Dynamic Mode Decomposition via Dictionary Learning for Foreground Modeling in Videos

Israr Ul Haq[1], Keisuke Fujii[1,2] and Yoshinobu Kawahara[1,3]
[1]*Center for Advanced Intelligence Project, RIKEN, Japan*
[2]*Graduate School of Informatics, Nagoya University, Japan*
[3]*Institute of Mathematics for Industry, Kyushu University, Fukuoka, Japan*
israr.haq@riken.jp, fujii@i.nagoya-u.ac.jp, kawahara@imi.kyushu-u.ac.jp

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

One of the fundamental computer vision objectives is to extract accurate dynamic information from video sequences. The basic application can be the separation of foreground and background information in videos. This is still considered to be a challenging task in practice because the true background is often difficult to estimate. To address this issue, 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 in (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 non-zero frequency modes (Kutz and Fu, 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 (cDMD) have been proposed in (Erichson et al., 2016). The compressed DMD achieves almost the same results as the standard DMD method but at low computation cost.

In this study, we advocate the use of DMD via dictionary learning (dl-DMD) for accurate extraction of dynamics in videos. For this purpose, 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 those contain the better representation of underlying dynamics in videos which results in a sharp extraction of foreground structures from the background than standard DMD and cDMD.

The remainder of this study can be organized as follows. First, we provide an overview of the dynamic mode decomposition in Section 2. Then in Section 3, we describe a problem formulation and procedure to perform dl-DMD. Experiments are presented in Section 4 along with performance evaluations. Finally, Section 5 summarizes and concludes the study.

2 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 analysis 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 $\boldsymbol{f}\colon \mathcal{M} \to \mathcal{M}$, $\mathcal{M}$ 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 $g\colon \mathcal{M} \to \mathbb{C}\,(\in \mathcal{F})$ 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 $\{g_1, \ldots, g_n\}$ that span over $\mathcal{G}$ are observed to exist. If $\boldsymbol{g} = [g_1, \ldots, g_n]^\top \colon \mathcal{M} \to \mathbb{C}^n$, 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 $\boldsymbol{\varphi}\colon \mathcal{M} \to \mathbb{C}^n$, 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:

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

with some coefficients ξ_i. Thus, we obtain

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.

Require: 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: $\boldsymbol{U}$,$\boldsymbol{S}$,$\boldsymbol{V}$ = svd($\boldsymbol{Y}_c$) ▷ SVD
4: $\boldsymbol{A}$ = $\boldsymbol{U}$ * $\boldsymbol{Y}'_c$ *$\boldsymbol{V}$*$\boldsymbol{S}^{-1}$ ▷ Least squares fit
5: $\boldsymbol{W}$,$\boldsymbol{\Delta}$ = eig($\boldsymbol{A}$) ▷ Eigenvalue decomposition
6: $\boldsymbol{\Phi}_c$ = $\boldsymbol{Y}_2$ * $\boldsymbol{V}$ * $\boldsymbol{S}^{-1}$ * $\boldsymbol{W}$ ▷ Compute DMD modes
7: $\boldsymbol{b}$ = lstsq($\boldsymbol{\Phi}$,$\boldsymbol{Y}_1$) ▷ Compute amplitudes by least square method

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

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 $\mathbb{C}^{n\times T}$:

$$\begin{aligned} \boldsymbol{Y}_1 &= [\boldsymbol{g}(\boldsymbol{x}_0), \ldots, \boldsymbol{g}(\boldsymbol{x}_{T-1})], \\ \boldsymbol{Y}_2 &= [\boldsymbol{g}(\boldsymbol{x}_1), \ldots, \boldsymbol{g}(\boldsymbol{x}_T)]. \end{aligned} \tag{1}$$

Then, the most popular variant of the DMD algorithm is described in Algorithm 1. In compressed DMD (cDMD) 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 summarized in Algorithm 2.

3 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 subsection. 3.2) which contain the better representation of underlying dynamics of input video and expected

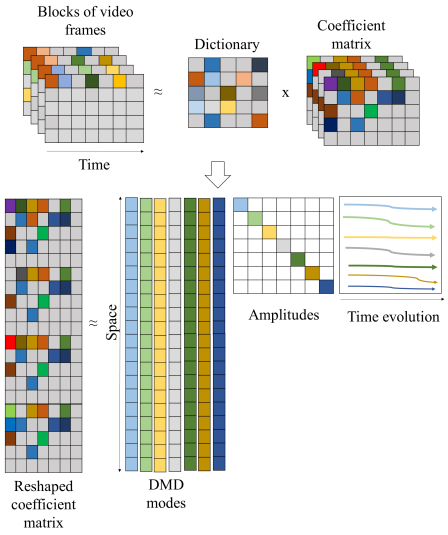

Figure 1: Illustration of dl-DMD for background/foreground separation in videos.

Algorithm 3: dl-DMD for foreground extraction in videos.

Require: Video frames $\mathbf{V}$, patch size d, dictionary atoms k.

1: Learn a dictionary $\mathbf{D}$ as in Eq. (2).
2: Calculate the coefficient matrices $\mathbf{B}_1$ and $\mathbf{B}_2$ as in Eqs. (3) and (4), respectively.
3: Perform DMD over coefficient matrices $\mathbf{B}_1$ and $\mathbf{B}_2$ (subsection 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. (9).

to cause accurate foreground/background separation based on the obtained eigenvalues and spatial modes (explained in subsection 3.3). However, in standard DMD and cDMD methods, DMD is directly applied over spatiotemporal matrices those are built from input frames of a video due to which it becomes difficult to extract dynamics and seperate foreground/background information. The overall procedure of dl-DMD is summarized in Algorithm 3, and the proposed method is further illustrated in Figure 1. The details of the main steps are described as follows:

3.1 Dictionary Learning

First, in case of a given 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 atoms or 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 give a representation 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 (Aharon et al., 2006). This can be performed by solving the following minimization problem:

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

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

3.2 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:

$$\tilde{\boldsymbol{\beta}}_{i,j}^1 = \underset{\boldsymbol{\beta}_{i,j}^1}{\arg\min} \|\boldsymbol{q}_{i,j} - \mathbf{D}\boldsymbol{\beta}_{i,j}^1\|^2 + \lambda_1 \|\boldsymbol{\beta}_{i,j}^1\|_1 \quad (i = 1, 2, \ldots, P,\ j = 1, 2, \ldots, T-1), \tag{3}$$

$$\tilde{\boldsymbol{\beta}}_{i,j}^2 = \underset{\boldsymbol{\beta}_{i,(j-1)}^2}{\arg\min} \|\boldsymbol{q}_{i,j} - \mathbf{D}\boldsymbol{\beta}_{i,(j-1)}^2\|^2 + \lambda_2 \|\boldsymbol{\beta}_{i,(j-1)}^2\|_1 \quad (i = 1, 2, \ldots, P,\ j = 2, \ldots, T), \tag{4}$$

where λ_1 and λ_2 in Eqs. (3) and (4) denote the regularization parameters to control the sparsity in the coefficient matrices $\mathbf{B}_1$ and $\mathbf{B}_2$, respectively.

3.3 Dynamic Mode Decomposition

The dynamic modes are computed by applying Algorithm 1 to the coefficient matrices $\mathbf{B}_1$ and $\mathbf{B}_2$. A set of dynamic modes $\boldsymbol{\Phi} := \{\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_r\}$ and the corresponding eigenvalues $\boldsymbol{\Delta} := \{\Lambda_1, \ldots, \Lambda_r\}$ are obtained, those represent the spatial and frequency information of the video. 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:

$$\boldsymbol{\omega}_j = \frac{\log(\Lambda_j)}{\Delta t}. \tag{5}$$

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

$$\tilde{\mathbf{B}}(t) \approx \sum_{j=1}^{r} \boldsymbol{\phi}_j \exp(\boldsymbol{\omega}_j t)\boldsymbol{\alpha}_j = \boldsymbol{\Phi}\exp(\boldsymbol{\Omega} t)\boldsymbol{\alpha}, \tag{6}$$

where $\boldsymbol{\phi}_j$ is a column vector of the i-th 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. (6) to $\{\tilde{\boldsymbol{\beta}}^1_{i,1}\}_{i=1}^{P} = \boldsymbol{\Phi}\boldsymbol{\alpha}$. Note that the matrix of eigenvectors is not square; thus, the initial amplitudes can be observed using the following pseudoinverse process:

$$\boldsymbol{\alpha} = \boldsymbol{\Phi}^{\dagger}\{\tilde{\boldsymbol{\beta}}^1_{i,1}\}_{i=1}^{P}. \tag{7}$$

3.4 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:

$$\tilde{\mathbf{B}} = \underbrace{\boldsymbol{\phi}_p \exp(\boldsymbol{\omega}_p t)\boldsymbol{\alpha}_p}_{\textit{Background}} + \underbrace{\sum_{j\neq p} \boldsymbol{\phi}_j \exp(\boldsymbol{\omega}_j t)\boldsymbol{\alpha}_j}_{\textit{Foreground}}, \tag{8}$$

where $\tilde{\mathbf{B}} = \{\tilde{\boldsymbol{\beta}}_{i,1}, \tilde{\boldsymbol{\beta}}_{i,2}, \ldots, \tilde{\boldsymbol{\beta}}^1_{i,T}\}_{i=1}^{P}$ 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 $\boldsymbol{\alpha}_p = \boldsymbol{\phi}_p^{\dagger}\{\tilde{\boldsymbol{\beta}}^1_{i,1}\}_{i=1}^{P}$ of the stationary background is constant for all the future time points, whereas $\boldsymbol{\alpha}_j = \boldsymbol{\phi}_j^{\dagger}\{\tilde{\boldsymbol{\beta}}^1_{i,1}\}_{i=1}^{P}, \forall j \neq p$ are the initial amplitudes of varying foreground structures. However, full flattened approximated video sequences are reconstructed with a learned dictionary (in subsection 3.1) by the following equation:

$$\{\tilde{\boldsymbol{q}}_{i,j}\}_{i=1,j=1}^{P,T} = \mathbf{D}\{\tilde{\boldsymbol{\beta}}_{i,j}\}_{i=1,j=1}^{P,T} \tag{9}$$

Foregrounds and background separation in a video is illustrated in Figure 2, that depicts the continuous time eigenvalues and temporal evolution of amplitudes (Erichson et al., 2016). 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, whereas the middle frame exhibits a full moving boat. The representation of these frames into modes that describe dynamics by applying dl-DMD provides an interesting insight related to the moving objects in the foreground, which can be achieved by factorizing 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.

4 EXPERIMENTAL RESULTS

We empirically investigated the performance of the proposed dl-DMD using synthetic data (Section 4.1) and a real video dataset, i.e., BMC (Section 4.2). 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 (Takeishi et al., 2017).

[1] http://changedetection.net/

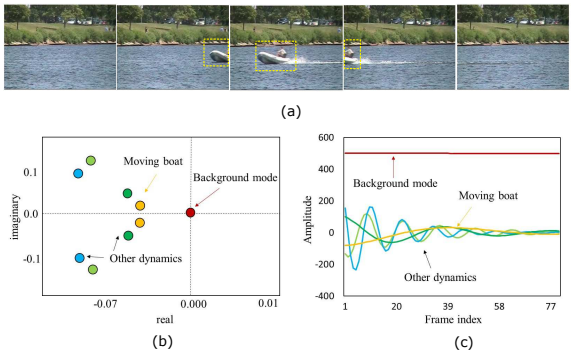

Figure 2: Splitting foreground and the background (Changedetection.net (Goyette et al., 2012) video sequence "boats") . (**a**) five frames of a moving boat. (**b**) the near zero eigenvalue corresponds to the background and rest to other dynamics. (**c**) temporal evolutions of amplitudes.

4.1 Synthetic Data

We quantitatively evaluated the performance using the synthetic data that were generated as follows. First, a sequence of noisy images $\{\mathbf{s}_t \in \mathbb{R}^{128\times128}\}$ was generated using the following equation:

$$\mathbf{s}_t = e_1^t \mathbf{p}_1 + e_2^t \mathbf{p}_2 + \mathcal{N}_t, \tag{10}$$

where $\mathbf{p}_1$, $\mathbf{p}_2 \in \mathbb{R}^{128\times128}$ 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 standard DMD, cDMD and dl-DMD methods were applied on these noisy sequence of images. The comparison of these methods demonstrates that the dl-DMD can approximate the underlying dynamics more accurately by estimating the true eigenvalues (e_1, e_2) even in the presence of noise compared to the standard DMD and cDMD. Table. 1 shows the estimated eignevalues by standard, compressed and dl-DMD method.

To demonstrate the effectiveness of the proposed method visually, another experiment is performed on a video of SBMnet[2] 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. Figure 3 (first row) shows every 20th frame of first 100 frames of a video. Second row shows the foregrounds extracted by standard DMD method. Third and fourth rows show the foregrounds extracted by the compressed DMD and the proposed method, respectively. It can be visualized that dl-DMD can extract the foreground dynamics more accurately than standard and cDMD methods. 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.

[2]http://scenebackgroundmodeling.net/

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 (proposed)	**0.991**	**0.90**

Figure 3: 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).

Parameters Selection: The parameters of dl-DMD were tuned manually for best results and set to $T_0 = 16$, $\lambda_1, \lambda_2 = 10^{-3}$, dictionary size = 64×128, patch size 8×8 with overlapping factor 1. 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 that in result increases the reconstruction error. Figure. 4 shows the decrease in reconstruction error by increasing the size of dictionary atoms. Another important parameter is the patch size, the relation between the patch size and the mean reconstruction error (between the DMD reconstructed output and input video) is shown in Table. 2. This relation shows that for a fixed number of dictionary atoms, increasing the patch size results in increasing the reconstruction error. Figure. 5 shows some of the learned dictionaries on BMC dataset.

4.2 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,

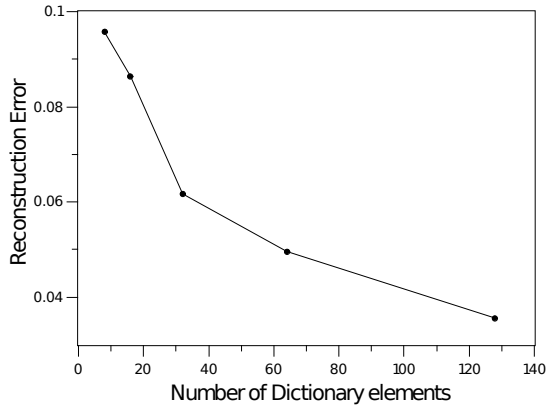

Figure 4: Reconstruction error decreases with increasing dictionary atoms.

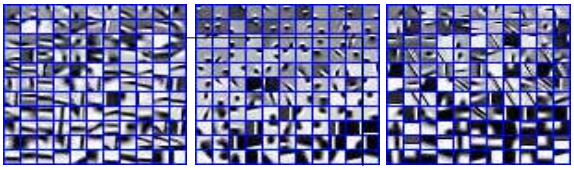

Figure 5: Trained dictionaries on BMC dataset of first three videos; (001) Boring parking, (002) Big trucks and (003) Wandering students.

2014). 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).

Some of the foreground extraction results of BMC videos (002), (003), (005) and (009) are shown in Figure 6 and evaluation results for all the nine videos on this dataset are presented in Table. 3 (Erichson et al., 2016). For pre-processing we cropped 200 consecutive frames of each video, and more than one background is estimated for those videos where background changes with time as in videos (001), (005) and (008).

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

Table 2: Relation b/w patch size and reconstruction error.

Patch size	Reconstruction error
4×4	0.0326
8×8	0.0369
12×12	0.0373
16×16	0.0411

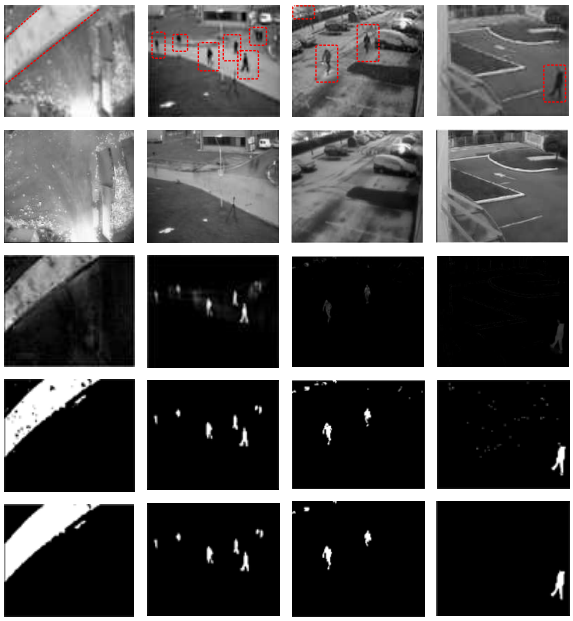

Figure 6: Foreground extraction corresponding to BMC videos: 002, 003, 005 and 009. The *top row* shows a single frame of each video. The *second row* shows the estimated backgrounds. The *third row* shows the difference between the original frames and backgrounds reconstructed. The *fourth row* shows the thresholded frames, and the *fifth row* shows the extracted foregrounds after applying morphological operations (closing and dilation to fill holes).

little variation in the background, high F-measure values were obtained. 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 which belong to the foreground.

Precision: It measures the number of accurately detected foreground pixels which 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

$$F = 2.\frac{Precision \times Recall}{Precision + Recall}, \tag{11}$$

dl-DMD achieves high F-measure values in videos (003), (004) and (009) because the backgrounds of these videos are almost static for the entire duration and in video (002) background is static at different in-

Table 3: Evaluation results (BMC dataset).

Measure		BMC videos								
		001	002	003	004	005	006	007	008	009
RSL De La Torre (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
Erichson *et al.*(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 (proposed)	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**

tervals of time. dl-DMD can extract 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.

5 CONCLUSIONS

We proposed dl-DMD for accurate foreground extraction in videos. In dl-DMD, DMD is performed on coefficient matrices estimated over a dictionary that is learned on the randomly selected patches from the video frames. The experiments on synthetic data reveals that the use of dictionary with DMD can extract complex dynamics in time series data more accurately than standard DMD and cDMD methods. Also, experiments on real video dataset demonstrates that our proposed method can extract foreground and background information in videos with comparable performance to other methods.

ACKNOWLEDGEMENTS

This work was supported by JSPS KAKENHI (Grant Number JP18H03287) and JST CREST (Grant Number JPMJCR1913).

REFERENCES

Aharon, M., Elad, M., and Bruckstein, A. (2006). *rmk*-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322.

Bouwmans, T., Sobral, and Javed, S. (2017). Decomposition into low-rank plus additive matrices for background/foreground separation. *Computer Science Review*, 23:1–71.

Candès, E. J., Li, X., Ma, Y., and Wright, J. (2011). Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11.

De La Torre, F. and Black, M. J. (2003). A framework for robust subspace learning. *International Journal of Computer Vision*, 54(1-3):117–142.

Erichson, N. B., Brunton, S. L., and Kutz, J. N. (2016). Compressed dynamic mode decomposition for background modeling. *Journal of Real-Time Image Processing*, pages 1–14.

Goldfarb, D., Ma, S., and Scheinberg, K. (2013). Fast alternating linearization methods for minimizing the sum of two convex functions. *Mathematical Programming*, 141(1-2):349–382.

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

Koopman, B. (1931). Hamiltonian systems and transformation in Hilbert space. *Proceedings of the National Academy of Sciences USA*, 17(5):315–318.

Kutz, J. N. and Fu (2015). Multi-resolution dynamic mode decomposition for foreground/background separation and object tracking. In *2015 IEEE (ICCVW)*, pages 921–929. IEEE.

Oliver, N., Rosario, B., and Pentland, A. (1999). A bayesian computer vision system for modeling human interactions. In *ICVS*, pages 255–272. Springer.

Schmid, P. J. (2010). Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics*, 656:5–28.

Sobral, A. and Vacavant, A. (2014). A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Computer Vision and Image Understanding*, 122:4–21.

Takeishi, N., Kawahara, Y., and Yairi, T. (2017). Sparse non-negative dynamic mode decomposition. In *2017*

IEEE Int. Conf. on Image Process. (ICIP'17), pages 2682–2686.

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

Zhou, T. and Tao, D. (2011). Godec: Randomized low-rank & sparse matrix decomposition in noisy case. In *ICML*. Omnipress.