). A dictionary with more number of dictionary atoms minimize the reconstruction error after applying the DMD at the cost of high computation time, whereas a dictionary with few atoms holds less information which in result increases the reconstruction error. Fig. 4 shows the decrease in reconstruction error by increasing the size of dictionary atoms. Reconstruction error is calculated using the original image sequences $\mathbf{V}$ and approximated image sequences $\tilde{\mathbf{Y}}$ (in Eq. (12), reshaped to input video size) as

$$\text{Reconstruction Error} = \sqrt{\frac{1}{E}\sum_{i=1}^{E} |\mathbf{V}(i) - \tilde{\mathbf{Y}}(i)|}, \tag{13}$$

where E is the total number of pixels in the input image sequences. Here we also introduce a correction matrix $\mathbf{C}$ to minimize the reconstruction error generated after applying the DMD that is obtained by solving the following minimization

$$\min_{\mathbf{C}} \left\|\mathbf{V} - \mathbf{C}\tilde{\mathbf{Y}}\right\|_F^2 + \lambda_c \left\|\tilde{\mathbf{Y}}\right\|_1, \tag{14}$$

where λ_c is the regularization parameter. Note that correction matrix $\mathbf{C}$ is applied after full reconstruction of input video sequences in Eq. (12).

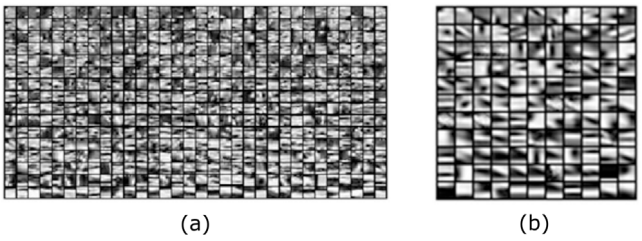

Fig. 3. Collection of 512 random blocks of size 8×8 for training; (b) Dictionary trained on Changedetection.net (Goyette et al., 2012) video sequence "boats".

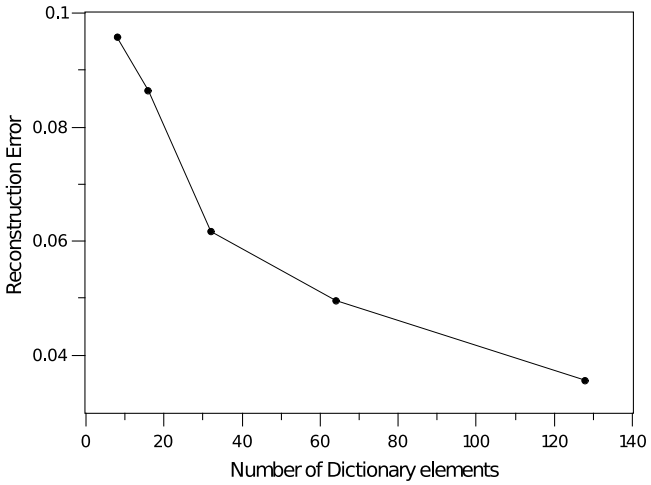

Fig. 4. Reconstruction error decreases with increasing dictionary atoms.

Coefficient matrix approximation: The approximation of spatiotemporal signals, $\mathbf{Y}_1$ and $\mathbf{Y}_2$, depends on the estimation of coefficient matrices $\mathbf{B}_1$ and $\mathbf{B}_2$, which can be estimated using any pursuit algorithm (Pati et al., 1993; Tropp and Gilbert, 2007). To achieve this efficiently, we applied the ℓ_1-norm regularization and solved Eqs. (6) and (7) using fast iterative shrinkage-thresholding algorithm (FISTA) (Beck and Teboulle, 2009). The numbers of non-zero coefficients in the sparse matrices are controlled by the regularization parameters, λ_1 and λ_2. The selection of these parameters is crucial to estimate the signals. In dl-DMD method, we manually set these parameters for each video, so that we can obtain a well approximated signal. Higher values of λ_1 and λ_2 results in a sparse coefficient matrix which helps to denoise the image sequences by just selecting few dictionary atoms. However, for better approximation of input image sequences small values of regularization parameters are recommended.

6. dl-DMD for local level dynamics

In standard DMD method, the foreground and background structures are separated by applying DMD on a set of frames without considering the local level information. This local level approach provides an insight about the dynamics of moving objects at different areas in the video, which can be achieved by considering the overlapping patches in the frames and is illustrated in Fig. 5. Fig. 5(a) (first row) shows the five samples of a slowly moving boat, whereas the corresponding foreground detected frames are shown in Fig. 5(a) (second row). Two 8×8 block patches are considered to analyze the static and varying portions in the video frames that are depicted in blue and red, respectively. The blue patch incorporates majority of the background information, which is constant over time, whereas the red patch contains the foreground information of the moving waves

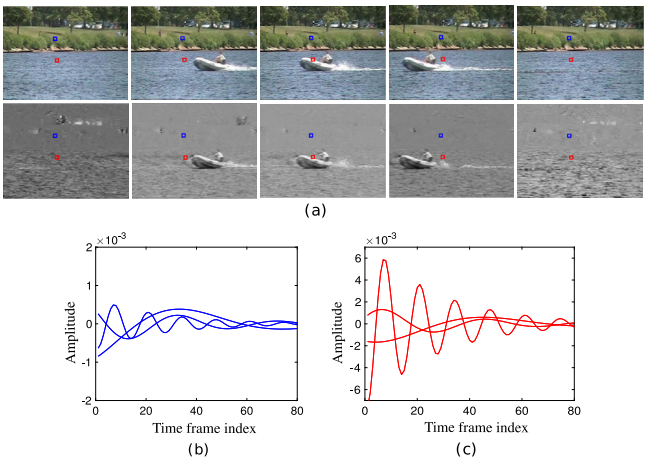

Fig. 5. Local level dynamics; (a) Original frames of video (first row) and detected foregrounds (second row); (b) amplitude evolution of blue patch that corresponds to background; (c) amplitude evolution of red patch that corresponds to foreground.

in the initial and final frames; further, a moving boat is observed in the middle frames. The corresponding graphs of these patches, which depict the amplitudes over time, are depicted in Fig. 5(b) and (c). First, the dynamics that correspond to the blue patch exhibit relatively small amplitudes in comparison to those exhibited by the other patch, indicating that it contains low frequency modes. Further, the red patch exhibits high amplitudes over time, which reveals the presence of high frequency modes due to the presence of moving waves and a boat. The initial amplitudes ($\boldsymbol{\alpha}_{patch}$) of both foregrounds and background for jth patch ($\boldsymbol{x}_{patch} \in \mathbb{R}^{64\times 1}$) in a given set of frames can be expressed by using the following equation as:

$$\{\boldsymbol{y}_{i,1}\}_{i=1}^{P} = \begin{cases} \{\boldsymbol{y}_{i,1}\}_{i=1}^{P} = \boldsymbol{x}_{patch}, & \text{if } i = j. \\ \{\boldsymbol{y}_{i,1}\}_{i=1}^{P} = 0, & \text{otherwise.} \end{cases} \tag{15}$$

$$\boldsymbol{\alpha}_{patch} = \boldsymbol{\Phi}^{\dagger}\{\tilde{\boldsymbol{\beta}}_{i,1}^{1}\}_{i=1}^{P}.$$

7. Experimental results

We empirically investigated the performance of the proposed dl-DMD using synthetic data (Section 7.1) and a real video datasets those are BMC and SBMnet (Section 7.2). Here, the values of the regularization parameters were manually tuned based on the results that were iteratively obtained for a specific dataset. For the synthetic data, we compared our proposed dl-DMD method with standard DMD and compressed DMD because the comparative results of other algorithms can be found in the research of Takeishi et al. (2017).

7.1. Synthetic data

We quantitatively evaluated the performance using the synthetic data that were generated as follows. First, a sequence of noisy images $\{\mathbf{s}_t \in \mathbb{R}^{128\times 128}\}$ was generated using the following equation:

$$\mathbf{s}_t = e_1^t \mathbf{p}_1 + e_2^t \mathbf{p}_2 + \mathcal{N}_t, \tag{16}$$

where $\mathbf{p}_1$, $\mathbf{p}_2 \in \mathbb{R}^{128\times 128}$ and $\mathcal{N}_t$ is the zero-mean Gaussian noise with standard deviation $\sigma = \{0.3\}$ for $t = 0, 1, \ldots, 15$. The dynamic modes of the noise-free image sequences are $\mathbf{p}_1$ and $\mathbf{p}_2$, where $e_1 = 0.99$ and $e2 = 0.9$, are the corresponding eigenvalues, respectively. The noisy sequence of images was applied to the standard DMD, compressed DMD and the proposed dl-DMD. In dl-DMD, a dictionary was initially trained using random patches of $\mathbf{s_t}$. Then, dmd is applied over coefficient matrices obtained over this learned dictionary (using Eqs. (6) and (7)). Further, these block patches were estimated over the atoms of the learned dictionary and finally eigenvalues are obtained (explained in Algorithm 1). The comparison of these results demonstrates that the proposed dl-DMD method can approximate the underlying dynamics more accurately by estimating the true eigenvalues (e_1, e_2) even in the presence of noise. Table 1 shows the estimated eigenvalues by dl-DMD are more close to the ground truth than standard and cDMD methods.

Table 1
Estimated and the ground-truth eigenvalues.

	e_1	e_2
Ground truth	0.99	0.9
Standard DMD	0.994	0.8319
Compressed DMD	0.994	0.8348
dl-DMD	**0.991**	**0.90**

Fig. 6. First-row: original video frames of moving people; second-row: extracted foregrounds with standard DMD method; Third-row: extracted foregrounds with compressed DMD; Last-row: extracted foregrounds with dl-DMD (proposed).

To demonstrate the effectiveness of the proposed method visually, another experiment is performed on a video of SBMnet[1] dataset, where people are strolling in a terrace with no original background provided in the dataset. To visualize the foreground structures extracted by the dl-DMD, standard and cDMD methods we chose 200 consecutive frames from the video and then applied all those three methods. Fig. 6 (first row) shows every 20th frame of first 100 frames of a video. Second row shows the foregrounds extracted by standard DMD method. Third row shows the foregrounds extracted by compressed DMD. Foregrounds in the last row is extracted by the proposed method. It can be visualized that standard and compressed DMD methods were unable to extract the foreground structures more accurately than dl-DMD method. Also with the introduction of dictionary in DMD, the replicating effect of moving objects introduced after applying the DMD can be minimized effectively. Note that, for this experiment size of sensing matrix in cDMD was set to $p_c = (n1 * n2)/2$ (see Algorithm 2), since too much compression will result in loss of spatial information.

7.2. Real video dataset

We further measured the quantitative performance of our proposed method on the publicly available BMC dataset (Vacavant et al., 2012; Sobral and Vacavant, 2014) and SBMnet dataset and also demonstrated that dl-DMD can separate the background structures from the foreground by extracting accurate underlying dynamics in video sequences. This dataset is a benchmark for background modeling of various outdoor surveillance scenarios, such as raining or snowing at different time intervals, illumination changes or snowing at different time intervals, illumination changes relative to outdoor lighting conditions, long duration of motionless foreground objects, and dynamic backgrounds (e.g., moving clouds or trees). The first frame of each of the nine real videos from the BMC dataset are shown in Fig. 7.

Pre-processing of videos: To efficiently process a video stream, we trimmed the background frames that did not change over time. For all

Fig. 7. BMC dataset: First frame of nine real videos.

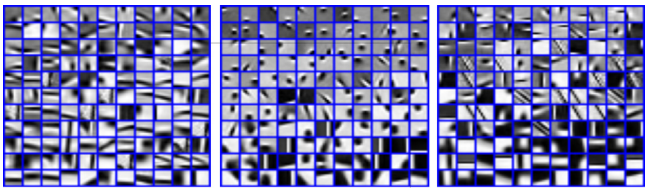

Fig. 8. Trained dictionaries on BMC dataset of first three videos; (001) Boring parking, (002) Big trucks and (003) Wandering students.

the videos, a set of 200 consecutive frames containing the foreground and background information was extracted and down-sampled by a factor of 2 to further decrease the computation time. An identical setting is employed on all BMC real videos to train the respective dictionaries. Since the algorithm is applied on a set of consecutive frames, any change appear after these frames is difficult to predict so more than one background is estimated for those videos where background changes with time as in videos (001), (005) and (008).

Evaluation settings: To achieve better performance for foreground and background separation, we ran a number of tests to tune the parameters for dictionary learning and signal estimation because these parameters exhibit a crucial effect on the overall performance of dl-DMD. A dictionary of size 64×128 pixels was selected to estimate the coefficient matrices and to approximate the signal patches. To train a dictionary using K-SVD algorithm, the total number of non-zero coefficients were set to $T_0 = 16$, $\lambda_1, \lambda_2 = 10^{-3}$, patch size 8×8 with overlapping factor 1. Also, this setting of parameters ensured a better convergence of Eqs. (6) and (7).

dl-dMD performance on BMC dataset. The trained dictionaries on the BMC dataset for videos (001), (002) and (003) are shown in Fig. 8. The reconstructed backgrounds for videos (001) to (009) are shown in Fig. 9, which are modeled on a fixed set of frames. Some of the foreground extraction results of BMC videos (002), (003), (005) and (009) are shown in Fig. 10. Fig. 10 (first row) shows the gray-scale frames of the moving objects that are highlighted in red, and their corresponding backgrounds are shown in (the second row). The difference image was calculated by considering the absolute difference between the original

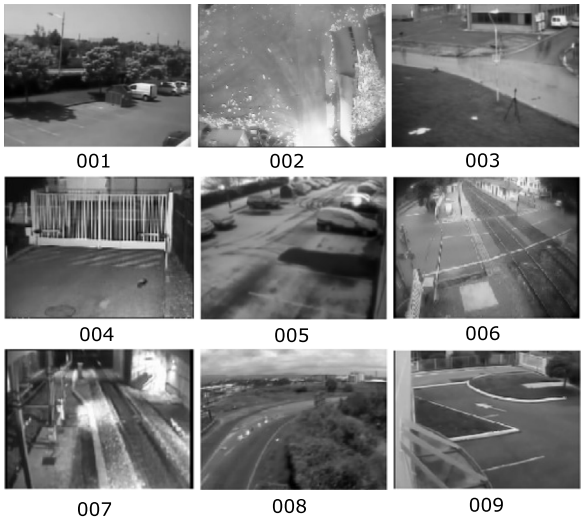

Fig. 9. BMC dataset: Reconstructed backgrounds using dl-DMD of nine videos (001) to (009).

frames and their respective backgrounds. To further enhance the accuracy of the extracted foreground structures, morphological operations were applied to fill the empty holes and to connect the unconnected binary pixels followed by thresholding. The evaluation results for all the nine videos in the BMC dataset are presented in Table 2. The recall, precision, and F-measure metrics were calculated to evaluate the real videos.

Recall: It measures the ability to accurately detect the foreground pixels that belong to the foreground.

Precision: It measures the number of accurately detected foreground pixels that are actually correct.

F-measure: It is the harmonic mean of recall and precision that provides an average value when the values are close, and calculated as

$$F = 2.\frac{Precision \times Recall}{Precision + Recall}. \tag{17}$$

These results indicate some of the strengths and limitations of the proposed method. Note that the proposed method is presented as a batch algorithm applied to a set of consecutive frames. Thus, any changes that occur later in time are difficult to detect, such as the sleeping foreground in video (001), when the cars are parked for a long period of time; this reduces the F-measure value. Another factor that reduces the F-measure value is the presence of non-periodic backgrounds, such as snow and moving clouds, which prominently appear in videos (005) and (008), respectively. However, in case of videos with little variation in the background, high F-measure values were obtained. dl-DMD achieves good F-measure values in videos (002), (003), (004) and (009) because the backgrounds of these videos are almost static for the entire duration. dl-DMD can detect the small and large moving foreground objects, such as a running rabbit in video (004) and the big moving trucks with illumination changes in video (002), respectively; additionally, the competitive F-measure values were obtained.

Dl-DMD performance on SBMnet dataset. To further validate the performance of our proposed method, we quantitatively evaluated it on SBMnet[1] dataset. This dataset contains challenging videos of different categories such as background motion, illumination change, jitter, long and short sequence of images and provides some of the videos with their respective ground-truth backgrounds. We compared the performance of our method with the following state-of-the-art methods (MSCL-Javed et al. (2017a), FSBE-Djerida et al. (2019), LaBGen-Laugraud et al. (2017), NExBI-Mseddi et al. (2019), Photomontage-Agarwala et al. (2004), Bidirectional Analysis-Minematsu et al. (2016),

Table 2
Evaluation results (BMC dataset).

Measure		BMC videos								
		001	002	003	004	005	006	007	008	009
RSL De La Torre et al. (De La Torre and Black, 2003)	Recall	0.800	0.689	0.840	0.872	0.861	0.823	0.658	0.589	0.690
	Precision	0.732	0.808	0.804	0.585	0.598	0.713	0.636	0.526	0.625
	F-Measure	**0.765**	0.744	0.821	0.700	**0.706**	**0.764**	0.647	0.556	0.656
LSADM Goldfarb et al. (Goldfarb et al., 2013)	Recall	0.693	0.535	0.784	0.721	0.643	0.656	0.449	0.621	0.701
	Precision	0.511	0.724	0.802	0.729	0.475	0.655	0.693	0.633	0.809
	F-Measure	0.591	0.618	0.793	0.725	0.549	0.656	0.551	**0.627**	0.752
GoDec Zhou and Tao (Zhou and Tao, 2011)	Recall	0.684	0.552	0.761	0.709	0.621	0.670	0.465	0.598	0.700
	Precision	0.444	0.682	0.808	0.728	0.462	0.636	0.626	0.601	0.747
	F-Measure	0.544	0.611	0.784	0.718	0.533	0.653	0.536	0.600	0.723
Standard DMD	Recall	0.542	0.664	0.762	0.690	0.610	0.683	0.548	0.451	0.551
	Precision	0.571	0.673	0.764	0.762	0.538	0.591	0.651	0.543	0.573
	F-Measure	0.556	0.668	0.763	0.724	0.571	0.633	0.595	0.492	0.561
cDMD (Erichson et al., 2016)	Recall	0.552	0.697	0.778	0.693	0.611	0.700	0.720	0.515	0.566
	Precision	0.581	0.675	0.773	0.770	0.541	0.602	0.823	0.510	0.574
	F-Measure	0.566	0.686	0.776	0.730	0.574	0.647	**0.768**	0.512	0.570
dl-DMD	Recall	0.584	0.732	0.806	0.882	0.493	0.608	0.565	0.456	0.713
	Precision	0.587	0.784	0.931	0.624	0.591	0.605	0.660	0.552	0.811
	F-Measure	0.586	**0.757**	**0.864**	**0.731**	0.537	0.607	0.608	0.500	**0.758**

Table 3
Evaluation results (SBMnet dataset).

	RMR	FC-FlowNet	BE-AAPSA	Bidire-ctional	Photo-montage	NExBI	LaBGen	FSBE	MSCL	dl-DMD
Basic (511)	5.3709	3.9735	4.0511	4.5214	5.79770	5.8916	4.8294	3.7414	4.2186	3.9203
	0.9457	0.9735	0.9744	0.9705	0.9488	0.9345	0.9475	0.9761	0.9703	0.9740
	26.3268	30.8573	30.0319	28.8396	26.6706	26.2599	27.6577	30.5804	30.080	30.4718
	28.3708	32.5541	31.8292	30.7336	28.7131	28.3762	29.5002	32.2388	31.878	32.2673
Basic (Blurred)	2.9910	2.6962	15.2057	2.4346	2.0214	2.5863	1.3990	3.1953	1.8057	7.9128
	0.9699	0.9902	0.8924	0.9924	0.9941	0.9909	0.9975	0.9882	0.9930	0.9685
	30.4749	36.3751	22.4556	37.4609	38.2473	36.3266	41.5779	31.8882	38.1747	27.0219
	31.0951	36.8199	23.3364	37.7694	38.5613	36.6845	41.6541	32.4592	38.5264	27.7682
Clutter (board)	7.0139	14.1523	25.4532	8.6680	13.4739	6.7738	8.0208	5.5795	6.0836	23.5365
	0.8337	0.8691	0.7629	0.8957	0.5029	0.9162	0.8491	0.9340	0.9322	0.5629
	28.3130	22.1587	15.6631	22.5686	18.8444	28.1156	27.4114	29.7845	29.2266	18.3706
	29.3061	23.2484	16.9305	23.7998	20.0911	29.0466	28.3713	30.7618	30.0739	19.5182
Clutter (boulevardJam)	4.8947	5.0200	5.1418	7.7770	12.1045	5.0516	8.2239	2.3321	5.0010	8.2029
	0.9282	0.8619	0.9219	0.8585	0.7604	0.8789	0.6851	0.9653	0.9100	0.7408
	29.2511	30.3476	28.9114	24.2706	20.9163	27.6165	22.6515	33.866	28.5787	26.7204
	30.5310	31.4309	30.0986	25.4284	22.1436	28.8454	23.9772	35.011	29.8099	28.0129
Jitter (boulevard)	13.4511	10.6830	10.8262	10.9028	9.7829	9.4182	10.1888	10.1060	5.8660	10.9366
	0.8198	0.8956	0.8821	0.8715	0.8995	0.9076	0.8946	0.9003	0.9699	0.8921
	19.5784	22.5246	21.1393	20.2970	21.6868	22.2455	21.4645	22.5280	26.0077	22.1579
	21.0043	24.0208	22.5861	21.8557	23.0513	23.7767	22.9249	23.8107	27.1642	23.5842
IntermittentMotion (AVSS2007)	9.2767	11.6751	20.6172	11.9126	12.0167	12.3242	8.3062	11.590	7.5256	12.2210
	0.9094	0.8726	0.7929	0.8198	0.8400	0.8799	0.9050	0.8830	0.9294	0.8492
	20.3096	20.7442	16.4960	19.6485	19.2860	21.1518	21.4577	20.110	22.3138	21.5860
	21.3404	21.7565	17.5546	20.7738	20.2173	22.0076	22.3158	21.211	23.0990	22.8299
IntermittentMotion (BusStation)	3.1366	4.3513	4.5206	4.3423	6.5309	3.0622	7.0296	4.399	3.4057	6.5157
	0.9631	0.9622	0.9621	0.9651	0.8872	0.9815	0.8889	0.984	0.9821	0.9498
	30.3210	31.1049	30.0286	28.0407	21.8651	35.2212	22.0988	33.107	34.2369	28.3636
	31.4297	31.7573	30.9833	29.0178	22.8979	35.7016	23.0664	33.712	34.8402	29.3982
Jitter (Badminton)	8.4681	5.5368	4.3975	5.1114	4.2924	5.2289	2.2670	6.5668	2.4174	10.2870
	0.7365	0.9367	0.9204	0.8954	0.9237	0.8726	0.9805	0.8636	0.9729	0.9532
	23.8541	29.9097	29.1352	27.2333	29.6868	26.7733	34.6482	27.3765	33.8911	26.5590
	24.7652	30.7442	29.9490	28.1560	30.4911	27.7054	35.2688	28.2185	34.5949	27.4705
Very long (bus stop morning)	5.9804	5.6795	5.7741	7.8382	6.1219	6.9018	5.8279	6.1012	5.7245	5.4876
	0.9793	0.9833	0.9836	0.9708	0.9834	0.9501	0.9849	0.9787	0.9856	0.9817
	28.4728	29.8590	29.48156	26.9699	28.3680	25.4774	29.3030	29.2377	29.1391	29.6083
	29.1260	30.5611	29.9870	27.7568	29.0803	26.0903	29.9402	29.9504	29.8815	30.2605
Very short (CUHK square)	6.7243	8.1204	12.1545	7.5873	4.6470	5.9819	4.4512	5.1623	4.9932	7.8274
	0.9359	0.9190	0.8273	0.9141	0.9672	0.9451	0.9713	0.9672	0.9710	0.9051
	25.3910	26.2069	20.5782	23.9515	29.1463	27.2737	30.2741	28.9319	30.2115	25.3864
	26.0475	26.8343	21.4139	24.6868	29.6655	27.8796	30.7823	29.4720	30.6559	26.2176
Very short (dynamic background)	14.5281	9.1948	9.8202	10.8638	10.4144	11.8629	6.7325	9.0226	7.3760	9.2529
	0.8664	0.9426	0.9318	0.9384	0.9268	0.9083	0.9661	0.9524	0.9637	0.9508
	19.8128	25.8003	24.2536	23.3870	24.5275	22.8422	28.3747	25.5196	27.9375	25.881
	20.6954	26.5256	24.9544	24.1915	25.2235	23.5873	28.9363	26.2648	28.6182	26.5334

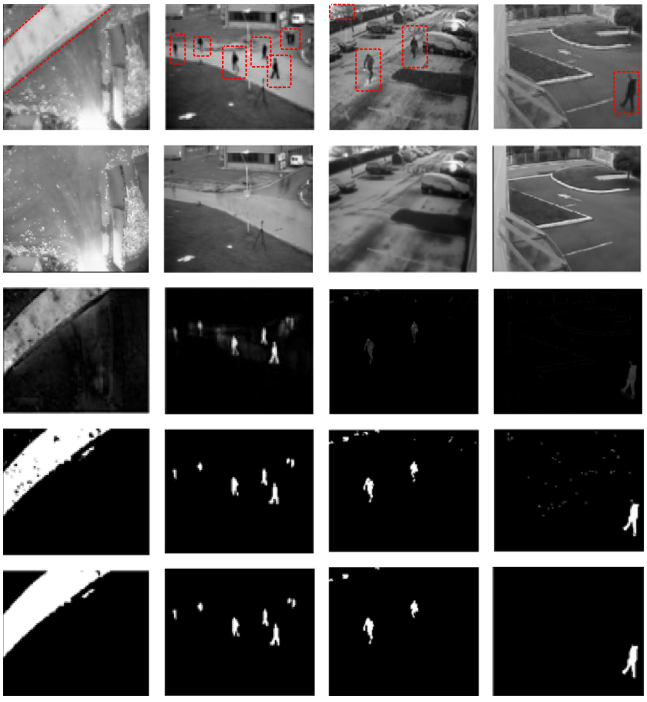

Fig. 10. Foreground extraction corresponding to BMC videos: 002, 003, 005 and 009. The *top row* shows a single frame of each video (moving objects are highlighted in red). The *second row* shows the estimated backgrounds of the respective videos. The *third row* shows the difference between the original frames and backgrounds reconstructed by dl-DMD. The *fourth row* shows the thresholded frames, and the *fifth row* shows the extracted foregrounds after applying morphological operations. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

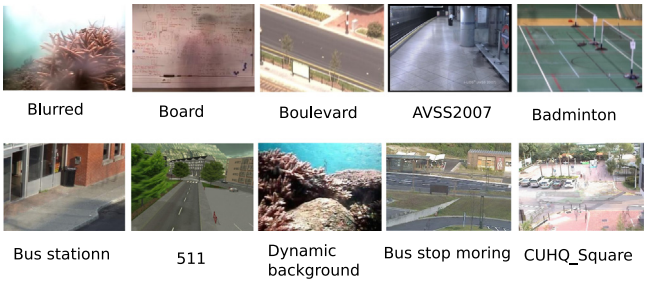

Fig. 11. Reconstructed backgrounds on SBMnet dataset.

BE-AAPSA-Ramirez Alonso et al. (2017), FC-FlowNet-Halfaoui et al. (2016) and RMR-Ortego et al. (2016)) and consider the following four metrics to measure the performance:

AGE: Average Gray-level Error. Average of the gray-level absolute difference between ground truth and the computed background image.

MSSSIM: MultiScale Structural Similarity Index). Estimate of the perceived visual distortion.

PSNR: (Peak-Signal-to-Noise-Ratio) Amounts to $10\log_{10}((L-1)^2/MSE)$ where L is the maximum number of gray levels and MSE is the Mean Squared Error between ground truth and computed background images.

CQM: (Color image Quality Measure). Based on a reversible transformation of the YUV color space and on the PSNR computed in the single YUV bands. It assumes values in db and the higher the CQM value, the better is the background estimate.

Reconstructed backgrounds on SBMnet dataset are shown in Fig. 11 and quantitatively evaluated results are presented in Table 3. We consider four metrics (AGE, MSSIM, PSNR and CQM) to measure the performance which are presented in Table 3 from top to bottom, respectively. Highest, second highest and third highest CMQ values are shown in red, blue and green color, respectively. The proposed method achieved competitive CMQ value for videos (511, AVSS2007, Bus-stop-morning and Dynamic-background) in comparison to other methods. For those videos in which the background exposure is for very short duration of time the shadow effect of moving object appears in the reconstructed background as shown in Fig. 11 (Board) which ultimately reduces the performance. However, for long sequence of images such as in bus-stop-morning; background appears for long duration of time and our method achieved almost the same CMQ value as FC-FlowNet method. Similarly, for Basic (511) and (AVSS2007) our method achieved the second highest CMQ value. However, for short image sequences our method achieved third highest CMQ value because in this category of video. These results show that dl-DMD method is competitive enough among the state-of-the-art algorithms to extract the backgrounds from challenging videos with complex underlying dynamics.

8. Conclusions

We proposed dl-DMD for accurate foreground extraction in videos. In the proposed method, DMD is performed on coefficient matrices estimated over a dictionary which is learned over the randomly selected patches from the video frames. The experiments on synthetic data reveals that dl-DMD method can extract accurate complex dynamics in time series data by extracting true modes and their corresponding eigenvalues. Also, experiments on real video dataset demonstrates that our method can extract foreground and background information in videos with comparable performance to other methods.

CRediT authorship contribution statement

Israr Ul Haq: Software, Validation, Methodology, Investigation, Data curation, Formal analysis, Writing - original draft. **Keisuke Fujii:** Resources, Validation, Data curation. **Yoshinobu Kawahara:** Methodology, Resources, Visualization, Supervision, Project administration, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by JSPS KAKENHI, Japan (Grant Numbers 18H03287) and JST CREST, Japan (Grant Number JPMJCR1913).

Supplementary Materials

Supplementary videos of the proposed method can be downloaded from the following link https://www.dropbox.com/sh/p3artb0lwwlto m1/AADjZ3JbPmwELiZv4Zf_me13a?dl=0. Code is available at https://github.com/Israr-r/dl-DMD-CVIU2020.

References

Agarwala, A., Dontcheva, M., Agrawala, M., Drucker, S., Colburn, A., Curless, B., Salesin, D., Cohen, M., 2004. Interactive digital photomontage. In: ACM SIGGRAPH 2004 Papers. pp. 294–302.

Aharon, M., Elad, M., Bruckstein, A., 2006. *rmk*-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Trans. Signal Process. 54, 4311–4322.

Beck, A., Teboulle, M., 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. 2, 183–202.

Bouwmans, T., Javed, S., Sultana, M., Jung, S.K., 2019. Deep neural network concepts for background subtraction: A systematic review and comparative evaluation. Neural Netw..

Bouwmans, T., Sobral, A., Javed, S., Jung, S.K., Zahzah, E.H., 2017. Decomposition into low-rank plus additive matrices for background/foreground separation: A review for a comparative evaluation with a large-scale dataset. Comp. Sci. Rev. 23, 1–71.

Braham, M., Van Droogenbroeck, M., 2016. Deep background subtraction with scene-specific convolutional neural networks. In: 2016 International Conference on Systems, Signals and Image Processing (IWSSIP). pp. 1–4.

Candès, E.J., Li, X., Ma, Y., Wright, J., 2011. Robust principal component analysis. J. ACM 58 (11).

David, C., Gui, V., Alexa, F., 2009. Foreground/background segmentation with learned dictionary. In: International Conference on Circuits, Systems and Signals, CSS 2009. pp. 197–201.

De La Torre, F., Black, M.J., 2003. A framework for robust subspace learning. Int. J. Comput. Vis. 54, 117–142.

Djerida, A., Zhao, Z., Zhao, J., 2019. Robust background generation based on an effective frames selection method and an efficient background estimation procedure (FSBE). Signal Process., Image Commun. 78, 21–31.

Ebadi, S.E., Ones, V.G., Izquierdo, E., 2016. Dynamic tree-structured sparse RPCA via column subset selection for background modeling and foreground detection. In: 2016 IEEE International Conference on Image Processing (ICIP). IEEE, pp. 3972–3976.

Elad, M., Aharon, M., 2006. Image denoising via sparse and redundant representations over learned dictionaries. IEEE Trans. Image Process. 15, 3736–3745.

Erichson, N.B., Brunton, S.L., Kutz, J.N., 2016. Compressed dynamic mode decomposition for background modeling. J. Real-Time Image Process. 1–14.

García González, J., Ortiz-de Lazcano-Lobato, J.M., Luque-Baena, R.M., Molina-Cabello, M.A., López-Rubio, E., 2019. Foreground detection by probabilistic modeling of the features discovered by stacked denoising autoencoders in noisy video sequences. Pattern Recognit. Lett. 125, 481–487.

Goldfarb, D., Ma, S., Scheinberg, K., 2013. Fast alternating linearization methods for minimizing the sum of two convex functions. Math. Program. 141, 349–382.

Goyette, N., Jodoin, P.M., Porikli, F., Konrad, J., Ishwar, P., 2012. Changedetection.net: A new change detection benchmark dataset. In: CVPRW, 2012 IEEE Computer Society Conference on. IEEE, pp. 1–8.

Guo, H., Qiu, C., Vaswani, N., 2014. An online algorithm for separating sparse and low-dimensional signal sequences from their sum. IEEE Trans. Signal Process. 62, 4284–4297.

Halfaoui, I., Bouzaraa, F., Urfalioglu, O., 2016. CNN-based initial background estimation. In: 2016 23rd International Conference on Pattern Recognition (ICPR). IEEE, pp. 101–106.

Hirsh, S.M., Harris, K.D., Kutz, J.N., Brunton, B.W., 2019. Centering data improves the dynamic mode decomposition. arXiv preprint arXiv:1906.05973.

Javed, S., Mahmood, A., Al-Maadeed, S., Bouwmans, T., Jung, S.K., 2018. Moving object detection in complex scene using spatiotemporal structured-sparse RPCA. IEEE Trans. Image Process. 28, 1007–1022.

Javed, S., Mahmood, A., Bouwmans, T., Jung, S.K., 2017a. Background–foreground modeling based on spatiotemporal sparse subspace clustering. IEEE Trans. Image Process. 26, 5840–5854.

Javed, S., Mahmood, A., Bouwmans, T., Jung, S.K., 2017b. Superpixels-based manifold structured sparse RPCA for moving object detection. In: Proceedings of the British Machine Vision Conference (BMVC 2017). London, UK, pp. 4–7.

Koopman, B., 1931. Hamiltonian systems and transformation in Hilbert space. Proc. Natl. Acad. Sci. USA 17, 315–318.

Kutz, J.N., Fu, X., Brunton, S.L., Erichson, N.B., 2015. Multi-resolution dynamic mode decomposition for foreground/background separation and object tracking. In: Computer Vision Workshop (ICCVW), 2015 IEEE International Conference on. IEEE, pp. 921–929.

Laugraud, B., Piérard, S., Van Droogenbroeck, M., 2017. LaBGen: A method based on motion detection for generating the background of a scene. Pattern Recognit. Lett. 96, 12–21.

Lee, H., Battle, A., Raina, R., Ng, A.Y., 2007. Efficient sparse coding algorithms. In: Advances in Neural Information Processing Systems. pp. 801–808.

Lim, L.A., Keles, H.Y., 2018. Foreground segmentation using convolutional neural networks for multiscale feature encoding. Pattern Recognit. Lett. 112, 256–262.

Mairal, J., Bach, F., Ponce, J., Sapiro, G., 2009. Online dictionary learning for sparse coding. In: Proceedings of the 26th Annual International Conference on Machine Learning. ACM, pp. 689–696.

Minematsu, T., Shimada, A., Taniguchi, R.i., 2016. Background initialization based on bidirectional analysis and consensus voting. In: 2016 23rd International Conference on Pattern Recognition (ICPR). IEEE, pp. 126–131.

Minematsu, T., Shimada, A., Uchiyama, H., Taniguchi, R.i., 2018. Analytics of deep neural network-based background subtraction. J. Imaging 4 (78).

Mseddi, W.S., Jmal, M., Attia, R., 2019. Real-time scene background initialization based on spatio-temporal neighborhood exploration. Multimedia Tools Appl. 78, 7289–7319.

Nathan Kutz, J., Benjamin Erichson, N., Askham, T., Pendergrass, S., Brunton, S.L., 2017. Dynamic mode decomposition for background modeling. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 1862–1870.

Oliver, N., Rosario, B., Pentland, A., 1999. A Bayesian computer vision system for modeling human interactions. In: International Conference on Computer Vision Systems. Springer, pp. 255–272.

Ortego, D., SanMiguel, J.C., Martínez, J.M., 2016. Rejection based multipath reconstruction for background estimation in video sequences with stationary objects. Comput. Vis. Image Underst. 147, 23–37.

Ouzir, N., Lairez, O., Basarab, A., Tourneret, J.Y., 2017. Tissue motion estimation using dictionary learning: Application to cardiac amyloidosis. In: Ultrasonics Symposium (IUS), 2017 IEEE International. IEEE, pp. 1–4.

Pati, Y.C., Rezaiifar, R., Krishnaprasad, P.S., 1993. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In: Signals, Systems and Computers, 1993. 1993 Conference Record of the Twenty-Seventh Asilomar Conference on. IEEE, pp. 40–44.

Ramirez Alonso, G., Ramirez-Quintana, J.A., Chacon-Murguia, M.I., 2017. Temporal weighted learning model for background estimation with an automatic re-initialization stage and adaptive parameters update. Pattern Recognit. Lett. 96, 34–44.

Rodriguez, P., Wohlberg, B., 2016. Incremental principal component pursuit for video background modeling. J. Math. Imaging Vision 55, 1–18.

Schmid, P.J., 2010. Dynamic mode decomposition of numerical and experimental data. J. Fluid Mech. 656, 5–28.

Seth D. Pendergrass, J.N., Brunton, S.L., 2016. Streaming GPU singular value and dynamic mode decompositions.

Sobral, A., Vacavant, A., 2014. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. Comput. Vis. Image Underst. 122, 4–21.

Sultana, M., Mahmood, A., Javed, S., Jung, S.K., 2019. Unsupervised deep context prediction for background estimation and foreground segmentation. Mach. Vis. Appl. 30, 375–395.

Takeishi, N., Kawahara, Y., Yairi, T., 2017. Sparse non-negative dynamic mode decomposition. In: 2017 IEEE Int. Conf. on Image Process. (ICIP'17). pp. 2682–2686.

Tirunagari, S., Poh, N., Bober, M., Windridge, D., 2016. Can dmd obtain a scene background in color?. In: 2016 International Conference on Image, Vision and Computing (ICIVC). IEEE, pp. 46–50.

Tropp, J.A., Gilbert, A.C., 2007. Signal recovery from random measurements via orthogonal matching pursuit. IEEE Trans. Inform. Theory 53, 4655–4666.

Vacavant, A., Chateau, T., Wilhelm, A., Lequièvre, L., 2012. A benchmark dataset for outdoor foreground/background extraction. In: Asian Conference on Computer Vision. Springer, pp. 291–300.

Vaswani, N., Bouwmans, T., Javed, S., Narayanamurthy, P., 2018. Robust subspace learning: Robust pca, robust subspace tracking, and robust subspace recovery. IEEE Signal Process. Mag. 35, 32–55.

Wang, Y., Jodoin, P.M., Porikli, F., Konrad, J., Benezeth, Y., Ishwar, P., 2014. Cdnet 2014: An expanded change detection benchmark dataset. In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on. IEEE, pp. 393–400.

Williams, M.O., Kevrekidis, I.G., Rowley, C.W., 2015. A Data–Driven approximation of the Koopman operator: Extending dynamic mode decomposition. J. Nonlinear Sci. 25, 1307–1346.

Zhao, C., Wang, X., Cham, W.K., 2011. Background subtraction via robust dictionary learning. EURASIP J. Image Video Process. 2011, 1–12.

Zheng, W., Wang, K., Wang, F.Y., 2019. A novel background subtraction algorithm based on parallel vision and Bayesian GANs. Neurocomputing 394, 178–200.

Zhou, T., Tao, D., 2011. Godec: Randomized low-rank & sparse matrix decomposition in noisy case. In: International Conference on Machine Learning. Omnipress, pp. 33–40.