

Classification and Retrieval of Traffic Video Using Auto-Regressive Stochastic Processes

Antoni B. Chan and Nuno Vasconcelos

Department of Electrical and Computer Engineering

University of California, San Diego

abchan@ucsd.edu, nuno@ece.ucsd.edu

Abstract— We propose to model the traffic flow in a video using a holistic generative model that does not require segmentation or tracking. In particular, we adopt the dynamic texture model, an auto-regressive stochastic process, which encodes the appearance and the underlying motion separately into two probability distributions. With this representation, retrieval of similar video sequences and classification of traffic congestion can be performed using the Kullback-Leibler divergence and the Martin distance. Experimental results show good retrieval and classification performance, with robustness to environmental conditions such as variable lighting and shadows.

I. INTRODUCTION

In recent years, the use of video systems for traffic monitoring has shown promise over that of traditional loop detectors. The analysis of traffic video can provide global information, such as overall traffic speed, lane occupancy, and individual lane speed, along with the capability to track individual cars. Because video systems are less disruptive and less costly to install than loop detectors, interest has grown in building and using large camera networks to monitor different aspects of traffic, such as traffic congestion.

Most of the existing work in monitoring traffic uses a vehicle segmentation and tracking framework. First, a potential vehicle is segmented from the scene using motion cues [1] [2], or through background subtraction [3]. Once segmentation is performed, the objects are tracked between frames using rule-based systems [1], Kalman filters, or Condensation [4]. In [3], object trajectories are represented as polynomial curves, which are used for video retrieval. The vehicle tracking framework has the disadvantage that its accuracy is dependent on the quality of the segmentation. The segmentation task becomes more difficult with the presence of adverse environmental conditions, such as lighting (e.g. overcast, glare, night), shadows, occlusion, and blurring. Furthermore, segmentation cannot be performed reliably on low resolution images where the vehicles only span a few pixels. Tracking algorithms also have problems when there are many objects in the scene, which is typically the case for highways scenes with congestion.

Several recent methods finesse the problems associated with vehicle tracking by analyzing the low-level motion vectors provided by MPEG video compression. In [5], the MPEG motion vector field is filtered to remove vectors that are not consistent with vehicle motion, and traffic flow is estimated by averaging the remaining motion vectors. The work of [6] uses a probabilistic approach that models each category of traffic as

a Gaussian-mixture hidden Markov model (GM-HMM), which is learned from feature vectors extracted from the MPEG video stream. Classification is performed by selecting the category corresponding to the GM-HMM of largest likelihood for the query video. While these two methods do not rely on vehicle tracking, they depend on the estimation of motion vectors, which may be subject to noise when there are many vehicles in the scene.

Since most of the information required for the classification of traffic video is contained in the interaction between the many motions that it contains, a holistic representation can be used to capture the variability of the motion field without the need for segmenting or tracking individual components. In this paper, we propose to model the entire motion field as a dynamic texture, which is an auto-regressive stochastic process with both a spatial and a temporal component [7]. Distances between dynamic textures can be computed using either information theoretical measures of divergence (e.g. the Kullback-Leibler divergence) between the associated probability distributions, or through geometric measures based on their observable space (e.g. the Martin distance). Using these distance measures, traffic videos similar to a query can be retrieved, or the traffic congestion of the query can be classified using the nearest neighbors classifier or a support vector machine (SVM) [8] based on the Kullback-Leibler kernel [9]. The probabilistic SVM framework combines the generalization guarantees of the large-margin SVM method, with the robustness of the underlying probabilistic models, and has been successfully applied to object [10] and speech [9] recognition.

Since only the motion is modeled, the proposed framework is inherently invariant to lighting changes. In addition, because the model does not rely on a dense motion field based on pixel similarity (e.g. correlation or optical flow), it is robust to occlusion, blurring, image resolution, and other image transformations. While the system proposed in this work was trained and tested on a single camera view, it can be augmented to handle multiple camera views by transforming the video from each camera view into a common frame of reference.

The paper is organized as follows. In Section II we discuss the dynamic texture model. The Kullback-Leibler divergence and the Martin distance between dynamic textures are introduced in Section III. Finally, Section IV presents results from a video retrieval experiment and a traffic congestion



Fig. 1. Frames from a video of highway traffic. (courtesy Washington State Department of Transportation)

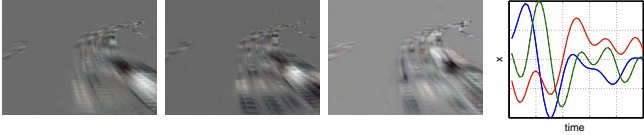


Fig. 2. Dynamic texture model for Figure 1: (left) the first three principal components; (right) the state space trajectory of the corresponding coefficients.

classification experiment.

II. MODELING MOTION FLOW

Various representations of a video sequence as a spatio-temporal texture have been proposed in the vision literature over the last decade. Early methods were based on motion features extracted from the video, e.g. the statistics of the normal flow [12]. More recently, temporal textures have been modeled as generative processes, such as the multi-resolution analysis tree [13] or the spatio-temporal autoregressive (STAR) model [11]. The temporal texture model adopted in this work is the dynamic texture model of [7], an auto-regressive random process (specifically, a linear dynamical system) that includes a hidden state variable and an observation variable. The motion flow is captured by a dynamic generative model, from which the hidden state vector is drawn. The observation vector is then drawn from a second generative model, conditioned on the state variable. Both the hidden state vector and the observation vector are representative of the entire image, enabling a holistic characterization of the motion for the entire sequence.

A. The Dynamic Texture Model

The dynamic texture model [7] is defined by

$$x_{t+1} = Ax_t + Bv_t \quad (1)$$

$$y_t = Cx_t + w_t \quad (2)$$

where $x_t \in \mathbb{R}^n$ is a sequence of n -dimensional hidden (state) random variables, $y_t \in \mathbb{R}^m$ a m -dimensional sequence of observed (video frame) random variables, $v_t \sim_{iid} \mathcal{N}(0, I_n)$ a n -dimensional driving process (typically $n \ll m$), and $w_t \sim_{iid} \mathcal{N}(0, R)$ an observation noise process. The model is parameterized by $\Theta = (A, B, C, R, x_0)$, where $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times n}$, $C \in \mathbb{R}^{m \times n}$, $R \in \mathbb{R}^{m \times m}$, and x_0 is a known initial state. Note that $Bv_t \sim \mathcal{N}(0, Q)$ where $Q = BB^T$. The covariance of the observation noise is assumed to be i.i.d, i.e. $R = \sigma^2 I_m$.

The sequence $\{y_t\}$ encodes the appearance component of the video frames, and the motion component is encoded into

the state sequence $\{x_t\}$. The hidden state is modeled as a first-order Gauss-Markov process, where the future state x_{t+1} is determined by the transition matrix A , the current state x_t , and the driving process v_t . The image frame y_t is a linear combination of the principal components of the entire video sequence, stored in the columns of C , with each component weighted by the corresponding coefficient in the state vector x_t . Figure 1 and 2 show an example of a traffic sequence, its first three principal components, and the corresponding state space coefficients.

B. Parameter Estimation

In general, the parameters of an autoregressive process can be learned by maximum likelihood (e.g. N4SID [14]). However, these solutions are infeasible for dynamic texture models, due to the high dimensionality of the observed image space. A suboptimal (but tractable) alternative [7] is to learn the spatial and temporal parameters separately. If $Y_1^N = [y_1, \dots, y_N] \in \mathbb{R}^{m \times N}$ is the matrix of observed video frames, and $Y_1^N = U\Sigma V^T$ is the singular value decomposition (SVD) this leads to

$$\hat{C} = U \hat{X}_1^N = \Sigma V^T \quad (3)$$

$$\hat{A} = \hat{X}_2^N (\hat{X}_1^{N-1})^\dagger \quad (4)$$

$$\hat{Q} = \frac{1}{N-1} \sum_{i=1}^{N-1} \hat{v}_i \hat{v}_i^T \quad (5)$$

where $\hat{X}_1^N = [\hat{x}_1, \dots, \hat{x}_N]$ is a matrix of state estimates, M^\dagger is the pseudo-inverse of M , and $\hat{v}_t = \hat{x}_{t+1} - \hat{A}\hat{x}_t$.

III. DISTANCE MEASURES BETWEEN DYNAMIC TEXTURES

Since the dynamic texture model provides a probabilistic description in both the state and the image space, two dynamic textures can be compared in either of these spaces. In the former case, the distribution of the first state space is projected into the second state space, and the Kullback-Leibler (KL) divergence is computed between the two distributions. In the latter, the KL divergence is computed between the observed distributions in image space. In addition to these two information theoretic measures, dynamic textures can also be compared using the Martin distance, which is a metric based on the angles between the observable linear subspaces. In the remainder of this section, we discuss these three distance measures.

A. Probability Distributions

The state of the dynamic texture is a Gauss-Markov process [15], where the conditional probability of state x_t given state x_{t-1} is Gaussian

$$p(x_t | x_{t-1}) = G(x_t, Ax_{t-1}, Q) \quad (6)$$

where $G(x, \mu, Q) = ((2\pi)^n |Q|)^{-\frac{1}{2}} \exp(-\frac{1}{2} \|x - \mu\|_Q^2)$, and $\|x\|_Q^2 = x^T Q^{-1} x$. Recursively substituting into (1), we obtain the probability of a single state x_t

$$p(x_t) = G(x_t, \mu_t, S_t) \quad (7)$$

with mean $\mu_t = A^t x_0$ and covariance $S_t = \sum_{i=0}^{t-1} A^i Q (A^i)^T$. Let $x_1^\tau = (x_1, \dots, x_\tau)$ be a sequence of τ state vectors, then the probability of a state sequence is also Gaussian

$$p(x_1^\tau) = G(x_1^\tau, \mu, \Sigma) \quad (8)$$

where $\mu = [\mu_1^T \ \dots \ \mu_\tau^T]^T$ and the covariance is

$$\Sigma = \begin{bmatrix} S_1 & (AS_1)^T & \dots & (A^{\tau-1}S_1)^T \\ AS_1 & S_2 & \dots & (A^{\tau-2}S_2)^T \\ \vdots & \vdots & \ddots & \vdots \\ A^{\tau-1}S_1 & A^{\tau-2}S_2 & \dots & S_\tau \end{bmatrix} \quad (9)$$

The image sequence y_1^τ is a linear transformation of the state sequence, and is thus given by

$$p(y_1^\tau) = G(y_1^\tau, \gamma, \Phi) \quad (10)$$

where $\gamma = \mathbf{C}\mu$ and $\Phi = \mathbf{C}\Sigma\mathbf{C}^T + \mathbf{R}$, and \mathbf{C} and \mathbf{R} are block diagonal matrices formed from C and R respectively.

B. Projection between State Spaces

The KL divergence between state spaces cannot be computed directly because each dynamic texture uses a different PCA space. Instead, the original state space x_1 must be projected into the second state space x_2 using the transformation $\hat{x}_1 = C_2^T C_1 x_1$. If x_1 is the state space with parameters (A_1, B_1, x_{01}) , then the transformed \hat{x}_1 will have parameters given by $\hat{A}_1 = (C_2^T C_1) A_1 (C_2^T C_1)^{-1}$, $\hat{B}_1 = (C_2^T C_1) B_1$, and $\hat{x}_{01} = (C_2^T C_1) x_{01}$. The KL divergence between state spaces can now be computed with this transformed state model.

C. KL Divergence between State Spaces

The KL divergence rate between two random processes with distributions, $p(X)$ and $q(X)$ over $X = (x_1, x_2, \dots)$, is defined as

$$D(p(X) \| q(X)) = \lim_{\tau \rightarrow \infty} \frac{1}{\tau} D(p(x_1^\tau) \| q(x_1^\tau)). \quad (11)$$

where $D(p \| q) = \int_{\mathcal{X}} p(x) \log \frac{p(x)}{q(x)} dx$ is the KL divergence between the two pdfs $p(x)$ and $q(x)$. If $p(x_1^\tau)$ and $q(x_1^\tau)$ are the state pdfs of two dynamic textures parameterized by (A_1, Q_1, x_{01}) and (A_2, Q_2, x_{02}) , the KL divergence on the RHS of (11) is given by [16],

$$\begin{aligned} \frac{1}{\tau} D(p(x_1^\tau) \| q(x_1^\tau)) &= \frac{1}{2} \left[\log \frac{|Q_2|}{|Q_1|} \right. \\ &+ \text{tr}(Q_2^{-1} Q_1) - n + \frac{1}{\tau} \|A_1 x_{01} - A_2 x_{02}\|_{Q_2}^2 \\ &\left. + \frac{1}{\tau} \sum_{i=2}^{\tau} \text{tr}(\bar{A}^T Q_2^{-1} \bar{A} (S_{i-1} + \mu_{i-1} \mu_{i-1}^T)) \right] \end{aligned} \quad (12)$$

where $\bar{A} = A_1 - A_2$, and S_{i-1} and μ_{i-1} are the covariance and mean associated with the state x_{i-1} of the first dynamic texture.

D. KL Divergence in Image Space

The KL divergence rate between image probabilities is found by noting that $p(y_1^\tau) \sim \mathcal{N}(\gamma_1, \Phi_1)$ and $q(y_1^\tau) \sim \mathcal{N}(\gamma_2, \Phi_2)$ and hence,

$$\begin{aligned} D(p(y_1^\tau) \| q(y_1^\tau)) &= \frac{1}{2} \left[\log \frac{|\Phi_2|}{|\Phi_1|} \right. \\ &\left. + \text{tr}(\Phi_2^{-1} \Phi_1) + \|\gamma_1 - \gamma_2\|_{\Phi_2}^2 - m\tau \right] \end{aligned} \quad (13)$$

Direct computation of (13) is intractable since the covariance matrices have $(m\tau)^2$ elements. By using several matrix identities, (13) can be rewritten into a computationally efficient recursive form [17].

E. Martin Distance

The Martin distance [21], introduced in [18] as a distance metric between two dynamic textures, is based on the principal angles between the subspaces of the extended observability matrices of the two textures [22]. Formally, let Θ_1 and Θ_2 be the parameters representing two dynamic textures, then the Martin distance is defined as,

$$d(\Theta_1, \Theta_2)^2 = \log \prod_{i=1}^n \frac{1}{\cos^2 \theta_i} \quad (14)$$

where θ_i is the i -th principal angle between the extended observability matrices \mathcal{O}_1^∞ and \mathcal{O}_2^∞ , defined as $\mathcal{O}_i^\infty = [C_i^T \ A_i^T C_i^T \ \dots \ (A_i^T)^n C_i^T \ \dots]^T$. It is shown in [22] that the principal angles can be computed as the solution to the following generalized eigenvalue problem:

$$\begin{bmatrix} 0 & \mathcal{O}_{12}^\infty \\ (\mathcal{O}_{12}^\infty)^T & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \lambda \begin{bmatrix} \mathcal{O}_{11}^\infty & 0 \\ 0 & \mathcal{O}_{22}^\infty \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (15)$$

subject to $x^T \mathcal{O}_{11}^\infty x = 1$ and $y^T \mathcal{O}_{22}^\infty y = 1$, where

$$\mathcal{O}_{ij}^\tau = (\mathcal{O}_i^\tau)^T \mathcal{O}_j^\tau = \sum_{t=0}^{\tau} (A_i^t)^T C_i^T C_j A_j^t \quad (16)$$

The first n largest eigenvalues are the cosines of the principal angles, i.e. $\lambda_i = \cos \theta_i$ for $i = 1, 2, \dots, n$.

IV. EXPERIMENTAL EVALUATION

The proposed motion model was evaluated in a retrieval and classification problem for traffic video sequences.

A. Traffic Video Database and Setup

The traffic video database consisted of 254 video sequences (20 total minutes) of daytime highway traffic in Seattle, collected from a single stationary traffic camera over two days [19]. The database contains a variety of traffic patterns and weather conditions (e.g. overcast, raining, sunny). Each video clip was recorded in color at 10 fps with a resolution of 320 x 240 pixels and had 42 to 52 frames. A representative video patch was created by converting the video to grayscale, shrinking the resolution by 4, and selecting a 48 x 48 window over the area with the most total motion. For each video patch, the mean image was subtracted and the pixel intensities were normalized to have unit variance.



Fig. 3. Examples of road patches used for retrieval and classification.

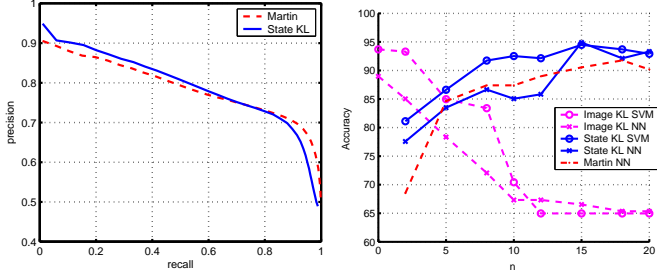


Fig. 4. (left) Precision-recall curve for traffic video retrieval; (right) Accuracy of congestion classification. The x-axis is the number of principal components (n) used in the dynamic texture model.

The parameters of the dynamic texture model were learned for each video patch using the method of Section II-B. The image KL between dynamic textures was computed using $\tau = 25$, and the state KL and Martin distance with $\tau = 250$. To ensure that the KL divergence converged, the transition matrix A was scaled so that the largest eigenvalues lie on the unit circle. In addition, the covariance of the driving process was regularized to prevent problems with singular matrices, i.e. we set $Q' = Q + I_n$.

The ground truth classification for each video clip was determined by hand, and the database contained a total of 44 instances of heavy traffic (slow or stop and go speeds), 45 of medium traffic (reduced speed), and 165 of light traffic (normal speed). Figure 3 shows a representative set of clips from this database. All experimental results were averaged over four trials, where in each trial the dataset was split differently with 75% used for training and cross-validation, and 25% reserved for testing.

B. Retrieval of Traffic Video

The motion model was tested in a video retrieval experiment where the goal was to retrieve instances of traffic patterns that were similar to the query video. For the experiment, dynamic texture models with 15 principal components were used, and the state KL divergence and Martin distance were used as similarity measures. The precision-recall curve for traffic video retrieval is presented in Figure 4 (left). The Martin distance and state KL divergence perform similarly, with the state KL distance performing slightly better. Figure 6 shows the results of several queries for light, medium, and heavy traffic using the state KL divergence. A query using nighttime sequences outside of the original database is presented in Figures 6d, 6e, and 6f, and shows that the retrieval system is robust to variable lighting conditions. In addition the framework is robust to occlusion and blurring due to raindrops on the camera lens,

		PREDICTED		
		heavy	medium	light
TRUE	heavy	37	7	0
	medium	4	39	2
	light	0	1	164

TABLE I

CONFUSION MATRIX FOR CONGESTION CLASSIFICATION

as seen in the 3rd and 5th results of Figure 6e.

C. Classification of Traffic Congestion

The traffic congestion in a video clip was classified using a nearest neighbors (NN) classifier with the image KL, state KL, or Martin distances. Classification was also performed using a support vector machine (SVM) with the Kullback-Leibler kernel [9]. A one-versus-all scheme was used to learn the multi-class problem, and the SVM parameters were selected using 3-fold cross-validation over the training set. The `libsvm` software package [20] was used for training and testing.

D. Classification Results

Figure 4 (right) presents the results obtained in the classification experiment. It can be seen that the two state KL classifiers outperform the Martin NN classifier. In particular, the state KL-SVM combination is consistently better, and the state KL-NN combination is better for $n \geq 15$ and also achieves a higher maximum accuracy. The SVM-KL achieves better classification performance than NN-KL, and the robustness of the SVM classifier to a poor selection of the number of components indicates that it has better generalization ability. The performance of the image KL classifiers drops as the number of principal components increases. This is because there is no abstraction of motion in the image space. Hence as the amount of motion that can be modeled is increased, the classification problem using this distance measure becomes harder.

The best classifier, the state KL-SVM with 15 components, has an overall accuracy of 94.5%. Table I shows its confusion matrix, averaged over the four test trials. Figure 7 shows several classification examples under different lighting conditions: (a) sunny lighting, including strong shadows; and (b) overcast lighting, including raindrops on the camera lens. Several night time videos outside the original database were also fed through the same classifier. Even though the classifier was trained with video taken during the day, it is still able to correctly label the nighttime video sequences. This is particularly interesting since the visual appearance of a car during the day is vastly different from its appearance at night. These results provide evidence that the dynamic texture model is indeed extracting relevant motion information, and that the SVM framework is capable of using the motion model to discriminate between motion classes.

A final experiment was conducted to identify the traffic pattern of the highway during the day. The SVM and nearest

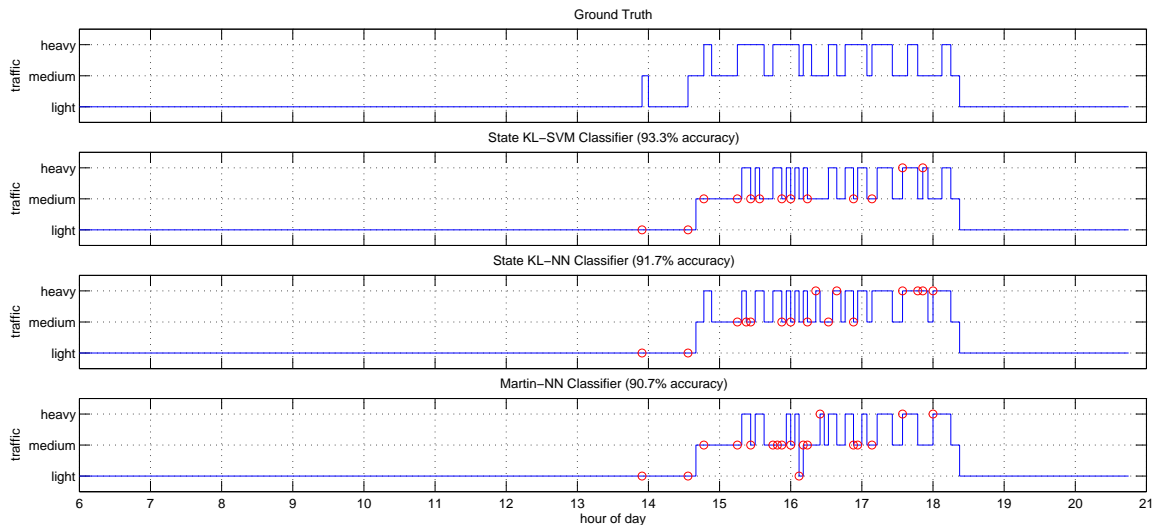


Fig. 5. Classification of congestion in traffic sequences spanning 15 hours: (top to bottom) ground truth; classification using state KL SVM, state KL nearest neighbors, and Martin distance nearest neighbors. Errors are highlighted with circles.

neighbor classifiers were trained using 61 sequences spanning 4 hours from the first day, and tested on 193 sequences spanning 15 hours from the following day. The ground truth classification and the outputs of the state-KL SVM, state-KL NN, and Martin distance NN classifiers are shown in Figure 5. The increase in traffic due to rush hour can be seen between 2:30 PM and 6:30 PM.

V. CONCLUSIONS

We presented a method for modeling traffic flow patterns holistically using dynamic texture models. When compared to previous solutions, the analysis of motion with these models has several advantages: a) it does not require segmentation or tracking of vehicles; b) it does not require estimation of a dense motion vector field; and c) it is robust to lighting variation, blurring, occlusion, and low image resolution. Experimental results using the holistic model show good performance in the domains of video retrieval and classification of traffic congestion.

REFERENCES

- [1] R. Cucchiara, M. Piccardi, and P. Mello. Image Analysis and Rule-Based Reasoning for a Traffic Monitoring System. *IEEE Trans. on Intelligent Transportation Systems*, Vol. 1(2), pp. 119-130, 2000.
- [2] B. Maurin, O. Masoud, and N. Papanikolopoulos. Monitoring Crowded Traffic Scenes. *IEEE Int. Conference on Intelligent Transportation Systems*, pp 19-24, 2002.
- [3] Y.K. Jung, K.W. Lee, and Y.S. Ho. Content-Based Event Retrieval Using Semantic Scene Interpretation for Automated Traffic Surveillance. *IEEE Trans. on Intelligent Transportation Systems*, Vol. 2(3), pp. 151-163, 2001.
- [4] M. Isard and A. Blake. Condensation – Conditional Density Propagation for Visual Tracking. *International Journal of Computer Vision*, Vol. 29(1), pp. 5-28, 1998.
- [5] X.D. Yu, L.Y. Duan, and Q. Tian. Highway Traffic Information Extraction from Skycam MPEG Video. *IEEE Int. Conference on Intelligent Transportation Systems*, pp. 37-41, 2002.
- [6] F. Porikli, and X. Li Traffic Congestion Estimation Using HMM Models Without Vehicle Tracking In *IEEE Intelligent Vehicle Symposium*, pp. 188-193, 2004.
- [7] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto. Dynamic textures. *International Journal of Computer Vision*, (2):91–109, 2003.
- [8] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer Verlag, 1995.
- [9] P. J. Moreno, P. Ho, and N. Vasconcelos. A kullback-leibler divergence based kernel for svm classification in multimedia applications. In *Advances in Neural Information Processing Systems*, Vancouver, Canada, Dec 2003.
- [10] N. Vasconcelos, P. Ho, and P. Moreno. The Kullback-Leibler Kernel as a Framework for Discriminant and Localized Representations for Visual Recognition. In *Proc. European Conference on Computer Vision, Prague, Czech Republic*, 2004.
- [11] M. Szummer and R. Picard. Temporal texture modeling. In *IEEE Conference on Image Processing*, volume 3, pages 823–6, 1996.
- [12] R. Polana and R. C. Nelson. Recognition of motion from temporal texture. In *IEEE Conference on Computer Vision and Pattern Recognition, Proceedings*, pages 129–34, 1992.
- [13] Z. Bar-Joseph, R. El-Yaniv, D. Lischinski, and M. Werman. Texture mixing and texture movie synthesis using statistical learning. *IEEE Transactions on Visualization and Computer Graphics*, 7(2):120–35, 2001.
- [14] P. Van Overschee and B. De Moor. N4sid: Subspace algorithms for the identification of combined deterministic-stochastic systems. *Automatica*, 30:75–93, 1994.
- [15] S. M. Kay *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice-Hall Signal Processing Series, 1993.
- [16] A. B. Chan, and N. Vasconcelos. Probabilistic Kernels for the Classification of Auto-Regressive Visual Processes. In *Proc. CVPR, San Diego*, 2005.
- [17] A. B. Chan, and N. Vasconcelos. Efficient Computation of the KL Divergence between Dynamic Textures. Technical Report SVCL-TR-2004-02, Statistical Visual Computing Lab, UCSD, <http://www.svcl.ucsd.edu>, Nov. 2004.
- [18] P. Saisan, G. Doretto, Y. Wu, and S. Soatto. Dynamic texture recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, Proceedings*, volume 2, pages 58–63, 2001.
- [19] <http://www.wsdot.wa.gov>
- [20] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [21] R. J. Martin. A metric for arma processes. *IEEE Transactions on Signal Processing*, 48(4):1164–70, April 2000.
- [22] K. De Cock and B. De Moor. Subspace angles between linear stochastic models. In *IEEE Conference on Decision and Control, Proceedings*, pages 1561–6, December 2000.

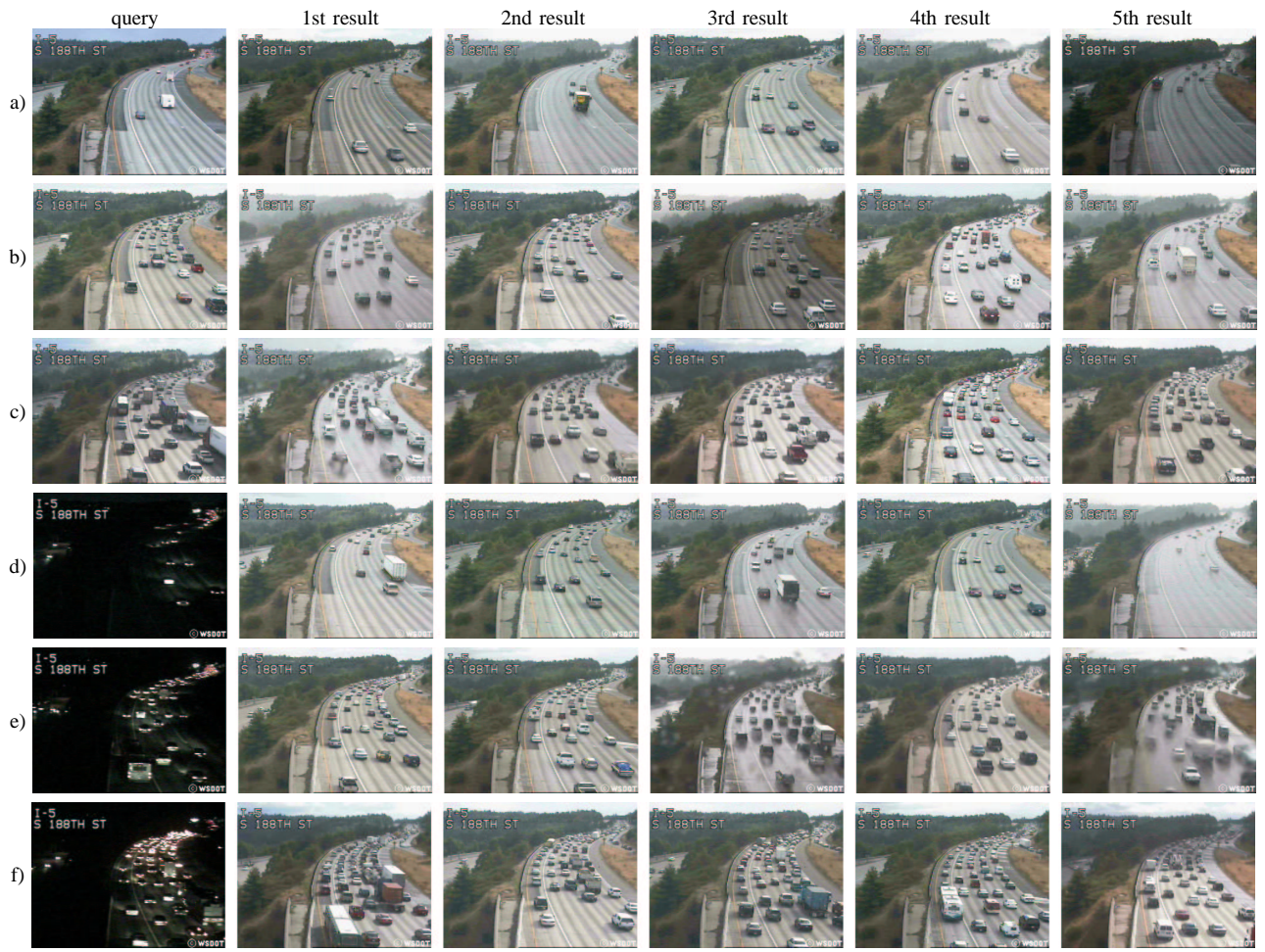


Fig. 6. Video retrieval results for (a) light traffic, (b) medium traffic, and (c) heavy traffic during the day. Retrieval using a night sequence outside the original database for (d) light, (e) medium, and (f) heavy traffic shows robustness to lighting conditions.

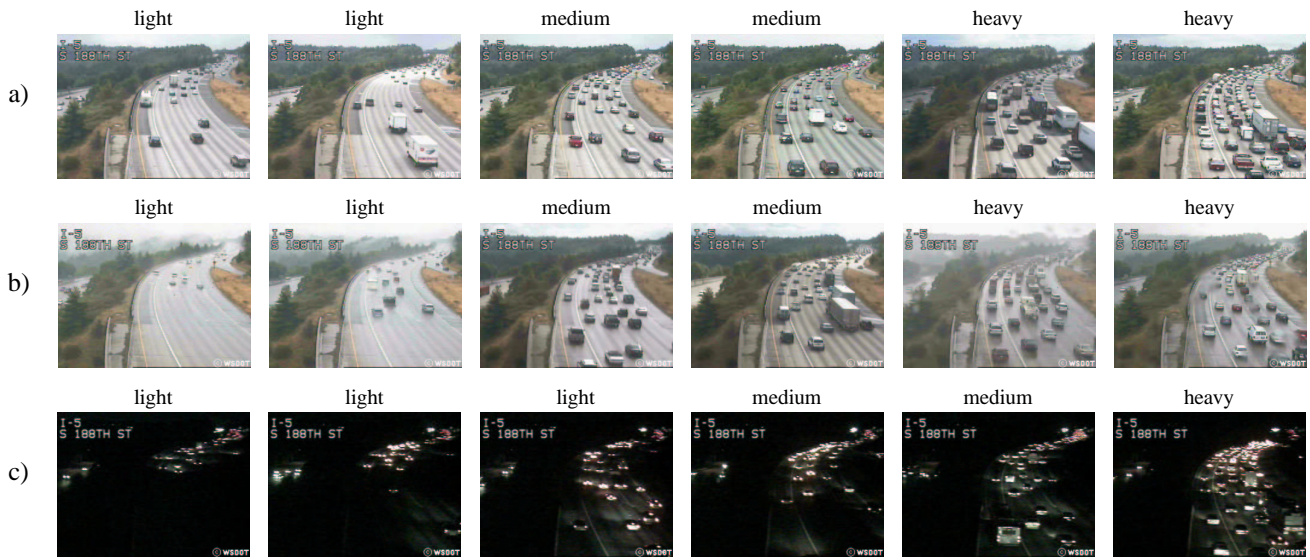


Fig. 7. Classification of traffic congestion in variable lighting conditions: (a) sunny, (b) overcast, and (c) nighttime.