Chaotic Features for Traffic Video Classification

Yong Wang [1] and Shiqiang Hu [1]
[1] School of Aeronautics and Astronautics, Shanghai Jiao Tong University
Shanghai, 200240 - China
[e-mail: wysjtu2008@gmail.com]
[e-mail: sqhu@sjtu.edu.cn]
*Corresponding author: Shiqiang Hu

Received December 20, 2013; revised March 11, 2014; revised April 18, 2014; accepted May 14, 2014; published August 29, 2014

Abstract

This paper proposes a novel framework for traffic video classification based on chaotic features. First, each pixel intensity series in the video is modeled as a time series. Second, the chaos theory is employed to generate chaotic features. Each video is then represented by a feature vector matrix. Third, the mean shift clustering algorithm is used to cluster the feature vectors. Finally, the earth mover's distance (EMD) is employed to obtain a distance matrix by comparing the similarity based on the segmentation results. The distance matrix is transformed into a matching matrix, which is evaluated in the classification task. Experimental results show good traffic video classification performance, with robustness to environmental conditions, such as occlusions and variable lighting.

Keywords: Traffic video classification, Chaotic features, Earth mover's distance

http://dx.doi.org/10.3837/tiis.2014.08.015

1. Introduction

Traffic monitoring is a fundamental issue confronting many urban centers. A key step in addressing this issue is to gather real-time information on traffic flows. Traditional solutions have mainly involved burying inductive-loop detectors underneath roads to count vehicles traveling over them. However, such methods are becoming less feasible because of installation costs and the disruption of roadways.

Video technology serves an increasingly important function in traffic monitoring systems [1–6, 18, 19]. The capability to monitor traffic flow automatically helps in reducing the workload of human operators, identifying illegal vehicles, and providing forensic clues, such as vehicle speed and traffic congestion. Building and using large camera networks to monitor traffic can reduce the number of accidents and traffic jams in urban highways.

Two approaches can be employed for traffic video classification. The first method is based on vehicle detection and tracking and involves three steps [2, 3, 20–22, 26]. First, vehicles are detected by motion segmentation [2] or background subtraction [3, 20]. Second, vehicles are tracked by various tracking algorithms [20, 22, 24, 25], such as rule-based reasoning [2] and the Kalman filter [3]. Third, trajectories are represented as curves or vehicle attributes (area, pattern, and direction). The second method models traffic flow holistically to avoid the need to track vehicles [1, 23]. In [4], features that describe traffic speed and density from MPEG video data are extracted as training sets. The training data are then learned by the Gaussian mixture and hidden Markov models to detect traffic conditions. The maximum likelihood criterion is used to calculate the confidence score, which determines the classification.

The aforementioned methods have the following disadvantages: (i) Motion detection is difficult to implement under varying environmental conditions, especially in crowd scenarios, lighting changes, and occlusions. (ii) Low resolution poses a great challenge for tracking. The drawback of the second approach is that extracting reliable motion cues attributed to traffic scenarios, especially congestion, is difficult to achieve. To overcome these drawbacks, new modeling techniques have been established. Linear dynamic systems (LDS) have been employed to model traffic flows. In [5], the autoregressive (AR) stochastic process with spatial and temporal components is employed to model the traffic flow, and classification performance shows promising results. Linear dynamic systems usually assume the model first-order Markov property or linearity, which restricts the modeling of adverse traffic video scenes.

Notably, the LDS-based model covers traffic flow in a holistic manner to some degree, but loses the motion information of the video. Meanwhile, the traditional pixel model preserves all spatial information, but typically fails to capture temporal information. To cover the integral video without losing the spatial information, the pixel intensity series is proposed as the video descriptor. When a car goes through the surveillance area, the pixel intensity series changes. The changes become more frequent and intense as more cars go through. Changes in the pixel intensity series also indicate traffic conditions, such as light traffic, medium traffic, or heavy traffic.

However, the constraint pixel intensity series has several limitations that prevent its use in this work. First is the alignment problem. The alignment algorithm should be applied to compare the pixel intensity series. Second is that the raw pixel intensity series includes more information than needed, similar to pixels in image analysis. Pixels in an image represent all

image information, whereas local descriptors are developed to depict the image precisely [32, 33]. The relationships among pixels are used to formulate a hypergraph for image classification [30]. Pixels are combined with contextual cues to detect salient regions [31]. Both approaches use pixels with other information to achieve better performance. Unlike the case in image analysis, motion information is important in video analysis. Numerous methods have been proposed for analyzing the time series. these methods include autoregressive models, moving average models, and autoregressive moving average models. Nonlinearity poses a great challenge for these methods. The chaos theory is chosen to characterize the time series to overcome the difficulty in identifying the model of the time series and to represent the time series accurately. The chaos theory has been studied for several decades. The theory is widely used in the field of econometrics and weather forecasting for its capability to characterize nonlinear systems effectively. The chaos theory has recently been introduced to the computer vision community for action recognition [27], anomaly detection [28], and dynamic scene recognition [29].

This paper proposes a framework to model the pixel intensity series for a holistic comparison of traffic conditions. The function of chaotic features in the representation of traffic videos is studied. The problem of finding generalizable methods for characterizing unconstraint traffic videos through a proposed feature vector is addressed. Finally, we demonstrate how the feature vector facilitates meaningful traffic video organization and accurate traffic video classification.

The remainder of the paper is organized as follows: Section 2 provides the workflow of the framework. Section 3 introduces the concept of the chaos theory and the chaotic features used in this work. Section 4 presents the feature vector clustering algorithm. Section 5 describes the feature matching algorithm. Section 6 presents the experimental results and discussions. Finally, Section 7 concludes the paper.

2. Overview of the Framework

Fig. 1 shows the change in the pixel intensity series over time. The central part of the figure shows one frame from a traffic video and the change in the intensity of four pixels over time. The x-axis denotes time, whereas the y-axis denotes the gray value. Accordingly, the video can be segmented into static and traffic parts based on the pixel intensity series. The different parts and the cluster center are conjectured to be two important clues for traffic condition categorizations. Therefore, the video is composed of a $W*L$ matrix of time series, where W and L are the dimensions of the frames in the video.

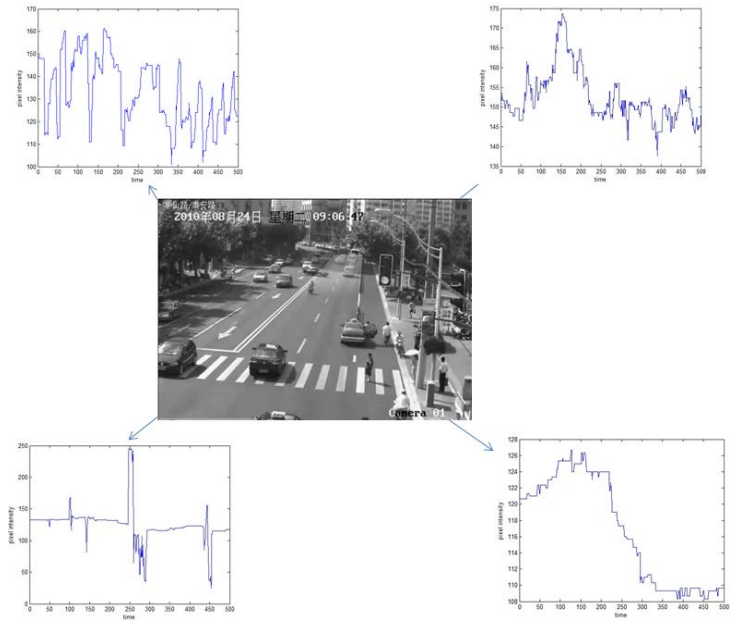

Fig. 1. Pixel intensity change in a traffic video over time

Fig. 2 shows a summary of our algorithm. Details on the generation of feature vectors and feature matching are presented. Each pixel intensity series is modeled as a chaotic time series in the traffic video, and chaotic features are extracted. These features are used to form a feature vector. A traffic video is represented by a feature vector matrix. The mean shift algorithm is applied to the feature vector matrix to summarize the distribution of the feature vectors in the form of a signature that consists of cluster centers and relative weights. The signature is a descriptive representation of the distribution of feature vectors in each video. Earth mover's distance (EMD) is employed to compare signatures in different videos. The entries of an EMD matrix record the similarity between each pair of signatures in the video dataset for use in classification tasks.

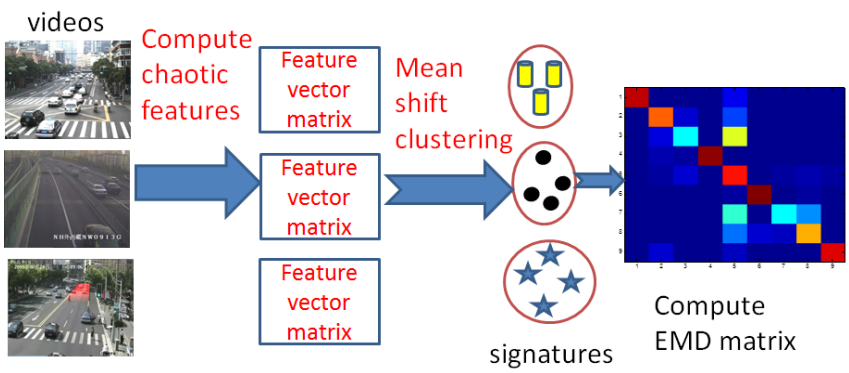

Fig. 2. Flow chart of the proposed algorithm

3. Chaotic Features

The chaotic features are introduced based on the chaos theory [7]. Embedding refers to the mapping from a one-dimensional space to an m-dimensional space. Dynamical systems are characterized as mapping functions that describe how variables change over time, i.e., $x(t) = f(x(t-1))$. The state variable $x(t) = [x_1(t), x_2(t), \cdots, x_n(t)] \in R^n$ defines the status of the system at time t. Takens' theorem [8] states that an embedding exists from an original state space to a reconstructed state space. The underlying idea is that, for a sufficiently large embedding dimension m and embedding delay τ, the vector $x'(t) = [x_i(t), x_{i+\tau}(t), \cdots, x_{i+m\tau}(t)]$ performs the same functions as the original variables of the system. The embedding delay τ is computed by using mutual information [9]. The embedding dimension d is computed by using the false nearest neighbors [10]. Once the two variables are determined, the state $x(t)$ can be written into a matrix

$$X = \begin{pmatrix} x_0 & x_\tau & \cdots & x_{(m-1)\tau} \\ x_1 & x_{\tau+1} & \cdots & x_{(m-1)\tau+1} \\ x_2 & x_{\tau+2} & \cdots & x_{(m-1)\tau+2} \\ \cdots & \cdots & \cdots & \cdots \end{pmatrix} \tag{1}$$

3.1 Box Counting Dimension

The box counting dimension [7] presents an upper bound on the Hausdorff dimension that characterizes the self-similarity of a set. If a point set is covered by a regular grid of boxes of length r, and $N(r)$ is the number of boxes which contain at least one point, then for a self-similar set,

$$D_b = \lim_{r\to 0} \frac{\ln N(r)}{\ln \frac{1}{r}} \tag{2}$$

D_b is called the box counting dimension.

3. 2 Information Dimension [7]

The information dimension specifies how this amount of information scales with the radius ε, which is defined as

$$D_I = \lim_{\epsilon\to 0} \frac{\langle \ln p_\varepsilon \rangle_\mu}{\ln \varepsilon} \tag{3}$$

where μ is a fractal measure defined in state space, $p_\varepsilon(x)$ denotes the probability of finding a typical trajectory in a ball of radius ε around x, and $\langle \ln p_\varepsilon \rangle_\mu$ is the average Shannon information needed to specify a point x with accuracy ε.

3. 3 Correlation Dimension (CD) [7]

The CD measures the change in the density of the phase space with respect to the neighboring point within a radius ϵ and can be calculated as the slope of a graph by plotting $\ln c(\epsilon)$ and $\ln \epsilon$.

$$D_c = \lim_{\epsilon\to 0} \frac{\ln c(\varepsilon)}{\ln \varepsilon} \tag{4}$$

3. 4 Feature Vector

The standard variance of the pixel intensity series encodes the fluctuation information of the time series. Such information is important for classification. The embedding dimension

and embedding delay characterize the geometry structure of the pixel intensity series. The standard variance S is integrated with the chaotic features in the feature vector, F={ τ, m, D_c, D_b, D_I, SV }. Given a *W*L*T* sequence, *W*, *L*, and *T* are the width, length, and time dimension of the sequence, respectively. The chaotic features of each pixel intensity series are extracted, and the video is represented by a W*L*6 dimensional feature matrix.

4. Feature Clustering

Several clustering algorithms can be used for feature clustering. Unlike the k-means and the Gaussian mixture model that need to pre-define the number of clusters, the mean shift algorithm can automatically cluster the features with only one parameter having to be fixed. The parameter, bandwidth, is easy to determine because it has a physical meaning. Therefore, the mean shift algorithm [11][12] is used for feature clustering.

Given n feature vectors f_i, $i = 1, \cdots, n$ in the d-dimensional space R^d, the mean feature vector is given by

$$M(f) = \frac{\sum_{i=1}^{n} G_H(f_i - f) w(f_i)(f_i - f)}{\sum_{i=1}^{n} G_H(f_i - f) w(f_i)} \tag{5}$$

where the profile of kernel G is defined as a function g: $[0,\infty) \rightarrow R$, such that $G(f) = g(\|f\|^2)$, and profiles k and g satisfy $g(f) = -k'(f)$. H is a symmetric positive definite d*d bandwidth matrix. $w(f_i) \geq 0$ is the weight of sample points. The goal of mean shift clustering is to identify the local maxima feature center f_c and assign a label to each feature.

The mean shift algorithm is shown as follows:

(1) The number of search windows is defined at a random location in the feature space.
(2) The initial feature vector f_0 is chosen.
(3) The neighbors of point f_0 are those within a kernel window centered at f_0. The mean shift vector is found as a weighted sum of neighbors, $f_1 = f_0 + M(f)$, where $M(f)$ is the mean shift vector at point f_0 and is computed by Equation 5.
(4) Step 3 is repeated until the mean shift vector is considered to be the zero vector because its magnitude is less than a predetermined threshold. Therefore, f_n is the mode of the component to which the point f_0 belongs.
(5) The feature vectors with similar modes (P') are then merged into components.
(6) Class labels are assigned to clusters.

5. Feature Matching

An appropriate similarity measure has to be defined to compute for the similarities between videos that are represented by feature clusters. The EMD algorithm [13], which compares similarities among images, perform promising results in several applications, such as content-based image retrieval and texture classification [14]. The feature cluster representation of a set of clusters is similar to the signature representation, which is defined as a set of k clusters and relative weights. Therefore, the EMD algorithm is applied to compute for the feature cluster similarities, as shown in **Fig. 3**. The feature cluster can be seen as a signature, e.g., $\left((p_i, wp_i) \middle| 1 \leq i \leq m\right)$, where each cluster is represented by the mean feature vector p_i and the weight of the feature vector wp_i. Computing the EMD cost is based on a solution to the transportation problem [15]. Matching feature clusters can be naturally cast as a transportation problem by defining one feature cluster in a feature vector matrix as the supplier

and the other as the consumer, as well as by setting the cost for a supplier–consumer pair to be equal to the ground distance between an element in the first feature cluster and an element in the second.

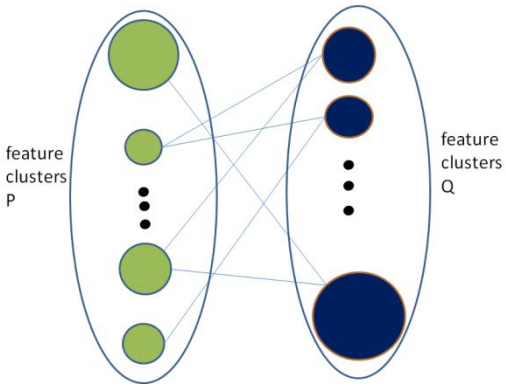

Fig. 3. Example of EMD-based matching between two feature clusters P and Q; lines indicate the flow between the two clusters

Let $P = \{((p_i, wp_i) | 1 \le i \le m)\}$ and $Q = \{((q_j, wq_j) | 1 \le j \le n)\}$ be two feature clusters, where p_i and q_i are the mean feature cluster, wp_i and wq_i are the weights of the feature cluster, and m and n are the number of feature clusters. The distance is defined as

$$\mathrm{EMD}(P, Q) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij}}{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}} \tag{6}$$

where $D = \{d_{ij}\}$ is the distance between the two feature cluster p_i and q_j. $F = [f_{ij}]$ is the flow between p_i and q_j. Equation 6 is governed by the following constraints:

$$f_{ij} \ge 0 \qquad 1 \le i \le m, 1 \le j \le n \tag{7}$$

$$\sum_{j=1}^{n} f_{ij} \le wp_i \qquad 1 \le i \le m \tag{8}$$

$$\sum_{i=1}^{m} f_{ij} \le wq_j \qquad 1 \le j \le n \tag{9}$$

$$\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} = \min\left(\sum_{i=1}^{m} wp_i,\ \sum_{j=1}^{n} wq_j\right) \tag{10}$$

The EMD cost is then used in the form of a Gaussian kernel as follows:

$$\mathrm{Kernel}(P, Q) = \exp(-\rho \mathrm{EMD}(P, Q)^2) \tag{11}$$

where ρ is the kernel parameter. The matching matrix used for traffic video classification is obtained by Equation 11.

6. Experimental Results

6.1 Experiment Setup

A dataset consisting of 225 traffic videos in four different surveillance areas was acquired. The dataset contains a variety of traffic scenes. **Fig. 4** shows sample images from each dataset.

Each video clip shrinks to a 50*50 resolution with 50 frames. The surveillance area of Dataset 1 (78 videos) is an intersection during the daytime. The surveillance area of Dataset 2 (50 videos) is an intersection at night. Pedestrians were seen across the road in the two datasets, and vehicles stopped at the red light in Dataset 2. The surveillance area of Dataset 3 (30 videos) is characterized by a light change, which significantly affects the segmentation results. The surveillance area of Dataset 4 (67 videos) is similar to that of Dataset 1, but in a different intersection. The surveillance camera was mounted at a low position. Thus, the vehicle shapes changed significantly as the vehicles approached the camera.

The ground truth classification for each video clip is determined manually. The dataset is classified into four categories according to traffic conditions: red, light, medium, and heavy. Red signifies the occurrence of a red light. Light signifies few vehicles on the road. Medium signifies several vehicles on the road. Heavy signifies the occurrence of traffic jams.

For the classification strategy, the K nearest neighbor classifier with k=5 is chosen as the classifier, and the one vs. all classification strategy is employed. At each time, one video is chosen for testing and the rest are used for training.

Fig. 4. Examples from our dataset

Fig. 5 illustrates part of the computed feature results. The pixel intensity series in different positions shows different chaotic features.

	position	Embedding delay	Embedding dimension	Box counting dimension	Information dimension	Correlation dimension	mean	Standard variance
	(10, 5)	3	6	0.41	1.01	1.33	120.75	19.55
	(25,10)	5	8	0.1	0.13	0.16	192.26	1.46
	(10, 10)	5	7	0.11	0.14	0.17	97.99	1.36
	(25,10)	3	9	0.02	0.04	0.03	103.48	1.97
	(10, 50)	3	3	0.51	1.0	0.78	105.49	4.3
	(10,10)	3	5	0.14	0.3	0.17	108.1	0.27

Fig. 5. Computed features

Fig. 6 provides an example of our segmentation result. Neighboring feature vectors with similar values are clustered. Traffic roads are separated with buildings. Segmentation results vary with different traffic conditions. The EMD algorithm is then employed to compare the traffic conditions according to the segmentation results.

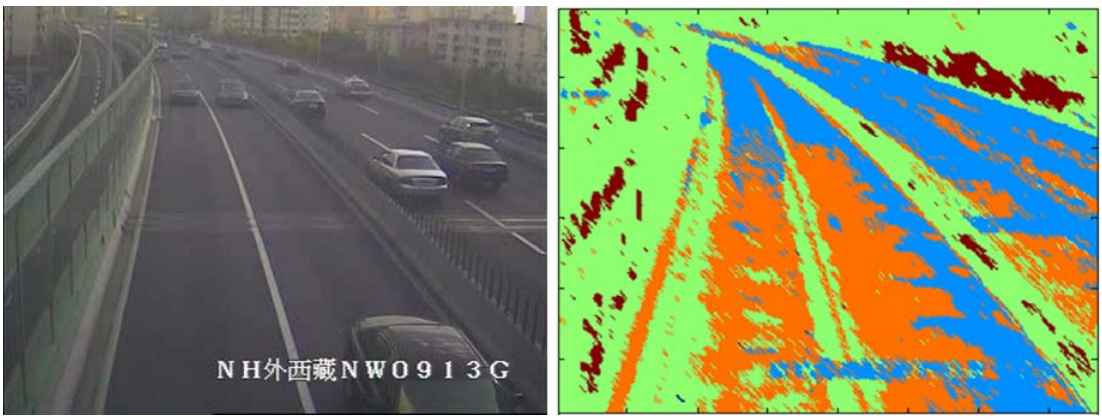

Fig. 6. Segmentation results

6.2 Traffic Classification Results

Fig. 7 shows the confusion matrix of our dataset. The overall classification performance is 73.33%. When the dataset is separated, the classification results for Datasets 1, 2, 3, and 4 are 65.38%, 64%, 83.33%, and 85.07%, respectively. The proposed scheme can approximately classify different traffic conditions.

As shown in **Fig. 7**, the majority of misclassifications occur between neighboring classes

(i.e., light vs. medium, and medium vs. heavy), which is reasonable for such matches because of the indeterminate nature of category boundaries and the corresponding ambiguities of generating ground truth. Furthermore, different surveillance areas and lighting conditions will deteriorate such scenarios. In Datasets 1 and 2, the traffic condition is more complex than that in Datasets 3 and 4. Hence, the boundary effect is more evident.

The reason for the confusion of light, heavy, and red is that light traffic largely depicts the background (i.e., few cars are present), whereas heavy traffic and red depict cars that are virtually at a standstill. From the viewpoint of the change in pixel intensity series, both scenes are similar, thus making mismatches reasonable, especially in Dataset 2 under night time conditions in which the color of the cars will be similar to the color of the road.

The classification rates in Datasets 1 and 2 are lower than that for the other two datasets. The reason for Dataset 1 is that several people occasionally cross the road before the red light flashes, which results in segmentation failures. The reason for Dataset 2 is that the colors of the cars at night are similar to that of the road, which also results in segmentation failures. In most cases, the proposed method can match different traffic conditions accurately.

	Light	Medium	Heavy	Red
Light	50	9	8	1
Medium	7	40	13	2
Heavy	5	6	53	4
Red	0	2	3	22

4 datasets

	Light	Medium	Heavy	Red
Light	24	4	6	0
Medium	3	14	8	0
Heavy	2	3	13	1
Red	0	0	0	0

dataset1

	Light	Medium	Heavy	Red
Light	2	1	1	1
Medium	2	0	1	2
Heavy	0	2	8	3
Red	0	2	3	22

dataset2

	Light	Medium	Heavy	Red
Light	2	3	0	0
Medium	1	13	0	0
Heavy	0	1	10	0
Red	0	0	0	0

dataset3

	Light	Medium	Heavy	Red
Light	22	1	1	0
Medium	1	13	4	0
Heavy	3	0	22	0
Red	0	0	0	0

dataset4

Fig. 7. Confusion matrix for our datasets

The ground truth of the overall dataset and our classification results are shown in **Fig. 8**. The misclassified results are highlighted with red squares.

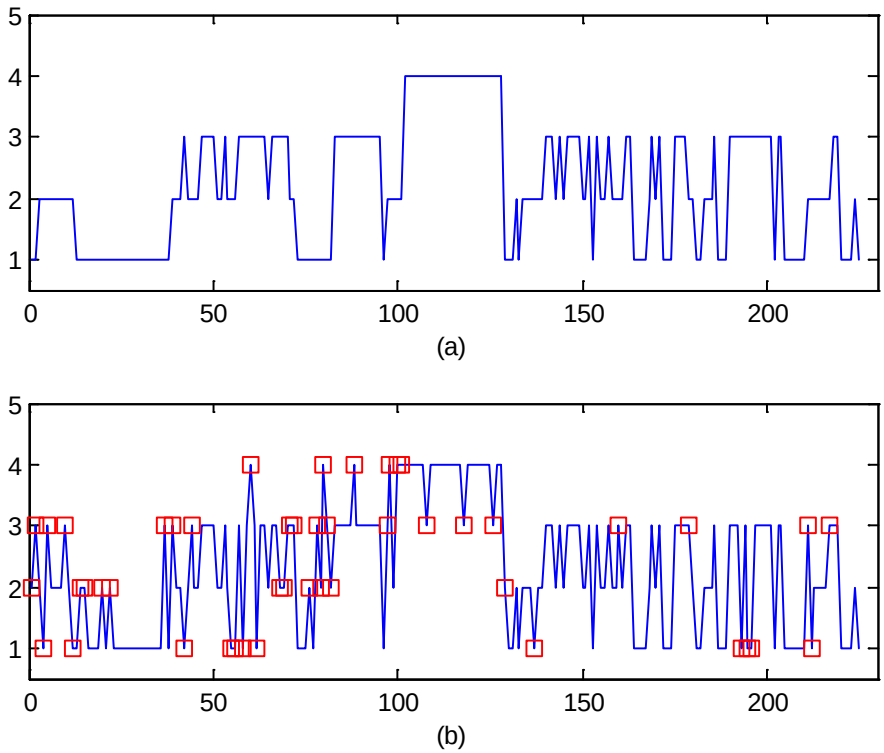

Fig. 8. Classification of traffic videos: (a) Ground truth; (b) Classification results of our method. Errors are highlighted with red squares

Part of the classification results are shown in **Figs. 9**, **11**, **13**, and **15**. The classification results are mainly determined by the EMD algorithm, which compares the segmentation between videos. As shown in Fig. 5, different parts of each video vary significantly. Different cluster centers are important factors affecting the EMD results. Several representative segmentation results with original frames are shown in **Figs. 10**, **12**, **14**, and **16**.

In **Fig. 9**, (a) is a correct classification result, whereas (b) is a wrong classification result. **Fig. 9(b)** shows that the medium condition is confused with the light condition. In the third and sixth columns, the videos are under light traffic condition. However, pedestrians crossing the road, as well as vehicles and motors traveling, affect the segmentation result. As a result, the two videos are similar to the medium condition.

In **Fig. 10**, (a) shows a light traffic condition with a few cars passing through quickly. The cars can be separated from the road. **Fig. 10(b)** shows a medium traffic condition, and the segmentation result shows that the cars integrate with part of the road, thus indicating the presence of more cars. **Fig. 10(c)** shows a heavy traffic condition. Several cars are passing through the road. Pixels of cars dominate each pixel intensity series. The segmentation result shows numerous connected parts.

In **Fig. 11**, (a) is a correct classification result, whereas (b) is a wrong classification result. **Fig. 11(b)** shows that the red light condition is confused with the heavy condition. In the first, fourth, and sixth columns, the videos are in red light, and pedestrians are crossing the road. In the remaining columns, a traffic jam occurs while the vehicles are on the road.

In **Fig. 12(a)**, the traffic light is on red, and pedestrians are crossing the road. The segmentation result shows the path of the people. In **Fig. 12(b)**, the cars are traveling at a slow

speed, which caused the car pixels to occupy a large part of the whole pixel intensity series. The property of light traffic condition pixel intensity series is similar to that of the heavy traffic condition. The segmentation result shows several cars passing through, but only a few cars pass through at low speeds.

In **Fig. 13**, (a) is a correct classification result, whereas (b) is a wrong classification result. In the wrong classification results, sunlight varied significantly in the videos, thus affecting the segmentation results.

In **Fig. 14(a)**, several cars pass through, and the road is separated. In **Fig. 14(b)**, sunlight appears and significantly affects the segmentation result.

In **Fig. 15**, (a) is a correct classification result, whereas (b) is a wrong classification result. Part of the heavy traffic condition is defined as that in which several cars turn right or left. **Fig. 15(b)** shows traffic jams occurring in the video of the first column and vehicles stopping for a long time, the motion information of which is similar to the light traffic condition.

In **Fig. 16 (a)**, several cars proceed northward and from east to south. The segmentation result shows the two paths. In **Fig. 16 (b)**, the segmentation result mainly shows the road and other backgrounds.

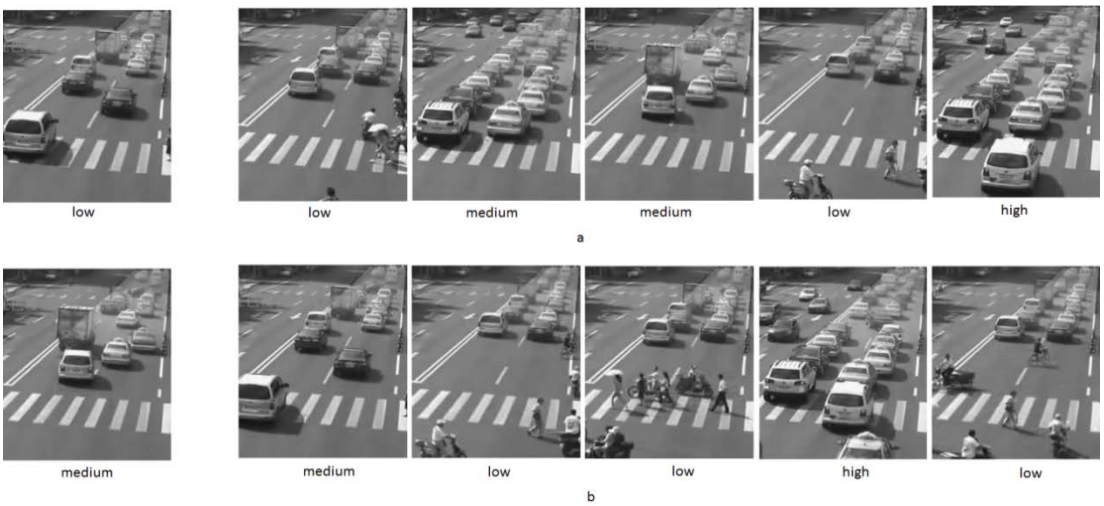

Fig. 9. Video classification results for Dataset 1. (a) A correct classification result; (b) A wrong classification result.

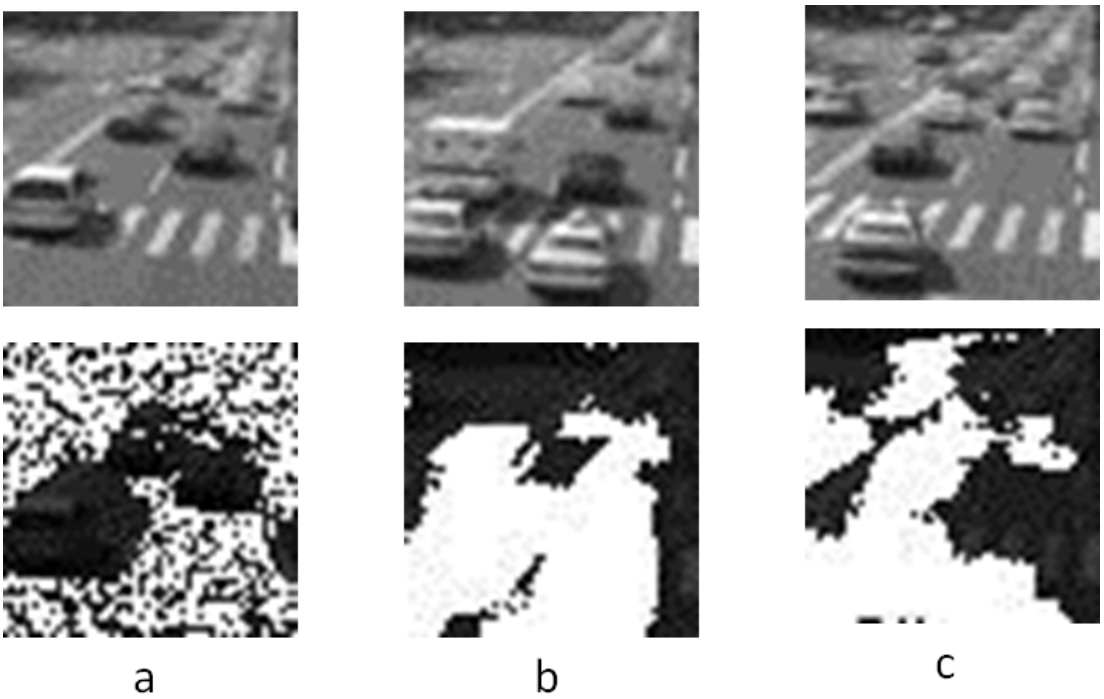

Fig. 10. Example frames of segmentation results. (a) Segmentation result of a light traffic condition; (b) segmentation result of a medium traffic condition; (c) Segmentation result of a heavy traffic condition.

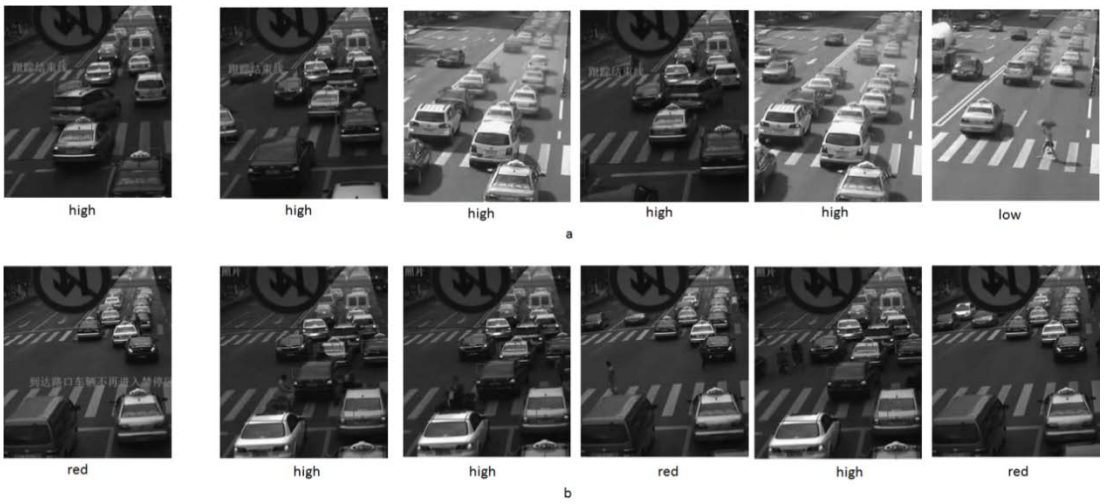

Fig. 11. Video classification results for Dataset 2. (a) A correct classification result; (b) A wrong classification result.

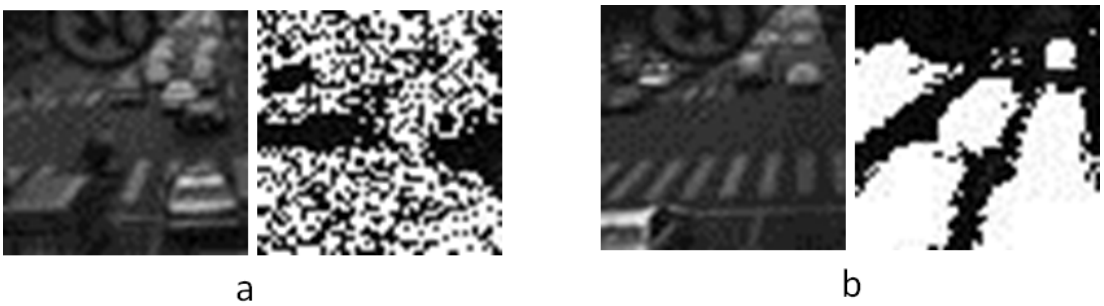

Fig. 12. Example frames of segmentation results. (a) Segmentation result of the traffic light is on red; (b) Segmentation result of the light traffic condition.

Fig. 13. Video classification results for Dataset 3. (a) A correct classification result; (b) A wrong classification result.

Fig. 14. Example frames of segmentation results. (a) Segmentation result of a light traffic condition; (b) Segmentation result of a light traffic condition in sunlight.

Fig. 15. Video classification results for Dataset 4. (a) A correct classification result; (b) A wrong classification result.

Fig. 16. Example frames of segmentation results. (a) Segmentation result of two paths; (b) Segmentation result of the road.

The experiments show that our proposed framework can effectively classify different traffic conditions and is robust to occlusion, low resolution, and sunlight. The segmentation results and feature vector ensure classification accuracy.

6.3 Comparison

LDS [16] is applied to the dataset described above. LDS is a parametric model for spatio-temporal data and can be represented by

$$x(t+1) = Ax(t) + w(t) \quad w(t) \sim N(0, R) \tag{12}$$
$$z(t+1) = Cx(t) + v(t) \quad v(t) \sim N(0, Q) \tag{13}$$

where $x(t)$ is the hidden state vector; $z(t)$ is the observation vector; and $w(t)$ and $v(t)$ are noise components that are modeled as normal with 0 mean and covariance R and Q, respectively. In this work, A is a state-transition matrix, and C is the observation matrix. Let $[z(1), z(2), \ldots z(\tau)] = U\Sigma V^T$, T be the singular value decomposition of the data matrix for τ observations. Then, the model parameters are calculated [16] as $\tilde{C} = U$ and $\tilde{A} = \Sigma V^T D_1 V (V^T D_2 V)^{-1} \Sigma^{-1}$, where $D_1 = [0\ 0; I_{\tau-1}\ 0]$ and $D_1 = [I_{\tau-1}\ 0; 0\ 0]$. The distance metric used was based on subspace angles [17]. The overall classification performance is

30.67%.

Overall, two results were observed for the traffic video representations when evaluated on the datasets. Our proposed feature vector outperforms the LDS approach. The poor performance of the LDS on the dataset can be explained by the viewpoint and illumination change.

A significant limitation of the LDS approach is that the metrics used for comparing traffic video are not designed to be invariant to changes in viewpoint and scale. As a consequence, these methods perform poorly when videos contain traffic scenarios with such variabilities.

Another shortcoming is that the choice of the metric used in these approaches requires that the training and testing data have the same number of pixels. This requirement poses a challenge when one wants to compare local regions of a video sequence, thus adding additional overhead for normalizing all video sequences to the same spatial size.

7. Conclusions

This paper introduced a feature vector matrix representation of traffic videos. Such representation measures traffic conditions under varying environmental conditions and at low resolutions. Compared with most extant approaches, the proposed approach has two advantages: (1) non-reliance on tracking or optical flow estimation and (2) robustness to lighting variation. The experiment was performed on a traffic dataset that we collected, which allowed us to test the descriptive power of our proposed feature vector. Classification results demonstrate that the algorithm is effective.

Acknowledgements

The authors would like to thank the anonymous reviewers for their constructive comments. This work was partly supported by the National Natural Science Foundation of China "61374161"and "61074106".

References

[1] X Yu, Xiao-Dong, Ling-Yu Duan, and Qi Tian, “Highway traffic information extraction from Skycam MPEG video,” in *Proc. of the IEEE Conference on. Intelligent Transportation Systems*, pp. 37-42, Sept.6-6, 2002.

[2] R. Cucchiara, M. Piccardi, and P. Mello, “Image Analysis and Rule-Based Reasoning for a Traffic Monitoring System,” *IEEE Transactions. on Intelligent Transportation Systems*, vol. 1, no. 2, pp. 119-130, January, 2000. Article (CrossRef Link)

[3] Y. K. Jung, K. W. Lee, and Y. S. Ho, “Content-Based Event Retrieval Using Semantic Scene Interpretation for Automated Traffic Surveillance,” *IEEE Transactions. on Intelligent Transportation Systems*, vol. 2, no. 3, pp. 151-163, February, 2001. Article (CrossRef Link)

[4] F. Porikli, and X. Li, “Traffic Congestion Estimation Using HMM Models without Vehicle Tracking,” In *IEEE Intelligent Vehicle Symposium*, pp. 188-193, June 14-17, 2004.

[5] A. B. Chan, and N. Vasconcelos, “Classification and Retrieval of Traffic Video using Auto-Regressive Stochastic Processes,” *IEEE Intelligent Vehicles Symposium*, pp. 771-776, June 6-8, 2005.

[6] Stallkamp, J., Schlipsing, M., Salmen, J., and Igel, C, “Introduction to the Special Issue on Machine Learning for Traffic Sign Recognition,“ *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1481-1483, April, 2012. Article (CrossRef Link)

[7] H. Kantz and T. Schreiber, *Nonlinear Time Series Analysis*, Cambridge: Cambridge University

Press, 1997.
[8] F. Taken, "Detecting Strange Attractors in Turbulence," *Lecture Notes in Mathematics*, ed D. A.Rand & L. S. Young, 1981.
[9] A. M. Fraser and H. L. Swinney, "Independent Coordinates for Strange Attractors from Mutual Information," *Physical Review A*, vol. 33, no. 2, pp. 1134-1140, February, 1986. Article (CrossRef Link)
[10] M. B.Kennel, R. Brown and H. D. I. Abarbanel, "Determining Embedding Dimension for Phase Space Reconstruction using A Geometrical Construction," *Physical Review A*, vol. 45, no. 6, pp. 3403-3411, June, 1992. Article (CrossRef Link)
[11] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions. on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790-799, August, 1995. Article (CrossRef Link)
[12] D. Comaniciu, and P. Meer, "Mean Shift: A Robust Approach Toward Feature Space Analysis," *IEEE Transactions. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603-619, May, 2002. Article (CrossRef Link)
[13] Y. Rubner, C. Tomasi, and L. Guibas, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, February, 2000. Article (CrossRef Link)
[14] D. Xu, and S. F. Chang, "Visual Event Recognition in News Video Using Kernel Methods with Multi-Level Temporal Alignment," in *Proc. of IEEE Conf. Computer Vision and Pattern Recognition*, pp. 1-8, June 17-22, 2007.
[15] G. B. Dantzig, "Application of the simplex method to a transportation problem," *Activity Analysis of Production and Allocation*, pp. 359–373. JohnWiley and Sons, 1951.
[16] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto, "Dynamic texture," *International Journal of Computer Vision*, vol. 51, no. 2, pp. 91-109, February, 2003. Article (CrossRef Link)
[17] R. Martin, "A metric for arma processes," *IEEE Transactions. on Signal Processing*, vol. 48, no. 4, pp. 1164–7, April, 2000. Article (CrossRef Link)
[18] N. Buch, S. A. Velastin and J. Orwell, "A Review of Computer Vision Techniques for the Analysis of Urban Traffic," *IEEE Trans. Intelligent Transportation Systems*, vol. 12, no. 3, pp. 920-939, March, 2011. Article (CrossRef Link)
[19] M. Vargas, S. L. Toral, J. M. Milla and F. Barrero, "A shadow removal algorithm for vehicle detection based on reflectance ratio and edge density," in *Proc. Of IEEE Conf. on Intelligent Transportation Systems*, pp. 1123-1128, Sept. 2010. Article (CrossRef Link)
[20] J. Lai, S. Huang and C. Tseng, "Image-based vehicle tracking and classification on the highway," in *Proc. Of IEEE on Green Circuits and Systems*, pp. 666-670, June 21-23 2010.
[21] K. Robert, "Night-Time Traffic Surveillance: A Robust Framework for Multi-vehicle Detection, Classification and Tracking," in *Proc. Of IEEE on Advanced Video and Signal Based Surveillance*, pp. 1-6, August 29-September 1. 2009.
[22] G. Gritsch, N. Donath, B. Kohn and M. Litzenberger, "Night-time vehicle classification with an embedded, vision system," in *Proc. Of IEEE Conf. on Intelligent Transportation Systems*, pp. 1-6, October 4-7. 2009.
[23] Y. Zou, G. Shi, H. Shi and H. Zhao, "Traffic incident classification at intersections based on image sequences by hmm/svm classifiers," *Multimedia Tools and Applications*, vol. 52, no. 1, pp. 133-145, January, 2011. Article (CrossRef Link)
[24] O. Akoz, M.E. Karsligil, "Severity detection of traffic accidents at intersections based on vehicle motion analysis and multiphase linear regression," in *Proc. of IEEE Conf. on Intelligent Transportation Systems*, pp. 474-479, September 19-22, 2010.
[25] M. Pucher, D. Schabus, P. Schallauer, Y. Lypetskyy, F. Graf, H. Rainer, M. Stadtschnitzer, S. Sternig, J. Birchbauer, W. Schneider, B. Schalko, "Multimodal highway monitoring for robust incident detection," in *Proc. of IEEE Conf. on Intelligent Transportation Systems*., pp. 837-842, September 19-22, 2010.
[26] H. Huang, Z. Cai, S. Shi, X. Ma and Y. Zhu, "Automatic Detection of Vehicle Activities Based on Particle Filter Tracking," *in International Symposium on Computer Science and Computational Technology*, pp. 381-384, December 26-28, 2009.

[27] S. Ali, A. Basharat, and M. Shah, "Chaotic invariants for human action recognition," in *Proc. of IEEE Conf. on Computer Vision*, PP. 1-8, October 14-20, 2007.
[28] S. Wu, B. Moore, and M. Shah, "Chaotic invariants of Lagrangian particle trajectories for anomaly detection in crowded scenes," in *Proc. of IEEE Conf.e on Computer Vision and Pattern Recognition*, pp. 2054-2060, June 13-18 2010.
[29] N. Shroff, P. Turaga, and R. Chellappa, "Moving Vistas: Exploiting Motion for Describing Scenes," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1911-1918, June 13-18, 2010.
[30] Rongrong Ji, Yue Gao, Richang Hong, Qiong Liu, Dacheng Tao, and Xuelong Li, "Spectral-Spatial Constraint Hyperspectral Image Classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 3, pp. 1811-1824, march 2013. Article (CrossRef Link)
[31] Rongrong Ji, Hongxun Yao, Qi Tian, Pengfei Xu, Xiaoshuai Sun, and Xianming Liu, "Context-Aware Semi-Local Feature Detector," *ACM Transactions on Intelligent System and Technology*, vol. 3, no. 3, pp. 44-71, 2012. Article (CrossRef Link)
[32] Yasmin Mussarat, Sharif Muhammad, Mohsin Sajjad and Irum Isma, "Content Based Image Retrieval Using Combined Features of Shape, Color and Relevance Feedback," *KSII Transactions on Internet and Information Systems*, vol. 7, no. 12, pp. 3149-3165. December, 2013. Article (CrossRef Link)
[33] Huy Hoang Nguyen, GueeSang Lee, SooHyung Kim and Hyung Jeong Yang, "An Effective Orientation-based Method and Parameter Space Discretization for Defined Object Segmentation," KSII Transactions on Internet and Information Systems, vol. 7, no. 12, pp. 3180-3199, December, 2013. Article (CrossRef Link)

Yong Wang is a Ph.D. candidate in control science and engineering in the School of Aeronautics and Astronautics at Shanghai Jiao Tong University. His research interests include visual tracking, pattern recognition, and machine learning.

Shiqiang Hu is a Professor and the Chairman of the Department of Aerospace Information and Control at Shanghai Jiao Tong University. He received his M.S. (1998) and Ph.D. (2002) degrees at Beijing Institute of Technology both in electronics and information technology. His research areas include intelligent information processing, image understanding, and nonlinear filtering.