Computer Vision and Image Understanding 224 (2022) 103560

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

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

Quaternion-based dynamic mode decomposition for background modeling in color videos

Juan Han [a], Kit Ian Kou [a,*], Jifei Miao [b]

[a] Department of Mathematics, Faculty of Science and Technology, University of Macau, Macau 999078, China
[b] School of Mathematics and Statistics, Yunnan University, Kunming, Yunnan, 650091, China

ARTICLE INFO

Communicated by Nikos Paragios

Keywords:
Background model initialization
Color videos
Dynamic mode decomposition
Quaternion

ABSTRACT

Scene Background Initialization (SBI) is one of the challenging problems in computer vision. Dynamic mode decomposition (DMD) is a recently proposed method to robustly decompose a video sequence into the background model and the corresponding foreground part. However, this method needs to convert the color image into a grayscale image for processing, which leads to the neglect of the coupling information between the three channels of the color image. In this study, we propose a quaternion-based DMD (Q-DMD), which extends the DMD by quaternion matrix analysis, so as to ultimately preserve the inherent color structure of the color image and the color video. We exploit the standard eigenvalues of the quaternion matrix to compute its spectral decomposition and calculate the corresponding Q-DMD modes and eigenvalues. The results on the publicly available benchmark datasets prove that our Q-DMD outperforms the exact DMD method, and experiment results also demonstrate that the performance of our approach is comparable to that of the state-of-the-art ones.

1. Introduction

Scene Background Initialization (SBI) remains an important task in the field of scene background modeling, and it has a wide range of applications, such as video surveillance, video segmentation, video compression, video inpainting, and privacy protection for videos, computation photography, and so on. The aim of SBI is to extract a background model with no foreground objects from a video sequence whose background is occluded by any number of foreground objects (Maddalena and Petrosino, 2015). In real scenes, there exist challenges to achieving this aim such as sudden illumination changes, night videos, low framerate, dynamic background, camera jitter, and so on. In order to address these challenges, various background modeling methods have been proposed over the last few decades. Background modeling approaches based on deep learning have been developed in recent years. Schofield et al. (1996) was the first to propose a Random Access Memory (RAM) based neural network method to identify sections of the background scene in each test image. However, this approach requires the background of the scene to correctly represent the images. Maddalena and Petrosino (2008) proposed a method based on self-organization through artificial neural networks named Self Organizing Background Subtraction (SOBS) which learns background motion trajectories in a self-organizing manner. Convolutional Neural Networks (CNNs) have also been used for background modeling by Braham and Van Droogenbroeck (2016), Bautista et al. (2016), and Cinelli (2017). Halfaoui et al. (2016) proposed a reliable CNN-based approach to obtain the initial background of a scene by using just a small set of frames containing foreground objects. The Generative Adversarial Network (GAN) based models have also been proposed by researchers to handle the task of Scene Background Initialization, such as Sultana et al. (2019, 2020). For a detailed overview of deep learning-based methods, we recommend Bouwmans et al. (2019). However, deep learning-based approaches usually rely heavily on a large number of training samples, which are usually limited, and obtaining labeled data, in reality, is often laborious and sometimes difficult. Therefore, when the training data is limited (for example, only one color video is available and needs to be decomposed), it is necessary to develop effective non-learning methods.

In addition, among all the methods for separating background and foreground, one of the representative frameworks is to decompose video sequences into low-rank matrices (i.e., background) and sparse matrices (i.e., foreground). Based on this viewpoint, Candès et al. (2011) proposed the first framework of robust principal component analysis (RPCA). There are many variants of RPCA according to the difference in decomposition, loss function, optimization problem and the solver used. The recent review (Bouwmans et al., 2017b) provides a detailed overview of some traditional and state-of-the-art methods. By incorporating spatiotemporal sparse subspace clustering into the framework of RPCA, Javed et al. (2017, 2018) proposed a Motion-assisted Spatiotemporal Clustering of Low-rank (MSCL) approach for both the

[*] Corresponding author.
E-mail address: kikou@umac.mo (K.I. Kou).

https://doi.org/10.1016/j.cviu.2022.103560
Received 11 January 2022; Received in revised form 6 August 2022; Accepted 10 September 2022
Available online 16 September 2022
1077-3142/© 2022 Elsevier Inc. All rights reserved.

background estimation and foreground segmentation. A method named LaBGen was developed by Laugraud et al. (2017) for generating a stationary scene background using a pixel-wise temporal median filter and a patch selection mechanism based on motion detection.

A data matrix can also be decomposed into a low-rank matrix and a sparse matrix by using dynamic mode decomposition (DMD). This method can easily distinguish the static background and dynamic foreground from the data matrix by differentiating between the near-zero temporal Fourier modes and the remaining modes bounded away from the origin (Kutz et al., 2015). DMD also has many variants, such as standard DMD (Brunton and Kutz, 2019), exact DMD (Brunton and Kutz, 2019), multi-resolution DMD (Kutz et al., 2015, 2016), and compressed DMD (Erichson et al., 2019). However, when it comes to the application of background modeling of video frames, these variant methods deal with grayscale video sequences directly. For color video processing, DMD and its variants ignore the mutual connection among red, green, and blue channels, because these methods are applied to red, green, and blue channels separately, which may cause color distortion when separating the foreground and background of color video frames.

In this paper, we use a quaternion matrix to represent a color image, and then extend the DMD method to the quaternion system. Many studies have shown that quaternion representation of color images can achieve excellent results in color image processing problems, such as Ell and Sangwine (2007), Li et al. (2015), Zou et al. (2016), Hosny and Darwish (2019), Miao and Kou (2020), Miao et al. (2020), Miao and Kou (2021a) and so on. The quaternion-based method is to represent a pixel of a color image with a pure quaternion, that is, the three channels RGB of a color pixel correspond to the three imaginary parts of a quaternion, respectively, which can be shown by the following formula:

$$\dot{p} = 0 + p_R\, i + p_G\, j + p_B\, k,$$

where $\dot{p}$ represents a color pixel, and p_R, p_G, p_B correspond to the pixel values of the three channels RGB of this color pixel, and i, j, k are the three imaginary units of a quaternion.

The main contribution of this paper is to extend the DMD method to the quaternion system, so as to make full use of the coupling information between the three color channels RGB of color video sequences when separating video frames into background models and foreground components. To this end, we need to obtain the spectral decomposition of a quaternion matrix. However, a quaternion matrix has infinite right eigenvalues (Zhang, 1997), and its left eigenvalues are difficult to obtain. Therefore, in this paper, we establish the spectral decomposition of a quaternion matrix by using the standard eigenvalues of the quaternion matrix and the corresponding eigenvectors.

The outline of the rest of this paper is organized as follows. Section 2 sets some notations and introduces preliminaries for quaternion algebra. Section 3 briefly reviews the theory of the DMD method and the DMD method for background modeling. In Section 4, we introduce the eigenvalues, eigenvectors, spectral decomposition, and singular value decomposition of a quaternion matrix. Then we propose the quaternion-based DMD (Q-DMD) which can be used to separate a color video sequence into an approximate low-rank structure and a sparse structure. Section 5 provides some experiments on the basis of benchmark datasets to illustrate the performance of our approach, and compares our method with the DMD method and some state-of-the-art methods. Finally, some conclusions are given in Section 6.

2. Notations and preliminaries

In this section, we first summarize some main notations and then give a brief review of some basic knowledge of quaternion algebra. In addition, for a more complete introduction to quaternion algebra, we recommend Girard (2007).

2.1. Notations

In this paper, the set of real numbers, the set of complex numbers and the set of quaternions are denoted by $\mathbb{R}$, $\mathbb{C}$, $\mathbb{H}$, respectively. In addition, $\mathbb{C}^+$ denotes the set of complex numbers with nonnegative imaginary parts. $\mathbb{N}_+$ denotes the set of all positive integers. Lowercase letters, e.g., a, boldface lowercase letters, e.g., $\mathbf{a}$, and boldface capital letters, e.g., $\mathbf{A}$ represent scalars, vectors and matrices in real and complex fields, respectively. A quaternion scalar, a quaternion vector and a quaternion matrix are written as $\dot{q}$, $\dot{\mathbf{q}}$, $\dot{\mathbf{Q}}$, respectively. $(\cdot)^{-1}$, $(\cdot)^{\dagger}$, $(\cdot)^T$, $(\cdot)^*$, and $(\cdot)^H$ represent the inverse, pseudoinverse, transpose, conjugation, and conjugate transpose, respectively.

2.2. Basic knowledge of quaternion algebra

Quaternion was first proposed by Hamilton (1844). Let $\dot{q} \in \mathbb{H}$ be a quaternion,

$$\dot{q} = q_0 + q_1 i + q_2 j + q_3 k,$$

where q_0, q_1, q_2, $q_3 \in \mathbb{R}$, and i, j, k are imaginary number units which obey the quaternion rules that

$$\begin{cases} i^2 = j^2 = k^2 = ijk = -1, \\ ij = -ji = k, jk = -kj = i, ki = -ik = j. \end{cases}$$

For every quaternion $\dot{q} = q_0 + q_1 i + q_2 j + q_3 k$, it can be uniquely rewritten as $\dot{q} = q_0 + q_1 i + (q_2 + q_3 i)j = c_1 + c_2 j$, where c_1, $c_2 \in \mathbb{C}$. In addition, a quaternion $\dot{q} \in \mathbb{H}$ can be decomposed into a scalar part $S(\dot{q})$ and a vector part $\mathbf{V}(\dot{q})$, that is

$$\dot{q} = S(\dot{q}) + \mathbf{V}(\dot{q})$$

where $S(\dot{q}) = q_0 \in \mathbb{R}$, and $\mathbf{V}(\dot{q}) = \dot{q} - S(\dot{q}) = q_1 i + q_2 j + q_3 k$. A quaternion which satisfies $S(\dot{q}) = 0$, is called a pure quaternion. And $\mathbb{P}$ denotes the set of pure quaternions.

For a quaternion $\dot{q}$, its conjugate quaternion is defined as $\dot{q}^* = q_0 - q_1 i - q_2 j - q_3 k$. And the norm of a quaternion $\dot{q}$ is defined as $|\dot{q}| = \sqrt{qq^*} = \sqrt{q^* q} = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}$. Different from the complex number field, the commutative law is generally not valid in the quaternion system, i.e., $\dot{q}_1 \dot{q}_2 \neq \dot{q}_2 \dot{q}_1$ in general.

Now, we introduce the definition of exponential and logarithm of a quaternion. Every (non-null) pure quaternion $\dot{\xi}$ can be presented by $\dot{\xi} = |\dot{\xi}|\tilde{\dot{\xi}}$, where $\tilde{\dot{\xi}}$ is a unit pure quaternion (i.e., $\tilde{\dot{\xi}} \in \mathbb{P}$, and $|\tilde{\dot{\xi}}| = 1$). And $\mathbb{U}$ denotes the set of unit pure quaternions.

Definition 1 (*The Exponential Function of a Pure Quaternion Ell et al., 2014*)**.** Assuming that $\dot{\xi}$ is a (non-null) pure quaternion, then its exponential function $\exp : \mathbb{P} \to \mathbb{H}$ can be defined by exploiting its power series expansion, which is given by the following formula:

$$\begin{aligned} e^{\dot{\xi}} &= \sum_{n=0}^{+\infty} \frac{\dot{\xi}^n}{n!} \\ &= \sum_{n=0}^{+\infty} \frac{|\dot{\xi}|^n \tilde{\dot{\xi}}^n}{n!} \\ &= \sum_{m=0}^{+\infty} (-1)^m \frac{|\dot{\xi}|^{2m}}{(2\ m)!} + \tilde{\dot{\xi}} \sum_{m=0}^{+\infty} (-1)^m \frac{|\dot{\xi}|^{2m+1}}{(2m+1)!} \\ &= \cos|\dot{\xi}| + \tilde{\dot{\xi}} \sin|\dot{\xi}|, \end{aligned} \tag{1}$$

since $\tilde{\dot{\xi}} \in \mathbb{U}$ satisfies the following formula:

$$\tilde{\dot{\xi}}^n = \begin{cases} (-1)^m, & \text{if} \quad n = 2\ m, \\ (-1)^m \tilde{\dot{\xi}}, & \text{if} \quad n = 2m + 1. \end{cases}$$

The exponential function of a pure quaternion can be easily represented by cosine and sine functions just as in the complex case. The difference is that the axis $\tilde{\xi}$ is a unit pure quaternion, while the argument is the modulus of $\dot{\xi}$. Obviously, the exponential of a pure quaternion is a full quaternion, with real part $\cos|\dot{\xi}|$ and vector part $\widetilde{\dot{\xi}}\sin|\dot{\xi}|$. Another important property different from the complex exponential case is that, in general, the product of two exponents of a pure quaternion is not an exponent whose argument is equal to the sum of the original exponential arguments (Ell et al., 2014), that is, for $\dot{p},\ \dot{q} \in \mathbb{P}$ with different unit pure quaternions $\widetilde{\dot{p}}$ and $\widetilde{\dot{q}}$, then

$$e^{|\dot{p}|\widetilde{\dot{p}}}e^{|\dot{q}|\widetilde{\dot{q}}} \neq e^{|\dot{p}|\widetilde{\dot{p}}+|\dot{q}|\widetilde{\dot{q}}}.$$

The exponential function of full quaternions can be defined based on Definition 1.

Definition 2 (*The Exponential Function of a Full Quaternion Ell et al., 2014*)**.** For a quaternion $\dot{q}$, its exponential function $\exp : \mathbb{H} \to \mathbb{H}$ is given by

$$\begin{aligned} e^{\dot{q}} &= \sum_{n=0}^{+\infty} \frac{\dot{q}^n}{n!} \\ &= e^{S(\dot{q})}e^{\mathbf{V}(\dot{q})} \\ &= e^{S(\dot{q})}(\cos|\mathbf{V}(\dot{q})| + \widetilde{\mathbf{V}(\dot{q})}\sin|\mathbf{V}(\dot{q})|), \end{aligned} \tag{2}$$

where $\mathbf{V}(\dot{q}) = |\mathbf{V}(\dot{q})|\widetilde{\mathbf{V}(\dot{q})} \in \mathbb{P}$, and $\widetilde{\mathbf{V}(\dot{q})} \in \mathbb{U}$ is the normalized vector part of $\dot{q}$.

Definition 3 (*The Logarithm of a Quaternion Ell et al., 2014*)**.** The logarithm of the quaternion $\dot{q}$ is the inverse of the exponential function. This means that for $\dot{p},\ \dot{q} \in \mathbb{H}$, if

$$e^{\dot{p}} = \dot{q},$$

then

$$\dot{p} = \ln \dot{q}. \tag{3}$$

There also exists an expression for the logarithm of $\dot{q} = q_0+q_1 i+q_2 j+q_3 k$ in terms of its elements, which can be considered as a generalization of the logarithm of a complex number and is given by the following formula:

$$\ln \dot{q} = \ln|\dot{q}| + \dot{\mu}_{\dot{q}}\phi_{\dot{q}}, \tag{4}$$

where

$$\begin{cases} \dot{\mu}_{\dot{q}} = \frac{\mathbf{V}(\dot{q})}{|\mathbf{V}(\dot{q})|}, \\ \phi_{\dot{q}} = \arctan(\frac{|\mathbf{V}(\dot{q})|}{S(\dot{q})}). \end{cases}$$

Similarly, a quaternion matrix $\dot{\mathbf{Q}} = (\dot{q}_{m,n}) \in \mathbb{H}^{M\times N}$ is denoted as $\dot{\mathbf{Q}} = \mathbf{Q}_0 + \mathbf{Q}_1 i + \mathbf{Q}_2 j + \mathbf{Q}_3 k$ with $\mathbf{Q}_t \in \mathbb{R}^{M\times N}(t = 0, 1, 2, 3)$. If $\mathbf{Q}_0 = 0$, then the corresponding quaternion matrix is called a pure quaternion matrix.

3. Dynamic mode decomposition in real field

3.1. Dynamic mode decomposition method

DMD is a data-driven method which spatiotemporally decomposes data from snapshots or measurements of a given system in time into a set of dynamic modes. In this section, we will give a brief review of the DMD theory. For specific details of DMD, please refer to Brunton and Kutz (2019), Kutz et al. (2015, 2016), and Erichson et al. (2019).

It is assumed here that the number of data points collected at a given time is n, with the number of unified samples being m, and the timestep is denoted by Δt. We use a vector $\mathbf{x}_l \in \mathbb{R}^n$ to denote the n data points collected at the time $t_l = l\Delta t$, $l = 1, 2, \dots, m$. Then, all the data points can form a real matrix $\mathbf{Z} = \begin{bmatrix}\mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_m\end{bmatrix} \in \mathbb{R}^{n\times m}$. And the data can be arranged into two matrices, $\mathbf{X}$, $\mathbf{Y}$:

$$\mathbf{X} = \begin{bmatrix}\mathbf{x}_1 & \mathbf{x}_2 & \mathbf{x}_3 & \cdots & \mathbf{x}_{m-1}\end{bmatrix},$$

$$\mathbf{Y} = \begin{bmatrix}\mathbf{x}_2 & \mathbf{x}_3 & \mathbf{x}_4 & \cdots & \mathbf{x}_m\end{bmatrix},$$

where $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{n\times(m-1)}$. Assuming that there exists a linear operator $\mathbf{A} \in \mathbb{R}^{n\times n}$ which describes the dynamic change between the data at time $t_l = l\Delta t$ and the data at time $t_{l+1} = (l+1)\Delta t$ such that $\mathbf{x}_{l+1} = \mathbf{A}\mathbf{x}_l$. Based on the assumption, the best fit linear map $\widehat{\mathbf{A}}$ which maps $\mathbf{X}$ to $\mathbf{Y}$ can be estimated in a least-squares sense. Suppose

$$\mathbb{R}_L = \{\mathbf{A}|\mathbf{A} \in \mathbb{R}^{n\times n},\ \|\mathbf{Y} - \mathbf{A}\mathbf{X}\|_F = \min\}, \tag{5}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. $\mathbb{R}_L$ is the set of least squares solutions of the matrix equation $\mathbf{A}\mathbf{X} = \mathbf{Y}$. Then the best fit linear map $\widehat{\mathbf{A}}$ is given by

$$\widehat{\mathbf{A}} = \mathbf{Y}\mathbf{X}^{\dagger}, \tag{6}$$

where $\mathbf{X}^{\dagger}$ is the pseudoinverse of $\mathbf{X}$, and $\widehat{\mathbf{A}}$ is the minimum norm least squares solution of the matrix equation $\mathbf{A}\mathbf{X} = \mathbf{Y}$. The eigenvectors and eigenvalues of $\widehat{\mathbf{A}}$ are defined as the DMD modes and eigenvalues, respectively. In many systems, the dimension n of these systems is larger than the number of snapshots m. However, when n is large, it is intractable to directly compute the spectral decomposition of the matrix $\widehat{\mathbf{A}}$.

Therefore, for high-dimensional data $\mathbf{x} \in \mathbb{R}^n$ (i.e., n is large), instead of explicitly computing $\widehat{\mathbf{A}}$ to obtain its dominant eigenvalues and eigenvectors, the DMD method obtains the dominant eigenvalues and eigenvectors of $\widehat{\mathbf{A}}$ by implementing dimensionality reduction. There are many variants of DMD. The main difference between those methods is the way to calculate the DMD modes and eigenvalues. In this paper, we will expand the exact DMD to the quaternion system, so we give a brief review of the exact DMD in this section. For more details please refer to Tu (2013).

The computation process of the exact DMD algorithm is as follows (Tu, 2013):

1. First, compute the SVD of the matrix $\mathbf{X}$:

   $$\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T, \tag{7}$$

   where $\mathbf{U} \in \mathbb{R}^{n\times r}$, $\boldsymbol{\Sigma} \in \mathbb{R}^{r\times r}$, and $\mathbf{V} \in \mathbb{R}^{m\times r}$, and $r \leq m \leq n$ is the rank (or the approximate rank) of $\mathbf{X}$.
2. Second, after calculating the pseudoinverse of $\mathbf{X}$, the full matrix $\widehat{\mathbf{A}}$ can be obtained as follows:

   $$\widehat{\mathbf{A}} = \mathbf{Y}\mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{U}^T. \tag{8}$$

   Instead of calculating $\widehat{\mathbf{A}}$ directly, the leading r eigenvalues and eigenvectors of $\widehat{\mathbf{A}}$ are computed by using dimensionality reduction, projecting $\widehat{\mathbf{A}}$ onto these leading singular vectors and resulting in a small matrix $\widetilde{\mathbf{A}}$:

   $$\widetilde{\mathbf{A}} = \mathbf{U}^T\widehat{\mathbf{A}}\mathbf{U} = \mathbf{U}^T\mathbf{Y}\mathbf{V}\boldsymbol{\Sigma}^{-1}, \tag{9}$$

   where $\widetilde{\mathbf{A}} \in \mathbb{R}^{r\times r}$. The reduced matrix $\widetilde{\mathbf{A}}$ has the same nonzero eigenvalues as the full matrix $\widehat{\mathbf{A}}$.
3. Next, compute the spectral decomposition of $\widetilde{\mathbf{A}}$, obtaining $\widetilde{\mathbf{W}}$ and $\widetilde{\Lambda}$ such that

   $$\widetilde{\mathbf{A}}\widetilde{\mathbf{W}} = \widetilde{\mathbf{W}}\widetilde{\Lambda}, \tag{10}$$

   where $\widetilde{\Lambda}$ is a diagonal matrix and its diagonal elements are the eigenvalues λ_s of both the matrix $\widetilde{\mathbf{A}}$ and the matrix $\widehat{\mathbf{A}}$, i.e., the DMD eigenvalues, while the columns of the matrix $\widetilde{\mathbf{W}}$ are the corresponding eigenvectors of the matrix $\widetilde{\mathbf{A}}$.
4. Finally, the leading spectral decomposition of $\widehat{\mathbf{A}}$ are obtained based on Eq. (10). Calculate the DMD modes $\boldsymbol{\Phi}$ by using the eigenvectors $\widetilde{\mathbf{W}}$ of the matrix $\widetilde{\mathbf{A}}$, given by

   $$\boldsymbol{\Phi} = \mathbf{Y}\mathbf{V}\boldsymbol{\Sigma}^{-1}\widetilde{\mathbf{W}}. \tag{11}$$

After obtaining the low-rank approximations of the eigenvalues and eigenvectors, i.e., the DMD eigenvalues and the DMD modes, the approximate data can be obtained for all times in the future. The approximated data $\mathbf{x}_{\mathrm{DMD}}$ at time $t_\upsilon = \upsilon\Delta t$ ($\upsilon \in \mathbb{N}_+$) for any time after the data vector $\mathbf{x}_1$ was collected can be obtained by $\mathbf{x}_{\mathrm{DMD}}(t_v) = \widehat{\mathbf{A}}^{v-1}\mathbf{x}_1$. For the convenience of spectral expansion in continuous time, a mapping of λ_s is defined as $\omega_s = \log(\lambda_s)/\Delta t$, $s = 1, 2, \ldots, r$. Then the approximate system state at all future times t_v, $\mathbf{x}_{\mathrm{DMD}}(t_v)$, is given by

$$\mathbf{x}_{\mathrm{DMD}}(t_v) = \sum_{s=1}^{r} b_s \boldsymbol{\varphi}_s e^{\omega_s (v-1)\Delta t} = \boldsymbol{\Phi}\boldsymbol{\Omega}^{(v-1)\Delta t}\mathbf{b}, \tag{12}$$

where

$$\boldsymbol{\Omega} = \begin{bmatrix} e^{\omega_1} & 0 & \cdots & 0 \\ 0 & e^{\omega_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e^{\omega_r} \end{bmatrix},$$

and $\mathbf{b}$ is generally calculated as

$$\mathbf{b} = \boldsymbol{\Phi}^{\dagger}\mathbf{x}_1 = (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\mathbf{x}_1. \tag{13}$$

The method is shown in Algorithm 1.

Algorithm 1 Exact DMD algorithm (Tu, 2013).

```
Input: Two matrices $\mathbf{X}$ and $\mathbf{Y}$ which are constructed from the data.
Output: DMD modes ($\boldsymbol{\Phi}$) and DMD eigenvalues.
$\mathbf{U}, \boldsymbol{\Sigma}, \mathbf{V} = \mathrm{SVD}(\mathbf{X}, r) \longleftarrow$ (reduced) QSVD.
$\widetilde{\mathbf{A}} \longleftarrow \mathbf{U}^H\mathbf{Y}\mathbf{V}\boldsymbol{\Sigma}^{-1}$.
$\widetilde{\mathbf{W}}, \widetilde{\boldsymbol{\Lambda}} \longleftarrow$ eigenvectors and eigenvalues of $\widetilde{\mathbf{A}}$
$\boldsymbol{\Phi} = \mathbf{Y}\mathbf{V}\boldsymbol{\Sigma}^{-1}\widetilde{\mathbf{W}} \longleftarrow$ Q-DMD modes.
return $\boldsymbol{\Phi}$ and $\widetilde{\boldsymbol{\Lambda}}$;
```

3.2. Background modeling using the dynamic mode decomposition method

Assuming that the total number of frames in a video sequence is m, and each frame has n pixels in total, then each frame can be vectorized into a vector denoted as $\mathbf{x}_s$, where $\mathbf{x}_s \in \mathbb{R}^n$, and $s = 1, 2, \ldots, m$. Assuming that frames are collected at evenly spaced time intervals Δt, then the time points of these collected frames can form a vector $\mathbf{t} = \begin{bmatrix} t_1 & t_2 & \cdots & t_m \end{bmatrix}$. For convenience, the time point of collecting the first frame is recorded as 0, and the time points of collecting the second frame to the mth frame are similarly recorded as 1 to $m-1$, i.e., $\mathbf{t} = \begin{bmatrix} 0 & 1 & \cdots & m-1 \end{bmatrix}$. Therefore, using the Q-DMD method, the full video sequence $\mathbf{X}$ can be reconstructed by the following formula

$$\mathbf{X}_{\mathrm{DMD}} = \sum_{s=1}^{r} b_s \boldsymbol{\varphi}_s e^{\omega_s \mathbf{t}} = \boldsymbol{\Phi}\boldsymbol{\Omega}^{\mathbf{t}}\mathbf{b}, \tag{14}$$

where $\boldsymbol{\varphi}_s \in \mathbb{R}^n$, $\mathbf{t} \in \mathbb{R}^{1\times m}$. The parameter r is related to the dimensionality reduction and it is fixed to $m-1$, i.e., one less than the number of frames of the video sequence in this paper. The key principle to separate the reconstructed video frames into the background (approximately low-rank) and foregrounds (approximately sparse) is to threshold the low-frequency modes of the corresponding eigenvalues. Generally, the portion of the first video frame that does not change with time, or changes very slowly in time satisfies $|\omega_p| \approx 0$, and is considered as the background. Assume that there exists a ω_p that satisfies $|\omega_p| \approx 0$, where $p \in \{1, 2, \ldots, r\}$, and the others (i.e. ω_s, $\forall\, s \neq p$) are bounded away from zero. Then we can obtain that

$$\begin{aligned} \mathbf{X}_{\mathrm{DMD}} &= \mathbf{L} + \mathbf{S} \\ &= \underbrace{b_p \boldsymbol{\varphi}_p e^{\omega_p \mathbf{t}}}_{\text{Background}} + \underbrace{\sum_{s \neq p} b_s \boldsymbol{\varphi}_s e^{\omega_s \mathbf{t}}}_{\text{Foreground}}. \end{aligned} \tag{15}$$

As can be seen from Eq. (15), each term of the DMD reconstruction is a complex matrix, i.e., $b_s\boldsymbol{\varphi}_s e^{\omega_s \mathbf{t}} \in \mathbb{C}^{n\times m}$. However, the reconstructed $\mathbf{X}_{\mathrm{DMD}}$ should be a real matrix because the matrix $\mathbf{Z} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_m \end{bmatrix} \in \mathbb{R}^{n\times m}$. Therefore, the modulus of each element within the matrix $\mathbf{X}_{\mathrm{DMD}}$ is calculated and used to estimate $\mathbf{X}_{\mathrm{DMD}}$. For specific details of DMD and the DMD method for background modeling, please refer to Brunton and Kutz (2019), Kutz et al. (2015, 2016), and Erichson et al. (2019).

4. Proposed quaternion-based dynamic mode decomposition

In this section, we will first introduce the calculation of eigenvalues, eigenvectors, and singular value decomposition of a quaternion matrix. Then we will extend DMD to the quaternion number system. The purpose is to use the advantages of quaternion for representing color images.

4.1. Eigenvalues and eigenvectors of a quaternion matrix

Due to the non-commutation of quaternion, there are two categories of eigenvalues of a quaternion matrix, left eigenvalues and right eigenvalues. In this paper, we only use the right eigenvalues, so a brief review of the right eigenvalues and eigenvectors of a quaternion matrix will be given. For convenience, we abbreviate right eigenvalues as the eigenvalues. For detailed overview of eigenvalues and eigenvectors of a quaternion matrix, we recommend Lee (1948) and Zhang (1997).

Definition 4 (*Eigenvalues Zhang, 1997*)**.** A quaternion $\dot{\lambda}$ is said to be a right (left) eigenvalue of a quaternion matrix $\dot{\mathbf{Q}}$ if it satisfies

$$\dot{\mathbf{Q}}\dot{\mathbf{x}} = \dot{\mathbf{x}}\dot{\lambda} \quad (\dot{\mathbf{Q}}\dot{\mathbf{x}} = \dot{\lambda}\dot{\mathbf{x}}). \tag{16}$$

Definition 5 (*Eigenvalue Class Chang et al., 2003*)**.** If $\dot{\lambda}$ is one right eigenvalue of a quaternion matrix $\dot{\mathbf{Q}} \in \mathbb{H}^{N\times N}$, then every element of the set $\Gamma \equiv \{\dot{q}\dot{\lambda}\dot{q}^{-1}$: where $\dot{q}$ is any unit quaternion with its norm $|\dot{q}| = 1\}$ is also a right eigenvalue of $\dot{\mathbf{Q}}$. Moreover, a single eigenvalue $\lambda_c \in \mathbb{C}^+$ will be contained in this set Γ, so this set is considered as the eigenvalue class of λ_c.

Therefore, the eigenvalues of a quaternion matrix are infinite, and the eigenvalues of a quaternion matrix are finite if and only if all eigenvalues of this quaternion matrix are real, however, there are finite eigenvalue classes.

Theorem 1 (*The Standard Eigenvalues of a Quaternion Matrix Zhang, 1997*)**.** *Any $N \times N$ quaternion matrix $\dot{\mathbf{Q}}$ has exactly N right eigenvalues which are complex numbers with nonnegative imaginary parts. Those eigenvalues are defined as the standard eigenvalues of the quaternion matrix $\dot{\mathbf{Q}}$.*

Definition 6 (*The Complex Representation of a Quaternion Matrix*)**.** Given a quaternion matrix $\dot{\mathbf{Q}} \in \mathbb{H}^{M\times N}$, and let $\dot{\mathbf{Q}} = \mathbf{Q}_0 + \mathbf{Q}_1 i + \mathbf{Q}_2 j + \mathbf{Q}_3 k = \mathbf{Q_a} + \mathbf{Q_b} j$, where $\mathbf{Q_a} = \mathbf{Q}_0 + \mathbf{Q}_1 i$, $\mathbf{Q_b} = \mathbf{Q}_2 + \mathbf{Q}_3 i \in \mathbb{C}^{M\times N}$, then the complex representation of $\dot{\mathbf{Q}}$ is defined as (Zhang, 1997)

$$\chi_{\dot{Q}} = \begin{bmatrix} \mathbf{Q_a} & \mathbf{Q_b} \\ -\mathbf{Q_b}^* & \mathbf{Q_a}^* \end{bmatrix}, \tag{17}$$

where $\mathbf{Q_a}^* = \mathbf{Q}_0 - \mathbf{Q}_1 i$, $\mathbf{Q_b}^* = \mathbf{Q}_2 - \mathbf{Q}_3 i$, and $\chi_{\dot{Q}} \in \mathbb{C}^{2M\times 2N}$.

There are many similar properties between the quaternion matrix and its corresponding complex representation matrix and the detail can be found in Zhang (1997).

Based on the eigenvalues and eigenvectors of $\chi_{\dot{Q}}$, we can compute the eigenvalues and eigenvectors of $\dot{\mathbf{Q}}$, which is presented in Theorem 2.

Theorem 2 (*The Calculation of Standard Eigenvalues of the Quaternion Matrix Chang et al., 2003*)**.** *Given a quaternion matrix $\dot{\mathbf{Q}} \in \mathbb{H}^{N\times N}$, then the complex eigenvalues of $\dot{\mathbf{Q}}$ are the same as the eigenvalue of $\chi_{\dot{Q}}$. Further, the complex eigenvalues of $\chi_{\dot{Q}}$ appear in conjugate pairs. Specially, if $\chi_{\dot{Q}}$ has any real eigenvalue, it occurs an even number of times. Therefore, N complex eigenvalues with nonnegative imaginary part of the quaternion matrix $\dot{\mathbf{Q}}$ can be obtained.*

The relation between the eigenvectors of the quaternion matrix $\dot{\mathbf{Q}}$ and the eigenvectors of $\chi_{\dot{Q}}$ is that if $(\mathbf{x})_{2N\times 1} = \begin{bmatrix}(\mathbf{x}_1)_{N\times 1}\\(\mathbf{x}_2)_{N\times 1}\end{bmatrix}$ is an eigenvector of the complex matrix $\chi_{\dot{Q}}$ corresponding to eigenvalue λ of $\chi_{\dot{Q}}$, then $(\dot{\mathbf{x}})_{N\times 1} = \mathbf{x}_1 - \mathbf{x}_2^* j$ is an eigenvector of the quaternion matrix $\dot{\mathbf{Q}}$ corresponding to eigenvalue λ of $\dot{\mathbf{Q}}$, where $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{C}^{N\times 1}$, and $\dot{\mathbf{x}} \in \mathbb{H}^{N\times 1}$.

4.2. Singular value decomposition of a quaternion matrix

Definition 7 (*The Rank of Quaternion Matrix Zhang, 1997*)**.** The rank of a quaternion matrix $\dot{\mathbf{Q}} \in \mathbb{H}^{M\times N}$ is the maximum number of right (left) linearly independent columns (rows) of $\dot{\mathbf{Q}}$.

Theorem 3 (*Singular Value Decomposition of a Quaternion Matrix (QSVD) Zhang, 1997*)**.** *Given any quaternion matrix $\dot{\mathbf{Q}} \in \mathbb{H}^{M\times N}$ of rank r, there exist two quaternion unitary matrices $\dot{\mathbf{U}} \in \mathbb{H}^{M\times M}$ and $\dot{\mathbf{V}} \in \mathbb{H}^{N\times N}$ such that*

$$\dot{\mathbf{Q}} = \dot{\mathbf{U}} \begin{bmatrix} \boldsymbol{\Sigma}_r & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \dot{\mathbf{V}}^H = \dot{\mathbf{U}}\boldsymbol{\Lambda}\dot{\mathbf{V}}^H, \tag{18}$$

where $\boldsymbol{\Sigma}_r$ is a real diagonal matrix with r positive entries on its diagonal (i.e. singular values of $\dot{\mathbf{Q}}$).

The computation of $\dot{\mathbf{U}}$, $\dot{\mathbf{V}}$ and the singular values of $\dot{\mathbf{Q}}$ can be obtained based on the SVD of its complex representation $\chi_{\dot{Q}}$. The calculation of QSVD is briefly summarized as follows (Xu et al., 2015):

1. Compute the SVD of $\chi_{\dot{Q}}$, and here we denote $\chi_{\dot{Q}} = \mathbf{U}\boldsymbol{\Lambda}^{\chi_{\dot{Q}}}\mathbf{V}^H$.
2. Then, we can get that

$$\begin{cases} \boldsymbol{\Lambda} = \mathrm{row}_{odd}(\mathrm{col}_{odd}(\boldsymbol{\Lambda}^{\chi_{\dot{Q}}})), \\ \dot{\mathbf{U}} = \mathrm{col}_{odd}(\mathbf{U}_1) + \mathrm{col}_{odd}(-(\mathbf{U}_2)^*)j, \\ \dot{\mathbf{V}} = \mathrm{col}_{odd}(\mathbf{V}_1) + \mathrm{col}_{odd}(-(\mathbf{V}_2)^*)j, \end{cases} \tag{19}$$

where

$$\mathbf{U} = \begin{bmatrix}(\mathbf{U}_1)_{M\times 2M}\\(\mathbf{U}_2)_{M\times 2M}\end{bmatrix}, \quad \mathbf{V} = \begin{bmatrix}(\mathbf{V}_1)_{N\times 2N}\\(\mathbf{V}_2)_{N\times 2N}\end{bmatrix},$$

and $\mathrm{row}_{odd}(\mathbf{M})$ and $\mathrm{col}_{odd}(\mathbf{M})$ represent the extraction of the odd rows and odd columns of matrix $\mathbf{M}$ respectively.

4.3. Computation of quaternion-based dynamic mode decomposition modes and eigenvalues

Assume that there are m snapshots of the state of a dynamic system and each snapshot is arranged into an $n \times 1$ quaternion vector with the form

$$\dot{\mathbf{X}}(t_l) \in \mathbb{H}^n,$$

where $l = 1, 2, \ldots, m$. These snapshots can form two data matrices, $\dot{\mathbf{X}}$ and $\dot{\mathbf{Y}} \in \mathbb{H}^{n\times(m-1)}$:

$$\dot{\mathbf{X}} = \begin{bmatrix}\dot{\mathbf{X}}(t_1) & \dot{\mathbf{X}}(t_2) & \dot{\mathbf{X}}(t_3) & \cdots & \dot{\mathbf{X}}(t_{m-1})\end{bmatrix},$$

$$\dot{\mathbf{Y}} = \begin{bmatrix}\dot{\mathbf{X}}(t_2) & \dot{\mathbf{X}}(t_3) & \dot{\mathbf{X}}(t_4) & \cdots & \dot{\mathbf{X}}(t_m)\end{bmatrix}.$$

Assuming that the data was collected by uniform sampling in time, with $t_l = l\Delta t$ for $l = 1, 2, \ldots, m$, and Δt is the timestep. Similarly, we assume that there is a linear operator $\dot{\mathbf{Q}}$ which reflects the changes of the two matrices $\dot{\mathbf{X}}$ and $\dot{\mathbf{Y}}$:

$$\dot{\mathbf{Y}} \approx \dot{\mathbf{Q}}\dot{\mathbf{X}}. \tag{20}$$

Mathematically, the quaternionic least squares (QLS) problem can be formulated as:

$$\mathbb{H}_L = \{\dot{\mathbf{Q}} | \dot{\mathbf{Q}} \in \mathbb{H}^{n\times n}, \|\dot{\mathbf{Y}} - \dot{\mathbf{Q}}\dot{\mathbf{X}}\|_F = \min\}, \tag{21}$$

where $\|\cdot\|_F$ denotes the Frobenius norm, and $\mathbb{H}_L$ is the set of least squares solutions of Eq. (20). The Frobenius norm of a quaternion matrix $\dot{\mathbf{Q}} \in \mathbb{H}^{N_1\times M_1}$ is defined as (Zhang, 1997): $\|\dot{\mathbf{Q}}\|_F = \sqrt{\sum_{n_1=1}^{N_1}\sum_{m_1=1}^{M_1} |\dot{q}_{n_1 m_1}|^2} = \sqrt{\mathrm{tr}((\dot{\mathbf{Q}})^H \dot{\mathbf{Q}})}$. And the minimum norm least squares solution of Eq. (20) is

$$\hat{\dot{\mathbf{Q}}} = \dot{\mathbf{Y}}\dot{\mathbf{X}}^\dagger, \tag{22}$$

where $\dot{\mathbf{X}}^\dagger$ is the quaternionic pseudoinverse of $\dot{\mathbf{X}}$ (Kyrchei, 2013). When the dimension of per time snapshot n is large, it is difficult to deal with the quaternion matrix $\hat{\dot{\mathbf{Q}}}$ directly. Therefore, we also reduce the dimension of the quaternion matrix $\hat{\dot{\mathbf{Q}}}$ to obtain its leading spectral decomposition. Before that, we first establish the special spectral decomposition of a quaternion matrix.

Theorem 4 (*Spectral Decomposition of a Quaternion Matrix*)**.** *Given any quaternion matrix $\dot{\mathbf{Q}} \in \mathbb{H}^{M\times M}$, after calculating the standard eigenvalues and eigenvectors of $\dot{\mathbf{Q}}$, there exist two quaternion matrices $\dot{\boldsymbol{\Phi}} \in \mathbb{H}^{M\times M}$ and $\dot{\boldsymbol{\Lambda}} \in \mathbb{H}^{M\times M}$ such that*

$$\dot{\mathbf{Q}}\dot{\boldsymbol{\Phi}} = \dot{\boldsymbol{\Phi}}\dot{\boldsymbol{\Lambda}}, \tag{23}$$

where $\dot{\boldsymbol{\Lambda}}$ is a quaternion diagonal matrix with M quaternion numbers on its diagonal, i.e., M standard eigenvalues of $\dot{\mathbf{Q}}$, and the columns of $\dot{\boldsymbol{\Phi}}$ are the corresponding eigenvectors.

Proof. By using Theorem 2, we can calculate the M standard eigenvalues of $\dot{\mathbf{Q}}$ and its corresponding M eigenvectors. We denote the vth eigenvalue of $\dot{\mathbf{Q}}$ as $\dot{\lambda}_v$, and $\dot{\boldsymbol{\phi}}_v$ is the eigenvector corresponding to $\dot{\lambda}_v$. Then the M eigenvectors can form a quaternion matrix which is denoted as $\dot{\boldsymbol{\Phi}}$, i.e., $\dot{\boldsymbol{\Phi}} = \begin{bmatrix}\dot{\boldsymbol{\phi}}_1 & \dot{\boldsymbol{\phi}}_2 & \cdots & \dot{\boldsymbol{\phi}}_M\end{bmatrix}$. Meanwhile another diagonal quaternion matrix $\dot{\boldsymbol{\Lambda}}$ can be formed, that is $\dot{\boldsymbol{\Lambda}} = \mathrm{diag}(\dot{\lambda}_1, \dot{\lambda}_2, \ldots, \dot{\lambda}_M)$, where $\dot{\boldsymbol{\Lambda}}$ is a quaternion diagonal matrix and has the M standard eigenvalues of $\dot{\mathbf{Q}}$ on its diagonal. Therefore, based on Definition 4, we have

$$\dot{\mathbf{Q}}\dot{\boldsymbol{\Phi}} = \dot{\boldsymbol{\Phi}}\dot{\boldsymbol{\Lambda}}. \quad \square \tag{24}$$

Now, we calculate the quaternion-based DMD (Q-DMD) modes and eigenvalues. The calculation process is as follows:

1. Firstly, we calculate the (reduced) QSVD of the matrix $\dot{\mathbf{X}}$, i.e. find $\dot{\mathbf{U}}$, $\boldsymbol{\Sigma}$ and $\dot{\mathbf{V}}$ such that

   $$\dot{\mathbf{X}} = \dot{\mathbf{U}}\boldsymbol{\Sigma}\dot{\mathbf{V}}^H, \tag{25}$$

   where $\dot{\mathbf{U}} \in \mathbb{H}^{n\times r}$, $\boldsymbol{\Sigma} \in \mathbb{R}^{r\times r}$, and $\dot{\mathbf{V}} \in \mathbb{H}^{m\times r}$, and $r \leq m \leq n$ is the rank (or the approximate rank) of $\dot{\mathbf{X}}$. The rank (or the approximate rank) r is related to the dimensionality reduction of the quaternion matrix $\hat{\dot{\mathbf{Q}}}$ and the low-rank trunction of the data. We note that $\dot{\mathbf{U}}^H\dot{\mathbf{U}} = \mathbf{I}$, $\dot{\mathbf{V}}^H\dot{\mathbf{V}} = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix.
2. Secondly, we compute the reduced quaternion matrix $\tilde{\dot{\mathbf{Q}}}$. According to Eq. (22), after calculating the quaternionic pseudoinverse of $\dot{\mathbf{X}}$, the full quaternion matrix $\hat{\dot{\mathbf{Q}}}$ can be obtained as follows:

   $$\hat{\dot{\mathbf{Q}}} = \dot{\mathbf{Y}}\dot{\mathbf{V}}\boldsymbol{\Sigma}^{-1}\dot{\mathbf{U}}^H. \tag{26}$$

   However, we are only interested in the leading spectral decomposition of $\hat{\dot{\mathbf{Q}}}$, and the reduced quaternion matrix $\tilde{\dot{\mathbf{Q}}}$ can be obtained by the following formula:

   $$\tilde{\dot{\mathbf{Q}}} = \dot{\mathbf{U}}^H\hat{\dot{\mathbf{Q}}}\dot{\mathbf{U}} = \dot{\mathbf{U}}^H\dot{\mathbf{Y}}\dot{\mathbf{V}}\boldsymbol{\Sigma}^{-1}. \tag{27}$$
3. Next, we compute the spectral decomposition of $\tilde{\dot{\mathbf{Q}}}$ by using Theorem 4:

   $$\tilde{\dot{\mathbf{Q}}}\widetilde{\dot{\mathbf{W}}} = \widetilde{\dot{\mathbf{W}}}\tilde{\dot{\boldsymbol{\Lambda}}}, \tag{28}$$

   where $\tilde{\dot{\boldsymbol{\Lambda}}}$ is a diagonal matrix and its diagonal elements are the standard eigenvalues ($\dot{\lambda}_s \in \mathbb{H}$) of the matrix $\tilde{\dot{\mathbf{Q}}}$, which are defined as the Q-DMD eigenvalues, while the columns of the matrix $\widetilde{\dot{\mathbf{W}}}$ are the corresponding eigenvectors of the matrix $\tilde{\dot{\mathbf{Q}}}$.

4. Finally, the leading spectral decomposition of $\hat{\dot{\mathbf{Q}}}$ are obtained based on Eq. (28). The standard eigenvalues of $\hat{\dot{\mathbf{Q}}}$ are given by Q-DMD eigenvalues, i.e., the diagonal elements of $\tilde{\dot{\boldsymbol{\Lambda}}}$. And Q-DMD modes are given by columns of $\dot{\boldsymbol{\Phi}}$:

$$\dot{\boldsymbol{\Phi}} = \dot{\mathbf{Y}}\dot{\mathbf{V}}\boldsymbol{\Sigma}^{-1}\tilde{\dot{\mathbf{W}}}. \tag{29}$$

These Q-DMD modes are the eigenvectors of $\hat{\dot{\mathbf{Q}}}$, since we have

$$\begin{aligned}\hat{\dot{\mathbf{Q}}}\dot{\boldsymbol{\Phi}} &= (\dot{\mathbf{Y}}\dot{\mathbf{V}}\boldsymbol{\Sigma}^{-1}\dot{\mathbf{U}}^{H})(\dot{\mathbf{Y}}\dot{\mathbf{V}}\boldsymbol{\Sigma}^{-1}\tilde{\dot{\mathbf{W}}})\\ &= \dot{\mathbf{Y}}\dot{\mathbf{V}}\boldsymbol{\Sigma}^{-1}(\dot{\mathbf{U}}^{H}\dot{\mathbf{Y}}\dot{\mathbf{V}}\boldsymbol{\Sigma}^{-1})\tilde{\dot{\mathbf{W}}}\\ &= \dot{\mathbf{Y}}\dot{\mathbf{V}}\boldsymbol{\Sigma}^{-1}\tilde{\dot{\mathbf{Q}}}\tilde{\dot{\mathbf{W}}}\\ &= \dot{\mathbf{Y}}\dot{\mathbf{V}}\boldsymbol{\Sigma}^{-1}\tilde{\dot{\mathbf{W}}}\tilde{\dot{\boldsymbol{\Lambda}}}\\ &= \dot{\boldsymbol{\Phi}}\tilde{\dot{\boldsymbol{\Lambda}}}.\end{aligned} \tag{30}$$

The approximate Q-DMD eigenvalues and Q-DMD modes are the standard eigenvalues and eigenvectors of quaternion matrix $\hat{\dot{\mathbf{Q}}}$ under certain conditions.

Theorem 5. *Define the quaternion matrix* $\dot{\boldsymbol{\Psi}}$ *as*

$$\dot{\boldsymbol{\Psi}} = \dot{\mathbf{U}}\tilde{\dot{\mathbf{W}}}, \tag{31}$$

where $\dot{\mathbf{U}}$, $\tilde{\dot{\mathbf{W}}}$ *are from Eqs.* (25) and (28)*, respectively. If the column space of* $\dot{\mathbf{Y}}$ *are spanned by columns of* $\dot{\mathbf{X}}$*, then the Q-DMD eigenvalues and Q-DMD modes defined by Eq.* (31) *are the standard eigenvalues and corresponding eigenvectors of quaternion matrix* $\hat{\dot{\mathbf{Q}}}$*.*

The proof of Theorem 5 can be found in the supplementary material. In the exact Q-DMD definition, Q-DMD modes (columns of $\dot{\boldsymbol{\Phi}}$) are calculated using $\dot{\boldsymbol{\Phi}} = \dot{\mathbf{Y}}\dot{\mathbf{V}}\boldsymbol{\Sigma}^{-1}\tilde{\dot{\mathbf{W}}}$ instead of $\dot{\boldsymbol{\Psi}} = \dot{\mathbf{U}}\tilde{\dot{\mathbf{W}}}$. In fact, if the column spaces of $\dot{\mathbf{X}}$ and $\dot{\mathbf{Y}}$ are the same, they tend to converge. The calculation steps of Q-DMD modes and eigenvalues are summarized in Algorithm 2.

Algorithm 2 Quaternion-based DMD algorithm (Q-DMD).

```
Input: Two matrices $\dot{\mathbf{X}}$ and $\dot{\mathbf{Y}}$ which are constructed from data.
Output: Q-DMD modes ($\dot{\boldsymbol{\Phi}}$) and Q-DMD eigenvalues.
1: $\dot{\mathbf{U}}$, $\boldsymbol{\Sigma}$, $\dot{\mathbf{V}}$ = QSVD($\dot{\mathbf{X}}, r$) $\longleftarrow$ (reduced) QSVD.
2: $\tilde{\dot{\mathbf{Q}}} \longleftarrow \dot{\mathbf{U}}^{H}\dot{\mathbf{Y}}\dot{\mathbf{V}}\boldsymbol{\Sigma}^{-1}$.
3: $\tilde{\dot{\mathbf{W}}}$, $\tilde{\dot{\boldsymbol{\Lambda}}}$ $\longleftarrow$ eigenvectors and standard eigenvalues of $\tilde{\dot{\mathbf{Q}}}$
4: $\dot{\boldsymbol{\Phi}} = \dot{\mathbf{Y}}\dot{\mathbf{V}}\boldsymbol{\Sigma}^{-1}\tilde{\dot{\mathbf{W}}}$ $\longleftarrow$ Q-DMD modes.
5: return $\dot{\boldsymbol{\Phi}}$ and $\tilde{\dot{\boldsymbol{\Lambda}}}$;
```

After calculating the low-rank approximations of eigenvalues (i.e., Q-DMD eigenvalues) and eigenvectors (i.e., Q-DMD modes) of the quaternion matrix $\hat{\dot{\mathbf{Q}}}$, the system state can be approximated by using the spectral decomposition:

$$\dot{\mathbf{x}}_{\text{Q-DMD}}(t_{v}) = \sum_{s=1}^{r}\dot{\boldsymbol{\phi}}_{s}\dot{\lambda}_{s}^{v-1}\dot{b}_{s} = \dot{\boldsymbol{\Phi}}\tilde{\dot{\boldsymbol{\Lambda}}}^{v-1}\dot{\mathbf{b}}, \tag{32}$$

where $t_v = v\Delta t$ ($v \in \mathbb{N}_{+}$), $\dot{\mathbf{b}} = \dot{\boldsymbol{\Phi}}^{\dagger}\dot{\mathbf{x}}(t_1)$ contains the initial amplitudes for the modes, and $\tilde{\dot{\boldsymbol{\Lambda}}} = \operatorname{diag}(\dot{\lambda}_1, \dot{\lambda}_2, \dots, \dot{\lambda}_r)$. The Q-DMD eigenvalues can be converted to the continuous form by using Definition 3 :

$$\dot{\omega}_{s} = \frac{\ln(\dot{\lambda}_{s})}{\Delta t}, \tag{33}$$

for $s = 1, 2, \dots, r$. Therefore, the right spectral decomposition above can be rewritten as

$$\dot{\mathbf{x}}_{\text{Q-DMD}}(t_{v}) = \sum_{s=1}^{r}\dot{\boldsymbol{\phi}}_{s}e^{\dot{\omega}_{s}(v-1)\Delta t}\dot{b}_{s} = \dot{\boldsymbol{\Phi}}\dot{\boldsymbol{\Omega}}^{(v-1)\Delta t}\dot{\mathbf{b}}, \tag{34}$$

where

$$\dot{\boldsymbol{\Omega}} = \begin{bmatrix} e^{\dot{\omega}_1} & 0 & \cdots & 0\\ 0 & e^{\dot{\omega}_2} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & e^{\dot{\omega}_r}\end{bmatrix}.$$

4.4. Background modeling using the quaternion-based dynamic mode decomposition method

The Q-DMD method can deal with a color video sequence because the frames of the video are normally uniformly sampled, and a color pixel can be represented by a quaternion. It is assumed here that the color video contains m frames, and each frame has n pixels in total, then each frame can be vectorized into a pure quaternion vector denoted as

$$\dot{\mathbf{x}}_{s} = R(:,s)\,i + G(:,s)\,j + B(:,s)\,k,$$

where $\dot{\mathbf{x}}_s \in \mathbb{H}^{n}$, and $s = 1, 2, \dots, m$. $R(:,s)$, $G(:,s)$, and $B(:,s) \in \mathbb{R}^{n}$ respectively represent the pixel values of the corresponding red, green and blue channels after vectorization of the sth frame. Then, all frames of the color video are vectorized and form a quaternion matrix $\dot{\mathbf{Z}} = \begin{bmatrix}\dot{\mathbf{x}}_1 & \dot{\mathbf{x}}_2 & \cdots & \dot{\mathbf{x}}_m\end{bmatrix} \in \mathbb{H}^{n\times m}$. And any given frame of the color video can be reconstructed by using Q-DMD method, that is by using Eq. (34). Given a color video sequence and assuming that frames are collected at uniform intervals, then the time points of these collected frames can form a vector $\mathbf{t} = \begin{bmatrix}t_1 & t_2 & \cdots & t_m\end{bmatrix}$. For convenience, here we assume $\mathbf{t} = \begin{bmatrix}0 & 1 & \cdots & m-1\end{bmatrix}$. Therefore, using the Q-DMD method, the full video sequence $\dot{\mathbf{X}}$ can be reconstructed by the following formula

$$\dot{\mathbf{X}}_{\text{Q-DMD}} = \sum_{s=1}^{r}\dot{\boldsymbol{\varphi}}_{s}e^{\dot{\omega}_{s}\mathbf{t}}\dot{b}_{s} = \dot{\boldsymbol{\Phi}}\dot{\boldsymbol{\Omega}}^{\mathbf{t}}\dot{\mathbf{b}}, \tag{35}$$

where $\dot{\boldsymbol{\varphi}}_s \in \mathbb{H}^{n}$, $\mathbf{t} \in \mathbb{R}^{1\times m}$. As we mentioned before, the parameter r is related to the dimensionality reduction and it is fixed to $m-1$, i.e., one less than the number of frames of the color video sequence in this paper. After obtaining the prediction of the whole video sequences by using Eq. (35), the separation of foregrounds and the background can be obtained by thresholding the low-frequency modes of the corresponding eigenvalues. Generally, any part of the first video frame that does not change with time, or changes very slowly in time satisfies $|\dot{\omega}_p| \approx 0$, and is regarded as the background. Therefore, suppose there is a $\dot{\omega}_p$ satisfying $|\dot{\omega}_p| \approx 0$, where $p \in \{1, 2, \dots, r\}$, and the others (i.e. $\dot{\omega}_s$, $\forall\, s \neq p$) are not near the origin. Then we can obtain that

$$\begin{aligned}\dot{\mathbf{X}}_{\text{Q-DMD}} &= \mathbf{L} + \mathbf{S}\\ &= \underbrace{\dot{\boldsymbol{\varphi}}_{p}e^{\dot{\omega}_{p}\mathbf{t}}\dot{b}_{p}}_{\text{Background}} + \underbrace{\sum_{s\neq p}\dot{\boldsymbol{\varphi}}_{s}e^{\dot{\omega}_{s}\mathbf{t}}\dot{b}_{s}}_{\text{Foreground}}.\end{aligned} \tag{36}$$

It is worth noting that quaternions do not satisfy the commutative law, so the product of the above Eq. (36) cannot change the order, and this is different from the real field generally, that is

$$\mathbf{X}_{\text{Q-DMD}} = \dot{\boldsymbol{\varphi}}_{p}e^{\dot{\omega}_{p}\mathbf{t}}\dot{b}_{p} + \sum_{s\neq p}\dot{\boldsymbol{\varphi}}_{s}e^{\dot{\omega}_{s}\mathbf{t}}\dot{b}_{s} \neq \dot{b}_{p}\dot{\boldsymbol{\varphi}}_{p}e^{\dot{\omega}_{p}\mathbf{t}} + \sum_{s\neq p}\dot{b}_{s}\dot{\boldsymbol{\varphi}}_{s}e^{\dot{\omega}_{s}\mathbf{t}}.$$

There exists another difference between DMD and Q-DMD. Each term of the DMD reconstruction is a complex matrix, i.e., $b_s\boldsymbol{\varphi}_s e^{\omega_s\mathbf{t}} \in \mathbb{C}^{n\times m}$. However, when the DMD term is decomposed into approximate low-rank and sparse components, real value output is required. Different from the output of DMD, each term of the Q-DMD reconstruction is a quaternion matrix, i.e., $\dot{\boldsymbol{\varphi}}_s e^{\dot{\omega}_s\mathbf{t}}\dot{b}_s \in \mathbb{H}^{n\times m}$, and the coefficient matrices corresponding to the three imaginary units i, j, k are real matrix and correspond to the three channels RGB of the color frame to be reconstructed.

5. Experimental results and discussion

In this section, we conduct numerical experiments on two publicly available benchmark datasets, i.e., Scene background modeling dataset (SBMnet dataset)[1] (Jodoin et al., 2017) and Scene background initialization dataset (SBI dataset)[2] (Maddalena and Petrosino, 2015; Bouwmans et al., 2017a), and illustrate that our Q-DMD method has the ability to separate color video sequences into background models and foreground components. We compare our method with the original DMD and several state-of-the-art approaches.

Quantitative assessment: To evaluate the performance of our proposed method, we employ the following six metrics suggested by the SBMnet dataset and the SBI dataset to measure the performance. For a detailed explanation for each metric we recommend Maddalena and Petrosino (2015) and Jodoin et al. (2017).

(1) Average gray-level error (AGE): It is the average of the absolute difference between the gray-level images' ground truth (GT) and the computed background (CB)

$$\mathrm{AGE} = \frac{1}{N}\sum_{z=0}^{N-1} |B_{gt_z} - B_{cbz}|, \tag{37}$$

where N is the total number of pixels in the image and B_{gt} and B_{cb} are the Y channel in the YCbCr color space of the GT image and the background image computed by a background initialization method, respectively. And the Y channel in the YCbCr color space is obtained by

$$\mathrm{Y} = 0.299 \times \mathrm{R} + 0.587 \times \mathrm{G} + 0.114 \times \mathrm{B}. \tag{38}$$

According to Eq. (37), the lower the AGE value is, the better the background estimate is.

(2) Percentage of error pixels (pEPs): It is the percentage of error pixels (number of pixels in CB whose value differs from the value of the corresponding pixel in GT by more than a threshold $\tau = 20$) with respect to the total number of pixels in the image.

$$\mathrm{pEPs} = \frac{1}{N}\sum_{z=0}^{N-1} \left(1_{|B_{gt_z} - B_{cbz}| > \tau} \right), \tag{39}$$

where 1 is an indicator function. The lower the pEPs value, the better is the background estimate.

(3) Percentage of clustered error pixels (pCEPS): It is the percentage of clustered error pixels (number of pixels whose 4-connected neighbors are also error pixels) with respect to the total number of pixels in the image. The lower the pCEPs value, the better is the background estimate.

(4) MultiScale Structural Similarity Index (MSSSIM): Estimate the perceived visual distortion. For a single scale, the SSIM of a squared image block $\mathbf{x}_{cb}$ of B_{cb} and the corresponding image block $\mathbf{x}_{gt}$ of B_{gt} is calculated as:

$$\mathrm{SSIM}(\mathbf{x}_{cb}, \mathbf{x}_{gt}) = \frac{(2\mu_{gt}\mu_{cb}+C_1)(2\sigma_{gt,cb}+C_2)}{(\mu_{gt}^2+\mu_{cb}^2+C_1)(\sigma_{gt}^2+\sigma_{cb}^2+C_2)}, \tag{40}$$

where μ_y and σ_y are the mean and the variance of $\mathbf{x}_y$, $y \in \{cb, gt\}$, respectively, $\sigma_{gt,cb}$ is the covariance of $\mathbf{x}_{cb}$ and $\mathbf{x}_{gt}$, $C_1 = K_1 L$, $C_2 = K_2 L$, $K_1 = 0.01$, $K_2 = 0.03$, and L is the range of pixel values. The SSIM of the whole images is computed as the mean of the values obtained for all corresponding image blocks. MSSSIM aggregates SSIM values computed at different image scales, thus providing hints on the similarity of the GT and the evaluated background images at both the global and the detail level. MS-SSIM is translation invariant. It assumes values in $[-1, 1]$; the higher the value of MSSSIM, the better the estimated background.

(5) Peak-signal-to-noise-ratio (PSNR): PSNR is defined as

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{\mathrm{MAX}^2}{\mathrm{MSE}}\right), \tag{41}$$

where MAX is 255 in this case and MSE is the mean squared error between B_{gt} and B_{cb},

$$\mathrm{MSE} = \frac{1}{N}\sum_{z=0}^{N-1}\left(B_{gt_z} - B_{cbz}\right)^2. \tag{42}$$

It assumes values in decibels (db); the higher the PSNR value, the better is the background estimate.

(6) Color image quality measure (CQM): By first converting RGB images to the YUV color space and then computing the PSNR value of each YUV channel separately, CQM is calculated based on the resulting PSNR values:

$$\mathrm{CQM} = \mathrm{PSNR}_Y \times R_W + \frac{\mathrm{PSNR}_U + \mathrm{PSNR}_V}{2} \times C_W, \tag{43}$$

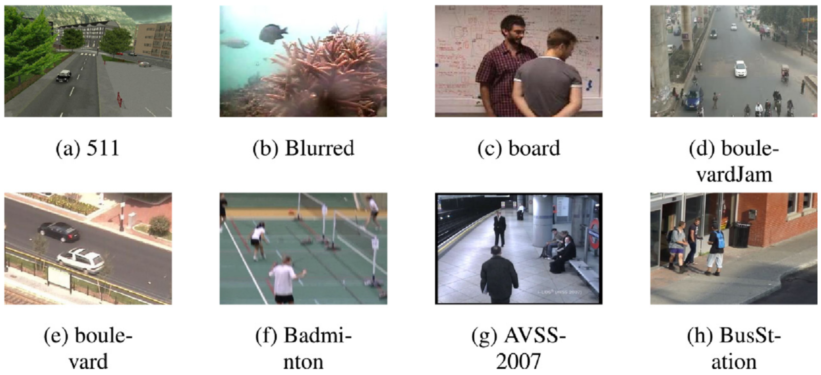
where R_W and C_W are biologically-inspired coefficients set to 0.9449 and 0.0551, respectively (For specific details, please refer to the original paper Yalman and ERTÜRK, 2013). It assumes values in decibels and the higher the CQM value, the better the background estimate.

Experiment on SBMnet dataset: We first simulated on SBMnet dataset to evaluate the proposed Q-DMD method for color video background modeling. The dataset consists of a wide range of challenging videos, such as camera jitter, intermittent object motion, background motion, abandoned object, illumination changes, long and short sequences of images, and the ground truth of some videos was provided. Representative frames of the eight videos from the SBMnet dataset are shown in Fig. 1. The spatial resolutions of those videos vary from 240 × 240 to 800 × 600. The spatial resolutions of the videos "511" and "AVSS2007" are 480 × 640 and 576 × 720, respectively, so the two videos were down-sampled by a factor of 2 to make the computational memory requirements manageable for personal computers. In order to process a color video efficiently, for the color video whose total number of frames in the SBMnet database is far more than 200 frames, we randomly selected 200 consecutive frames containing the foreground and background information for experiments and the frames that did not change over time were trimmed. We compared the performance of our method with the DMD, and several other existing state-of-the-art approaches, including MSCL (Javed et al., 2017), FSBE (Djerida et al., 2019), LaBGen (Laugraud et al., 2017), NExBI (Mseddi et al., 2019), Photomontage (Agarwala et al., 2004), SC-SOBS-C4 (Maddalena and Petrosino, 2016), BE-AAPSA (Ramirez-Alonso et al., 2017), ABM (Avola et al., 2017), and FC-FlowNet (Halfaoui et al., 2016).

The results of the generated backgrounds by different methods on SBMnet dataset are displayed in Fig. 2. For the three sequences "511", "Blurred", and "boulevard" with long background exposure time and mild object motion, our method generates clear backgrounds with satisfied perceptual quality like other methods except for BE-AAPSA on "Blurred". However, for those videos ("board" and "boulevardJam") whose background is visible for a short time, the generated backgrounds by Q-DMD contain the shadow effect of moving objects. And for "board", except for NExBI, FSBE, and MSCL, all the other methods fail to initialize a high-quality background. For "boulevardJam", compared with other methods other than FSBE, our method generates a relatively better result. There exists camera jittering in the sequences "badminton", and the instability (e.g., vibration) of the camera and the movement of the foreground will cause the generated background to be interfered with by the two movements. Although the background of sequences "badminton" generated by Q-DMD has also been corrupted by blurring effects, the result is still better than most other methods. "AVSS2007" and "BusStation" belong to the intermittent motion category. The challenge of this category is that

[1] http://www.SceneBackgroundModeling.net

[2] http://sbmi2015.na.icar.cnr.it/SBIdataset.html



(a) 511

(b) Blurred

(c) board

(d) boulevardJam

(e) boulevard

(f) Badminton

(g) AVSS2007

(h) BusStation

Fig. 1. Representative frames of the eight videos from SBMnet dataset.

Table 1
Quantitative quality indexes of DMD method and the proposed Q-DMD method for the 8 color videos on SBMnet dataset.

Videos	AGE		pEPs		pCEPs		MSSSIM		PSNR		CQM	
	DMD	Q-DMD	DMD	Q-DMD	DMD	Q-DMD	DMD	Q-DMD	DMD	Q-DMD	DMD	Q-DMD
511	4.0569	**4.0425**	0.0378	**0.0373**	0.0013	**0.0013**	0.9780	**0.9783**	30.2008	**30.2637**	31.8512	**32.1079**
Blurred	11.3377	**9.4219**	0.1512	**0.1033**	0.1233	**0.0821**	0.9561	**0.9596**	24.3609	**25.5840**	25.3249	**26.4187**
board	30.1856	**28.5044**	0.6962	**0.4943**	0.6066	**0.4073**	0.5112	**0.5549**	17.4713	16.6794	18.3299	18.2303
boulevardJam	4.1226	**3.6353**	0.0095	**0.0095**	0.0029	0.0031	0.9314	**0.9325**	31.4713	**31.5366**	32.5999	**32.6002**
boulevard	9.4585	**9.4098**	0.1375	**0.1372**	0.0277	**0.0276**	0.9116	**0.9116**	22.6940	**22.7186**	24.1742	**24.2013**
Badminton	5.6188	**3.9600**	0.0283	**0.0257**	0.0132	**0.0115**	0.9541	**0.9570**	30.2312	**31.3778**	31.0495	**32.2622**
AVSS2007	17.2716	17.3175	0.2610	**0.2589**	0.2065	**0.2031**	0.7127	0.7123	19.3044	**19.3450**	20.2489	**20.2835**
BusStation	6.4834	**6.3356**	0.0497	0.0611	0.0255	0.0328	0.9512	0.9505	28.3237	28.0328	29.2942	29.1266

the foreground moves then stops for a short while, and then starts moving again, which causes "ghosting" artifacts in the reconstructed background. For "AVSS2007", almost all the methods fail to generate a high-quality background. The background generated by Q-DMD is better than that generated by BE-AAPSA and Photomontage. For the sequence "BusStation", the performance of Q-DMD is better than that of ABM, SC-SOBS-C4, Photomontage, and LaBGen.

The quantitatively evaluated results are shown in Tables 1 and 2. Six evaluation metrics (AGE, pEPs, pCEPs, MSSSIM, PSNR, and CQM) are used to demonstrate the performance of the proposed Q-DMD method. It can be found from Table 1 that the performance of our Q-DMD method is better than that of the DMD method in the vast majority of images. The CQM is a color image quality measure based on reversible luminance and chrominance (YUV) color transformation and PSNR measure (Yalman and ERTÜRK, 2013). Therefore, in addition to the advantage on PSNR, the good performance of our method on CQM also demonstrates the advantages of the quaternion-based model.

As can be seen from Table 2, our method achieves competitive CQM values in four color videos ("511", "boulevardJam", "boulevard", and "Badminton") compared with other methods. For those videos with very short background exposure duration, such as "Board", "AVSS2007", "boulevardJam", and "Badminton", as shown in Fig. 2, there exists the shadow effect of moving objects in the generated background, which leads to a reduction in performance. However, for long sequence images in which the background appears for a long time, such as "Basic (511)", "Jitter (boulevard)", the CQM value obtained by our method is almost the same as that obtained by FC-FlowNet method. In particular, our method achieved the second highest CQM value on "Clutter (boulevardJam)" and "Jitter (boulevard)". These results show that compared with the state-of-the-art approaches, the Q-DMD method is competitive enough to extract background from challenging videos.

Experiment on the SBI dataset: We conducted additional experiments on the SBI dataset to evaluate our background modeling results by comparing them with the background initialization results obtained by the DMD method. The SBI dataset also contains a large amount of data extracted from original publicly available sequences, which are frequently used in the literature to evaluate background initialization algorithms (Maddalena and Petrosino, 2015). We evaluated our approach on 9 sequences of the SBI dataset. The remaining videos, "Cavignal" and "CAVIAR1" are objects with intermittent object motion which are not defined as moving objects, "Snellen" and "PeopleAndFoliage" are those videos with very short background exposure duration, and "Toscana" has only five frames and is not suitable for Q-DMD method. The background frames that did not change over time were trimmed and among the nine videos, except those with less than 200 frames, we extracted 200 frames of videos for experiments to reduce the computing time. We calculated the six metrics (AGE, pEPs, pCEPs, MSSSIM, PSNR, and CQM) suggested by the SBI dataset to measure the reconstructed background models, and the results are shown in Table 3. Generated backgrounds with Q-DMD method on SBI dataset are shown in Fig. 3.

As can be seen from Fig. 3, for the three color videos, "Board", "HighwayI", and "HighwayII", the backgrounds reconstructed by the DMD method had obvious different color intensity compared to the ground truth, however the background models generated by Q-DMD method do not have this problem. Therefore, we have reason to believe that this is mainly due to the advantage of quaternion in representing color pixel values. As shown in Table 3, for most videos on the SBI dataset, our AGE, pEPs, and pCEPs were lower than those of the DMD method, which indicates that it has a lower pixel-wise difference between the reconstructed background model and the ground truth model. PSNR, MS-SSIM, and CQM also show that the Q-DMD method has advantages over the DMD method.

In addition, we also compare our method with several deep neural network-based methods, which are BI-GAN (Sultana et al., 2020), KNN (Zivkovic and Van Der Heijden, 2006), and BE-AAPSA (Ramirez-Alonso et al., 2017). The comparison results are shown in Table 4. We used the new version of MATLAB tools available on the SBI website to calculate the quantitative AGE, pEPs, pCEPs, MSSSIM, PSNR, and CQM values of DMD and Q-DMD. However, when comparing with BI-GAN, KNN, and BE-AAPSA, there is no comparison of CQM values, because the CQM values of those methods to be compared were calculated using the old version of Matlab tool, that included a bug, as indicated in the SBI website. As can be seen from Table 4, Q-DMD performed better than other methods for the video sequence "CAVIAR2". No one method always performs best. In most cases, BE-AAPSA achieved

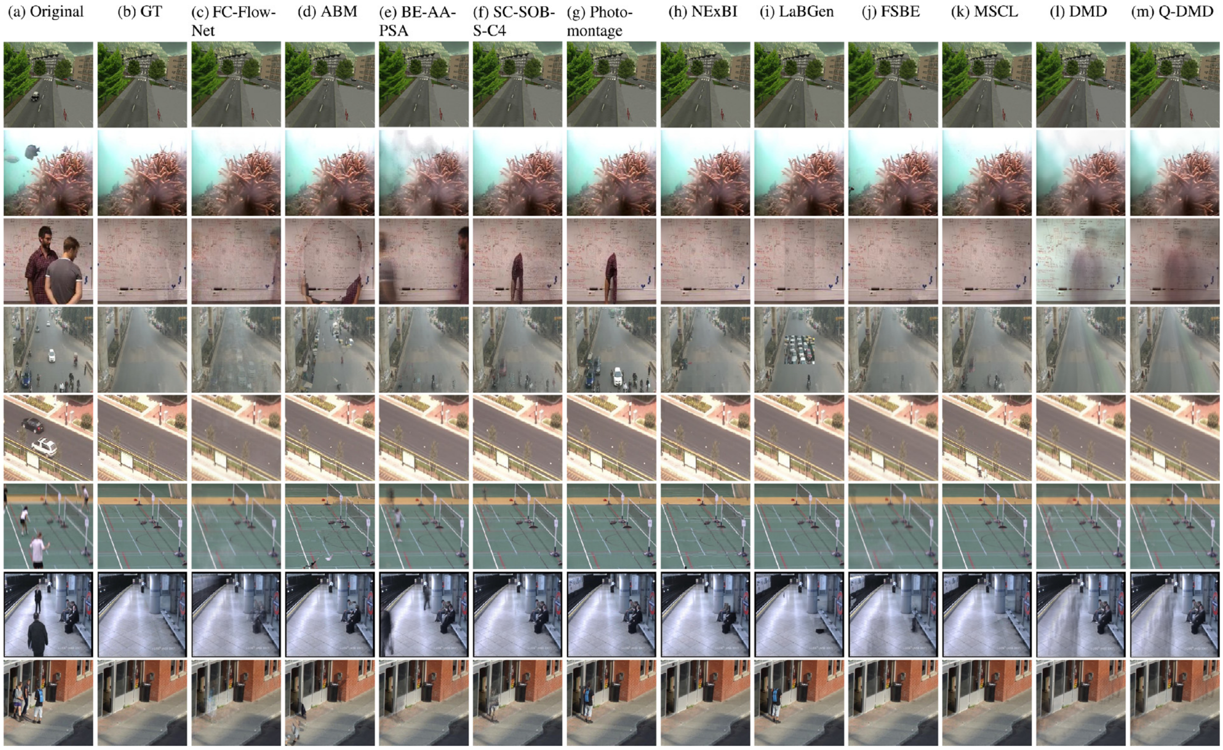


Fig. 2. (a) Representative original frames of the eight videos from the SBMnet dataset. (b) Ground truth of the background. (c)–(m) are the generated backgrounds results of FC-FlowNet, ABM, BE-AAPSA, SC-SOBS-C4, Photomontage, NExBI, LaBGen, FSBE, MSCL, DMD, Q-DMD, respectively. From top to bottom: First-row: 511; Second-row: Blurred; Third-row: board; Fourth-row: boulevardJam; Fifth-row: boulevard; Sixth-row: Badminton; Seventh-row: AVSS-2007; Last-row: BusStation.

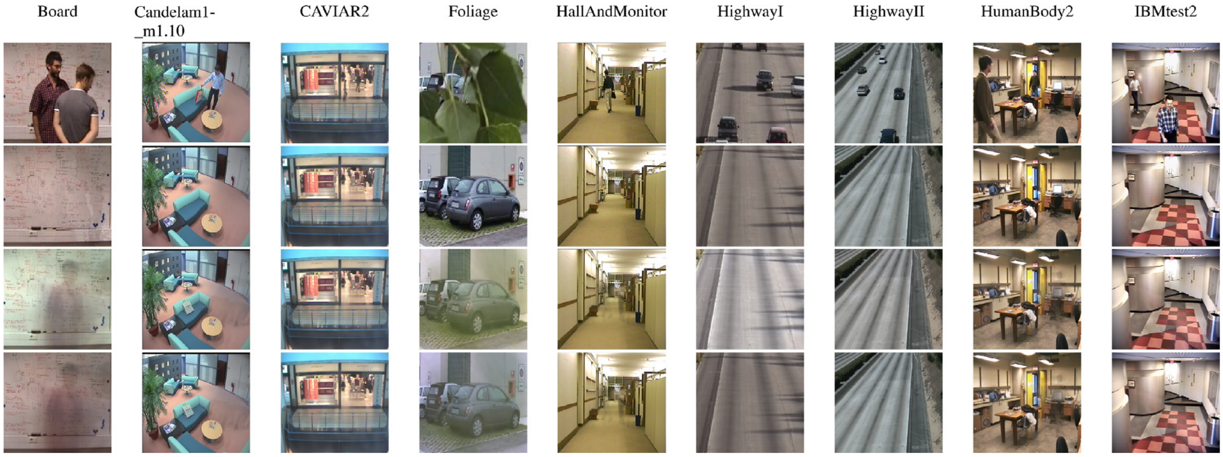


Fig. 3. First-row: Representative frames of the nine videos from the SBI dataset; second-row: Ground truth of the background; Third-row: Generated backgrounds with DMD method; Last-row: Generated backgrounds using Q-DMD method.

the highest AGE of all comparison methods, while Q-DMD achieved the second highest AGE. For the video sequences "HallAndMonitor" and "HumanBody2", on some numerical indicators, BE-AAPSA and BI-GAN performed better than Q-DMD, while KNN performed poorly in terms of almost all metrics. For the video sequences "HighwayI" and "HighwayII", Q-DMD and KNN outperformed BE-AAPSA in almost all metrics, except AGE and PSNR (see Table 4), and Q-DMD achieved the highest MSSSIM among all the compared methods. The reason behind this is that in these sequences, the foreground objects always move (e.g., HighwayI, HighwayII, and CAVIAR2), which fits the assumptions of Q-DMD, and therefore, Q-DMD performs better in these categories. Overall, Q-DMD is competitive compared to these methods.

Results analysis of Q-DMD method: The Q-DMD method only considers the part of the image sequence that does not change with time as the background model. When this condition is met, the Q-DMD can achieve better performance. For example, for the three sequences of SBI dataset, "HighwayI", "HighwayII", and "IBMtest2", the foregrounds (people or vehicles) do not remain stationary anywhere in the scene throughout the sequence. Therefore, for these color video sequences, Q-DMD can generate a superb background model by eliminating the foreground part of continuous motion. The results of these three videos, "511", "Blurred", and "boulevard" in SBMnet dataset also reflect this phenomenon. However, when this condition is not satisfied, it will affect the results of Q-DMD. For example, on the " Board" sequence of SBI dataset, the background models reconstructed by Q-DMD and DMD had human shadow. This is because there are two men in the video sequence who occupy a large proportion of the background in the whole video sequence, and the man standing on the right side of

Table 2
Comparison of AGE, MSSSIM, PSNR, and CQM between proposed Q-DMD method and other state-of-the-art background initialization methods on SBMnet dataset. Numbers that are bold and underlined, numbers that are bold and dashed, and numbers that are bold represent the highest, second and third highest CQM values on each video, respectively.

Videos		FC-Flow-Net	ABM	BE-AA-PSA	SC-SOB-S-C4	Photo-montage	NExBI	LaBGen	FSBE	MSCL	Q-DMD
Basic (511)	AGE	3.9735	6.1224	4.0511	5.5782	5.7977	5.8916	4.8294	3.7414	4.2186	4.0425
	MSSSIM	0.9735	0.9407	0.9744	0.9553	0.9488	0.9345	0.9475	0.9761	0.9703	0.9783
	PSNR	30.8573	26.5925	30.0319	27.2563	26.6706	26.2599	27.6577	30.5804	30.0808	30.2637
	CQM	**32.5541**	28.5767	31.8292	29.2562	28.7131	28.3762	29.5002	**32.2388**	31.8784	**32.1079**
Basic (Blurred)	AGE	2.6962	2.4574	15.2057	2.0878	2.0214	2.5863	1.3990	3.1953	1.8057	9.4219
	MSSSIM	0.9902	0.9906	0.8924	0.9934	0.9941	0.9909	0.9975	0.9882	0.9930	0.9596
	PSNR	36.3751	36.5219	22.4556	37.8907	38.2473	36.3266	41.5779	31.8882	38.1747	25.5840
	CQM	36.8199	36.8815	23.3364	38.2289	**38.5613**	36.6845	**41.6541**	32.4592	**38.5264**	26.4187
Clutter (board)	AGE	14.1523	13.9757	25.4532	13.2993	13.4739	6.7738	8.0208	5.5795	6.0836	28.5044
	MSSSIM	0.8691	0.7647	0.7629	0.5086	0.5029	0.9162	0.8491	0.9340	0.9322	0.5549
	PSNR	22.1587	20.3477	15.6631	19.0239	18.8444	28.1156	27.4114	29.7845	29.2266	16.6794
	CQM	23.2484	21.4077	16.9305	20.3248	20.0911	**29.0466**	28.3713	**30.7618**	**30.0739**	18.2303
Clutter (boulevardJam)	AGE	5.0200	7.6215	5.1418	6.7592	12.1045	5.0516	8.2239	2.3321	5.0010	3.6353
	MSSSIM	0.8619	0.7546	0.9219	0.8752	0.7604	0.8789	0.6851	0.9653	0.9100	0.9325
	PSNR	30.3476	24.7166	28.9114	26.8135	20.9163	27.6165	22.6515	33.8660	28.5787	31.5366
	CQM	**31.4309**	26.0921	30.0986	28.0968	22.1436	28.8454	23.9772	**35.0117**	29.8099	**32.6002**
Jitter (boulevard)	AGE	10.6830	11.4377	10.8262	9.7795	9.7829	9.4182	10.1888	10.1060	5.8660	9.4098
	MSSSIM	0.8956	0.8776	0.8821	0.8998	0.8995	0.9076	0.8946	0.9003	0.9699	0.9116
	PSNR	22.5246	20.6235	21.1393	21.7477	21.6868	22.2455	21.4645	22.5280	26.0077	22.7186
	CQM	**24.0208**	22.0462	22.5861	23.1910	23.0513	23.7767	22.9249	23.8107	**27.1642**	**24.2013**
Jitter (Badminton)	AGE	5.5368	7.7833	4.3975	3.6988	4.2924	5.2289	2.2670	6.5668	2.4174	3.9600
	MSSSIM	0.9367	0.7872	0.9204	0.9386	0.9237	0.8726	0.9805	0.8636	0.9729	0.9570
	PSNR	29.9097	23.9886	29.1352	30.8739	29.6868	26.7733	34.6482	27.3765	33.8911	31.3778
	CQM	30.7442	24.9043	29.9490	31.8162	30.4911	27.7054	**35.2688**	28.2185	**34.5949**	**32.2622**
IntermittentMotion (AVSS2007)	AGE	11.6751	9.6720	20.6172	10.8835	12.0167	12.3242	8.3062	11.5900	7.5256	17.3175
	MSSSIM	0.8726	0.8809	0.7929	0.8625	0.8400	0.8799	0.9050	0.8830	0.9294	0.7123
	PSNR	20.7442	21.6609	16.4960	21.3609	19.2860	21.1518	21.4577	20.1106	22.3138	19.3450
	CQM	21.7565	**22.5576**	17.5546	22.2359	20.2173	22.0076	**22.3158**	21.2110	**23.0990**	20.2835
IntermittentMotion (BusStation)	AGE	4.3513	6.5147	4.5206	5.4672	6.5309	3.0622	7.0296	4.3997	3.4057	6.3356
	MSSSIM	0.9622	0.9128	0.9621	0.9212	0.8872	0.9815	0.8889	0.9847	0.9821	0.9505
	PSNR	31.1049	24.4467	30.0286	25.1635	21.8651	35.2212	22.0988	33.1076	34.2369	28.0328
	CQM	31.7573	25.5588	30.9833	26.3344	22.8979	**35.7016**	23.0664	**33.7125**	**34.8402**	29.1266

Table 3
Evaluation results (SBI dataset).

Videos	AGE		pEPs		pCEPs		MSSSIM		PSNR		CQM	
	DMD	Q-DMD	DMD	Q-DMD	DMD	Q-DMD	DMD	Q-DMD	DMD	Q-DMD	DMD	Q-DMD
Board	28.3809	**24.5430**	67.8445	**43.4299**	59.6707	**36.6799**	0.5406	**0.5967**	18.0398	17.9744	17.7580	**18.1454**
Candelam1_m1.10	3.1744	**3.1249**	1.3780	1.4076	0.7339	0.7576	0.9653	**0.9654**	31.6111	**31.6436**	30.8939	30.8877
CAVIAR2	1.0730	**1.0687**	0.0041	**0.0010**	0.0000	**0.0000**	0.9988	**0.9988**	43.3340	**43.7556**	42.5350	**42.9473**
Foliage	22.2094	**20.8727**	43.2361	48.8854	31.1806	34.1458	0.7975	0.7644	19.2886	**20.0956**	19.2072	**19.6507**
HallAndMonitor	3.7244	3.8173	2.9616	**2.9096**	1.6359	**1.6098**	0.9561	**0.9561**	29.9361	**29.9827**	29.9230	**29.9749**
HighwayI	43.2153	**5.4775**	99.9688	**0.1198**	99.8958	**0.0039**	0.9015	**0.9649**	15.2265	**31.3369**	15.1443	**31.2429**
HighwayII	3.1864	**2.7014**	0.3164	**0.3125**	0.0000	0.0013	0.9911	**0.9917**	35.1937	**36.1269**	34.8043	**35.6207**
HumanBody2	7.1215	**7.0930**	6.6536	6.7174	3.9896	4.0156	0.9511	**0.9515**	25.9354	**25.9580**	25.5177	25.4730
IBMtest2	4.9182	**4.8208**	2.7122	**2.2513**	0.8477	**0.6133**	0.9826	**0.9827**	30.7902	**30.9653**	29.9444	**30.0359**

the sequence has been rotating for some time in the sequence, which leads to the background covered by the foregrounds for a long time.

In terms of quantitative metrics, CQM values reflect the advantages of our method compared with DMD, and the CQM considers the biological characteristics of the human eye, which is an enhanced version of PSNR. In addition, from a visual point of view, the background images of “Board”, “HighwayI”, “HighwayII”, and “IBMtest2” sequences in the SBI dataset reconstructed by DMD show that processing three color channels separately may lead to color distortion, and Q-DMD can overcome this problem. The reason behind all this is that Q-DMD naturally handles the coupling between the three color channels, and makes full use of the information of color images.

6. Conclusions

In this paper, we propose a quaternion-based DMD (Q-DMD) method for color video background modeling using quaternion matrix analysis. Quaternion representation treats color pixels as vector units rather than scalars, naturally processes the coupling between color channels, and fully retains the color information of the color image or video. A high-order real tensor can be used to present a color video, however, the color structure will be destroyed in the process of metrication (e.g., *mode-k unfolding*). Using the standard eigenvalue of quaternion, we establish the spectral decomposition of the quaternion matrix, and then extend DMD to the quaternion system, i.e., Q-DMD. The results demonstrate that compared with DMD, our method shows advantages in reconstructing color video background model, and compared with several state-of-art methods, the proposed method still has competitive performance (w.r.t., CQM). Note that the proposed method can reconstruct a better background model for videos that meet the requirements (i.e., the Q-DMD method only considers the parts of the color videos that do not change with time as the background model). On the contrary, for videos that do not meet the conditions, the effect of the reconstructed background model will be reduced.

Table 4
Comparison of AGE, pEPs, pCEPs, MSSSIM, and PSNR between proposed Q-DMD method and other state-of-the-art background initialization methods on SBI dataset.

Videos		AGE	pEPs	pCEPs	MSSSIM	PSNR
Candelam1_m1.10	BI-GAN	4.9075	0.9033	0.0488	**0.9800**	31.3445
	KNN	11.2176	10.0507	5.817	0.8158	17.3467
	BE-AAPSA	**2.2656**	**0.0116**	**0.0065**	0.9733	**31.9643**
	Q-DMD	3.1249	1.4076	0.7576	0.9654	31.6436
CAVIAR2	BI-GAN	12.7988	13.6963	2.0020	0.9809	24.8151
	KNN	7.1935	4.8910	1.4404	0.8469	18.9970
	BE-AAPSA	1.1718	**0.0000**	**0.0000**	0.9983	43.7194
	Q-DMD	**1.0687**	0.0010	**0.0000**	**0.9988**	**43.7556**
HallAndMonitor	BI-GAN	3.9255	0.1709	**0.0000**	**0.9899**	34.1297
	KNN	3.9413	**0.0121**	0.0021	0.9519	28.2208
	BE-AAPSA	**2.4425**	0.3200	0.0400	0.9892	**36.4218**
	Q-DMD	3.8173	2.9096	1.6098	0.9561	29.9827
HumanBody2	BI-GAN	7.1523	4.1748	0.2930	**0.9744**	**28.7653**
	KNN	20.9423	18.5130	15.2188	0.7783	14.5871
	BE-AAPSA	**6.3274**	**0.0797**	**0.0550**	0.9528	24.9434
	Q-DMD	7.0930	6.7174	4.0156	0.9515	25.9580
HighwayI	BI-GAN	–	–	–	–	–
	KNN	6.1277	**0.0616**	**0.0003**	0.8506	25.1521
	BE-AAPSA	**4.3721**	2.7600	0.6900	0.9442	31.1332
	Q-DMD	5.4775	0.1198	0.0039	**0.9649**	**31.3369**
HighwayII	BI-GAN	–	–	–	–	–
	KNN	3.2112	**0.0085**	**0.0001**	0.9851	32.0981
	BE-AAPSA	**2.5181**	0.2800	0.0100	0.9903	**36.2738**
	Q-DMD	2.7014	0.3125	0.0013	**0.9917**	36.1269

Therefore, in the future, we will work on proposing improved versions of the Q-DMD method to achieve better performance. For example, Kutz et al. (2015, 2016) has demonstrated that a multi-resolution dynamic mode decomposition approach (multi-resolution DMD) combining DMD with multi-resolution analysis can robustly separate complex systems into multi-resolution time-scale component hierarchies, which is a good hint for us to improve Q-DMD in the future. We will also consider introducing nonlinearity to make the model applicable to a wider range of data types. Additionally, Alford-Lago et al. (2022) proposed a deep learning enhanced DMD algorithm, which inspired us to develop a Q-DMD-based learning algorithm in the future. Finally, inspired by quaternion-based higher-order singular value decomposition (QHOSVD) (Miao and Kou, 2021b), we plan to further construct a quaternion-based higher-order dynamic mode decomposition theory based on the Q-DMD method by combining tensors and quaternions to better handle the task of Scene Background Initialization or other color video processing problems.

CRediT authorship contribution statement

Juan Han: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing. **Kit Ian Kou:** Conceptualization, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Jifei Miao:** Methodology, 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 University of Macau (MYRG2019-00039-FST), Science and Technology Development Fund, Macao S.A.R (FDCT/0036/2021/AGJ).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.cviu.2022.103560.

References

Agarwala, A., Dontcheva, M., Agrawala, M., Drucker, S., Colburn, A., Curless, B., Salesin, D., Cohen, M., 2004. Interactive digital photomontage. ACM Trans. Graph. 23 (3), 294–302. http://dx.doi.org/10.1145/1015706.1015718.

Alford-Lago, D.J., Curtis, C.W., Ihler, A.T., Issan, O., 2022. Deep learning enhanced dynamic mode decomposition. Chaos 32 (3), 033116. http://dx.doi.org/10.1063/5.0073893.

Avola, D., Bernardi, M., Cinque, L., Foresti, G.L., Massaroni, C., 2017. Adaptive bootstrapping management by keypoint clustering for background initialization. Pattern Recognit. Lett. 100, 110–116. http://dx.doi.org/10.1016/j.patrec.2017.10.029.

Bautista, C.M., Dy, C.A., Mañalac, M.I., Orbe, R.A., Cordel, M., 2016. Convolutional neural network for vehicle detection in low resolution traffic videos. In: 2016 IEEE Region 10 Symposium. TENSYMP, pp. 277–281.

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. 117, 8–66. http://dx.doi.org/10.1016/j.neunet.2019.04.024.

Bouwmans, T., Maddalena, L., Petrosino, A., 2017a. Scene background initialization: A taxonomy. Pattern Recognit. Lett. 96, 3–11. http://dx.doi.org/10.1016/j.patrec.2016.12.024.

Bouwmans, T., Sobral, A., Javed, S., Jung, S.K., Zahzah, E.-H., 2017b. Decomposition into low-rank plus additive matrices for background/foreground separation: A review for a comparative evaluation with a large-scale dataset. Comput. Sci. Rev. 23, 1–71. http://dx.doi.org/10.1016/j.cosrev.2016.11.001.

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.

Brunton, S.L., Kutz, J.N., 2019. Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control. Cambridge University Press.

Candès, E.J., Li, X., Ma, Y., Wright, J., 2011. Robust principal component analysis? J. ACM 58 (3), 1–37. http://dx.doi.org/10.1145/1970392.1970395.

Chang, J.-H., Ding, J.-J., et al., 2003. Quaternion matrix singular value decomposition and its applications for color image processing. In: Proceedings 2003 International Conference on Image Processing (Cat. No. 03CH37429), vol. 1, pp. I–805.

Cinelli, L.P., 2017. Anomaly Detection in Surveillance Videos Using Deep Residual Networks (Ph.D. thesis). Ph.D. thesis, Universidade Federal do Rio de Janeiro.

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. http://dx.doi.org/10.1016/j.image.2019.06.001.

Ell, T.A., Le Bihan, N., Sangwine, S.J., 2014. Quaternion Fourier Transforms for Signal and Image Processing. John Wiley & Sons.

Ell, T.A., Sangwine, S.J., 2007. Hypercomplex Fourier transforms of color images. IEEE Trans. Image Process. 16 (1), 22–35. http://dx.doi.org/10.1109/TIP.2006.884955.

Erichson, N.B., Brunton, S.L., Kutz, J.N., 2019. Compressed dynamic mode decomposition for background modeling. J. Real Time Image Process 16 (5), 1479–1492. http://dx.doi.org/10.1007/s11554-016-0655-2.

Girard, P.R., 2007. Quaternions, Clifford Algebras and Relativistic Physics. Springer Science & Business Media.

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

Hamilton, W.R., 1844. II. On quaternions; or on a new system of imaginaries in algebra. London, Edinburgh, Dublin Philosoph. Mag. J. Sci. 25 (163), 10–13. http://dx.doi.org/10.1080/14786444408644923.

Hosny, K.M., Darwish, M.M., 2019. New set of multi-channel orthogonal moments for color image representation and recognition. Pattern Recognit. 88, 153–173. http://dx.doi.org/10.1016/j.patcog.2018.11.014.

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 (2), 1007–1022. http://dx.doi.org/10.1109/TIP.2018.2874289.

Javed, S., Mahmood, A., Bouwmans, T., Jung, S.K., 2017. Background–foreground modeling based on spatiotemporal sparse subspace clustering. IEEE Trans. Image Process. 26 (12), 5840–5854. http://dx.doi.org/10.1109/TIP.2017.2746268.

Jodoin, P.-M., Maddalena, L., Petrosino, A., Wang, Y., 2017. Extensive benchmark and survey of modeling methods for scene background initialization. IEEE Trans. Image Process. 26 (11), 5244–5256. http://dx.doi.org/10.1109/TIP.2017.2728181.

Kutz, J.N., Fu, X., Brunton, S.L., 2016. Multiresolution dynamic mode decomposition. SIAM J. Appl. Dyn. Syst. 15 (2), 713–735. http://dx.doi.org/10.1137/15M1023543.

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: 2015 IEEE International Conference on Computer Vision Workshop. ICCVW, pp. 921–929. http://dx.doi.org/10.1109/ICCVW.2015.122.

Kyrchei, I., 2013. Explicit representation formulas for the minimum norm least squares solutions of some quaternion matrix equations. Linear Algebra Appl. 438 (1), 136–152. http://dx.doi.org/10.1016/j.laa.2012.07.049.

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. http://dx.doi.org/10.1016/j.patrec.2016.11.022.

Lee, H.-C., 1948. Eigenvalues and canonical forms of matrices with quaternion coefficients. In: Proceedings of the Royal Irish Academy. Section A: Mathematical and Physical Sciences, Vol. 52, Royal Irish Academy, pp. 253–260.

Li, H., Liu, Z., Huang, Y., Shi, Y., 2015. Quaternion generic Fourier descriptor for color object recognition. Pattern Recognit. 48 (12), 3895–3903. http://dx.doi.org/10.1016/j.patcog.2015.06.002.

Maddalena, L., Petrosino, A., 2008. A self-organizing approach to background subtraction for visual surveillance applications. IEEE Trans. Image Process. 17 (7), 1168–1177. http://dx.doi.org/10.1109/TIP.2008.924285.

Maddalena, L., Petrosino, A., 2015. Towards benchmarking scene background initialization. In: International Conference on Image Analysis and Processing. pp. 469–476.

Maddalena, L., Petrosino, A., 2016. Extracting a background image by a multi-modal scene background model. In: 2016 23rd International Conference on Pattern Recognition. ICPR, IEEE, pp. 143–148.

Miao, J., Kou, K.I., 2020. Quaternion-based bilinear factor matrix norm minimization for color image inpainting. IEEE Trans. Signal Process. 68, 5617–5631. http://dx.doi.org/10.1109/TSP.2020.3025519.

Miao, J., Kou, K.I., 2021a. Color image recovery using low-rank quaternion matrix completion algorithm. IEEE Trans. Image Process. 31, 190–201. http://dx.doi.org/10.1109/TIP.2021.3128321.

Miao, J., Kou, K.I., 2021b. Quaternion higher-order singular value decomposition and its applications in color image processing. arXiv preprint arXiv:2101.00364.

Miao, J., Kou, K.I., Liu, W., 2020. Low-rank quaternion tensor completion for recovering color videos and images. Pattern Recognit. 107, 107505. http://dx.doi.org/10.1016/j.patcog.2020.107505.

Mseddi, W.S., Jmal, M., Attia, R., 2019. Real-time scene background initialization based on spatio-temporal neighborhood exploration. Multim. Tools Appl. 78 (6), 7289–7319. http://dx.doi.org/10.1007/s11042-018-6399-1.

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.

Schofield, A.J., Mehta, P., Stonham, T.J., 1996. A system for counting people in video images using neural networks to identify the background scene. Pattern Recognit. 29 (8), 1421–1428. http://dx.doi.org/10.1016/0031-3203(95)00163-8.

Sultana, M., Mahmood, A., Bouwmans, T., Jung, S.K., 2020. Unsupervised adversarial learning for dynamic background modeling. In: International Workshop on Frontiers of Computer Vision. Springer, pp. 248–261.

Sultana, M., Mahmood, A., Javed, S., Jung, S.K., 2019. Unsupervised deep context prediction for background estimation and foreground segmentation. Mach. Vis. Appl. 30 (3), 375–395. http://dx.doi.org/10.1007/s00138-018-0993-0.

Tu, J.H., 2013. Dynamic mode decomposition: Theory and applications (Ph.D. thesis). Princeton University.

Xu, Y., Yu, L., Xu, H., Zhang, H., Nguyen, T., 2015. Vector sparse representation of color image using quaternion matrix analysis. IEEE Trans. Image Process. 24 (4), 1315–1329. http://dx.doi.org/10.1109/TIP.2015.2397314.

Yalman, Y., ERTÜRK, İ., 2013. A new color image quality measure based on YUV transformation and PSNR for human vision system. Turkish J. Electr. Eng. Comput. Sci. 21 (2), 603–612. http://dx.doi.org/10.3906/elk-1111-11.

Zhang, F., 1997. Quaternions and matrices of quaternions. Linear Algebra Appl. 251, 21–57. http://dx.doi.org/10.1016/0024-3795(95)00543-9.

Zivkovic, Z., Van Der Heijden, F., 2006. Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recognit. Lett. 27 (7), 773–780. http://dx.doi.org/10.1016/j.patrec.2005.11.005.

Zou, C., Kou, K.I., Wang, Y., 2016. Quaternion collaborative and sparse representation with application to color face recognition. IEEE Trans. Image Process. 25 (7), 3287–3302. http://dx.doi.org/10.1109/TIP.2016.2567077.