CLUSTER-BASED DICTIONARY LEARNING AND LOCALITY-CONSTRAINED SPARSE RECONSTRUCTION FOR TRAJECTORY CLASSIFICATION

Ce Li, Feng Yang

College of Mechanical Electronic and Information Engineering
China University of Mining & Technology, Beijing, China

ABSTRACT

Trajectory classification has been extensively investigated in recent years, however, the problems about automatically modeling unlabeled and incomplete trajectories are far from being solved. In this paper, we propose a Cluster-based Dictionary Learning (CDL) approach that firstly constructs an initial cluster-based dictionary by K-means clustering and incrementally updates by exploring the importance of the label consistency constraint and classification optimization. Finally, a multiple-category classifier for trajectory is obtained with Locality-constrained Sparse Reconstruction (LSR) that combines both sparsity and local adaptability for robust trajectory classification. Experimental results show that our approach outperforms several recent approaches.

Index Terms— Trajectory classification, Cluster-based dictionary learning, Locality-constrained sparse reconstruction

1. INTRODUCTION

Visual trajectory classification contributes to a variety of applications [1, 2], including the identification of crowdedness [3], behaviors, activities [1] and events [4] of video scenes. Although extensively investigated, the limited sizes of labeled sample sets [5], and the local variation and noises of the trajectories [1] are still an open research problems.

For trajectory representation [6] are commonly used as feature vectors according to the movements of objects. On trajectories of variable length, re-sampling and linear interpolation strategies [7] are often used to align the vectors, while function approximation methods[8], polynomial based curve fitting, Haar wavelet transform, minimum error-based polygonal approximation, B-spline curves, and Discrete Fourier Transform (DFT) coefficients [6-8] etc. can be used to improve the local adaptability of the representation. Among these methods, LCSCA outperforms other representations, for the least-square fitting procedure holds better fidelity to trajectories and insensibility to the variation of trajectory length [8]. Based on the representation, trajectory similarity measure is defined [9-11]. DTW tolerates temporal misalignments, so it is usually taken by lots of compromises.

On the other hand, various learning based features have attracted attentions in recent years. Existing trajectory clustering, i.e. Self-Organizing Map (SOM) [12], hierarchical fuzzy K-means [13], and trajectory modeling, i.e. Gaussian Mixture Models (GMMs) [5], hierarchical Bayesian Model [14] and Hierarchical Hidden Markov Model [15], still have the limitation of modeling trajectories incrementally to deal with the scarcity of labeled training trajectories. Recently, trajectory classification has been casted as a sparse reconstruction problem [16-19][30]. Considering the combination of the structural clustering [20] and the label consistency constraint [21-23] with dictionary, a discriminative dictionary can be learned to effectively weaken the limitation of clustering based on the trajectory centroid vectors. Moreover, locality is as important as sparsity in trajectory representation [16], so Locality-constrained Sparse Reconstruction (LSR) can be adaptive to trajectory noises, incompleteness and local variation for robust trajectory classification.

In this paper, we proposed a Cluster-based Dictionary Learning (CDL) and Locality-constrained Sparse Reconstruction (LSR) for trajectory classification in surveillance video, as in Fig.1. The framework includes two stages: learning and classifying. Given a video scenario, we firstly assign trajectories into clusters with K-means clustering based on the LCSCA feature vectors. An initial dictionary is obtained with the clustered trajectories and their corresponding labels. Then, we incorporate a label consistency constraint and optimal criteria into the objective function to learn the dictionary. As the model separately and incrementally updates the dictionary and the sparse code, the set of labels is also updated according to clustering the sparse reconstruction coefficients. In the trajectory classification stage, a multiple-category classifier for trajectory is used to estimate trajectory label based on the LSR with learned dictionary.

The rest of the paper is organized as follows. We describe the proposed CDL and LSR in section 2 and section 3. We demonstrate experimental results in section 4, and conclude the paper in section 5.

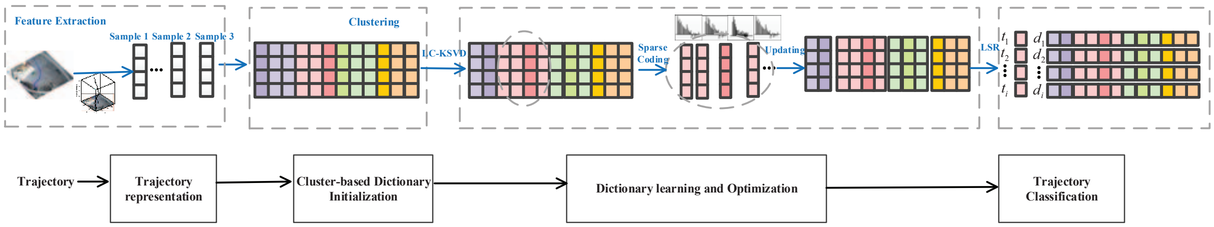

Fig.1. Overview of the proposed approach. (Better View in Color)

2. CLUSTER-BASED DICTIONARY LEARNING

In this section, we describe our CDL method in details. In the learning stage, we firstly construct the initial dictionary with the training vectors based on LCSCA and clustering labels based on K-means. Then an updating label consistent K-SVD (LC-KSVD) is used to learn an optimized dictionary for classification. During this procedure, the incremental dictionary learning is applied to update the model. When given the learned dictionary, a multi-category classifier is constructed and used for trajectories classification.

2.1. Trajectory Representation

LCSCA-based trajectory feature representation method [8] is achieved by approximating each spatial-temporal trajectory with a uniform cubic B-spline curve parameterized by time. It is more robust than other point-based distances between trajectories with variable length. Consequently, we extract LCSCA-based feature vector as trajectory representation, shown in Fig.1. Given a trajectory $t_i = \{(P_1^x, P_1^y), \cdots, (P_n^x, P_n^y)\}$, where n is the length of trajectory, (P_n^x, P_n^y) represents n-th position point of the trajectory. The control point-based feature representation of t_i is $y_i = \{(C_1^x, C_1^y), \cdots, (C_p^x, C_p^y)\}$ with the predefined B-spline basis function in [19], p is the number of control points, and (C_p^x, C_p^y) represents the p-th control point, where C_p^x and C_p^y represent its normalized x-coordinate and y-coordinate, respectively.

2.2. Cluster-based Dictionary Initialization

We firstly apply the K-means clustering [13] with DTW distance [9] to cluster trajectories, which are represented by the LCSCA control points. Given N trajectories as $T = [t_1, \cdots, t_N] \in \mathbb{R}^{p \times N}$, we get the LCSCA feature vector y_i for each trajectory t_i. K centroids are defined among N feature vectors, y_i is assigned to the cluster, the centroid c_j has the least DTW distance with y_i. When all feature vectors assigned, the positions of the K new centroids is recalculated. We repeat the re-clustering and recalculation until J reaches the local optima. Let Y be a set of p-dimensional N input feature vectors, *i.e.* $Y = [y_1, \cdots, y_N] \in \mathbb{R}^{p \times N}$. Learning a reconstructive dictionary D with M items for sparse representation of Y can be achieved by solving the following problem in Eq. (1). We initialize $D^{(0)}$ with N training vectors as Eq. (2), where $C^{(k)}$ are the clustering results, and $\sum_{k=1}^{K} N_k = N$.

$$\langle D, X \rangle = \underset{D,X}{\operatorname{argmin}} \|Y - DX\|_2^2, \quad s.t. \forall i, \|x_i\|_0 \le \varepsilon. \tag{1}$$

$$D^{(0)} = \left[C^{(1)}, \cdots, C^{(k)}, \cdots C^{(K)}\right], \quad C^{(k)} = \left[y_1^{(1)}, \cdots, y_{N_k}^{(k)}\right]. \tag{2}$$

2.3. Dictionary Learning and Optimization

There are two steps to learn the optimal D: (1) keeping D fixed to find X-sparse coding; (2) keeping X fixed to find D - SVD decompositions. Sparse coding x_i on a discriminative dictionary D can be used for classification by computing the sparse representation x_i of y_i on given D using the Orthogonal Matching Pursuit algorithm (OMP) [24]. While separating the dictionary learning from the classifier learning might make suboptimal for classification. As in [17, 25, 26], it is possible to jointly learn the dictionary and classification model by optimizing the dictionary. While inspired by the intuition that a label consistency constraint on the sparse codes makes the element in the dictionary D highly peak in one class, a method named LC-KSVD is employed for dictionary learning. With the definition in [21], the objective function is rewritten and optimized as

$$\begin{aligned} \langle D, W, A, X \rangle &= \underset{D,W,A,X}{\operatorname{argmin}} \|Y_{new} - D_{new} X\|_2^2 \\ & s.t. \forall i, \|x_i\|_0 \le \varepsilon \\ Y_{new} &= \left(Y^T \quad \sqrt{\alpha} Q^T \quad \sqrt{\beta} H^T\right)^T \\ D_{new} &= \left(D^T \quad \sqrt{\alpha} A^T \quad \sqrt{\beta} W^T\right)^T \end{aligned} \tag{3}$$

where A and W is related to discriminative sparse code error and classification error, respectively. α and β are useful for controlling the relative contribution between reconstruction and label consistence regularization.

We also apply the same incremental framework to update the model as training vectors come. During the incremental dictionary optimization, the model separately and incrementally updates the dictionary and sparse code by LC-KSVD [21], the set of labels is also updated according to clustering the sparse reconstruction coefficients with respect to the dictionary.

3. LOCALITY-CONSTRAINED SPARSE RECONSTRUCTION

In the trajectory classification stage, a multiple-category classifier for trajectory is used to classify a trajectory based on the LSR with learned dictionary. LSR approach is presented by author in [16] and describes how to partition the trajectory, and how to perform sparse linear reconstruction with locality-constrained dictionary.

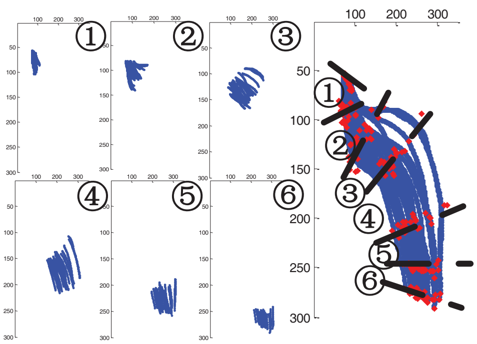

Fig.2. Examples of a class of similar trajectories and their partitioned tracklets. Left: six partitioned tracklets from a class of similar trajectories (blue curves). Right: the whole trajectories with control points (red dots).

The objective of trajectory partition is to divide a long trajectory into tracklets, shown in Fig.2. Based on the fact that trajectories of a same category should often have similar shapes and share control points on the cubic B-spline curves. We partition the trajectories into local tracklets (local shapes) based on the control points and then align the tracklets via the DTW algorithm to construct a local dictionary [16] as

$$D=\begin{bmatrix} d_1 & \cdots & d_i & \cdots & d_{p-1}\end{bmatrix}^T,\ i=1,\ldots,p-1. \tag{4}$$

where $d_i = \{a_1^1(i), a_1^2(i), \ldots, a_1^K(i), \ldots, a_J^K(i)\}$ represents the i-th tracklets of J kinds trajectories in a scene after partition, each class of trajectory holds K tracks. Accordingly, a long feature vector is divided into a set of short sub-vectors, each of which will be better represented by a linear model.

After partition, each tracklet t_i of T can be approximately represented as a linear superposition of the d_i just in the D as $t_i \approx d_i\psi_i$, where ψ_i represents the coefficient vector for superposition. Because so many trajectory routes in a scene, the dictionary is often large, a tracklet can be represented with as few as tracklets of local similarity, and ψ_i should be sparse.

Then, a discriminate encoding and a loss weighted decoding strategy for classification is further proposed. Tracklets represent local shapes of trajectories, however, they loss the global information trajectories. Therefore, a combination of reconstruction results from multiple tracklets is required for accurately classifying a trajectory. We propose to use a discriminate code matrix M to perform multi-class trajectory classification, element M_{ij} that corresponds to assign the i-th tracklet to the j-th class is calculated as Eq. (5). With the code matrix, a trajectory is finally classified by using a loss-based decoding strategy. The objective is to find a matrix that weights a loss function and adjusts the decisions of the sparse reconstruction. The loss weighted decoding process is described in [16].

$$M_{ij}=\begin{cases}0, & if\ j\neq \underset{j}{argmin}(\varepsilon_{ij})\\ 1, & if\ j=\underset{j}{argmin}(\varepsilon_{ij})\end{cases},\quad \begin{array}{c}\varepsilon_{ij}(t_i)=\left\|t_i-d_i\delta_j(\psi_i)\right\|_2\\ j=1,\ldots,N_c\end{array}. \tag{5}$$

4. EXPERIMENTAL RESULTS

We evaluate our approach on two trajectory datasets: the CAVIAR ("INRIA") and the Carpark. And we compare our approaches with Hierarchical K-Means (HKM) [13], Parallel Spectral Cluster (PSC) [29], GMMs [5].

4.1. Datasets

The CAVIAR dataset [27] contains a series of trajectories in an entrance lobby with 11 entry-exit routes. The Carpark dataset [28] contains 269 trajectories with 8 categories of trajectories in three crossroads. Considering the traversal orientations, we randomly select half of the trajectories per category as training and the other half for testing, so we have 1100 trajectories in the training set and 1121 trajectories in the testing set of CAVIAR, and 124 training trajectories and 145 testing trajectories in the Carpark.

4.2. Performance and comparison

In the CAVIAR dataset, we use the LCSCA with seven control points to represent trajectories, and DTW distance to measure the similarity between trajectories. In the experiments, α and β are respectively set to 0.01 and 1 to learn the dictionary with 110 items (M=110). We evaluate the classification ability of our approach compared with three methods using 5, 15, 25, 35, 45, and 50 training samples per category, respectively. The experimental results are summarized in Table 1. Our approach consistently outperforms other approaches. The essential reason for the good classification performance, even with only a few training samples, is that the updating label consistency constraint ensures the input vectors from the same class always have similar sparse codes, even with a small training sample set. Moreover, we show some samples of classified categories in Fig.3. The samples in a row are the results of a method, and the samples in a column are the trajectories classified into a same cluster. There are well classified to cluster 1, while the cluster 4, cluster 9 and cluster 19 have some misclassification. The samples from cluster 9 and cluster 19 have similar positions and shapes, so they are prone to be misclassified. The results demonstrate our approach performs more effectively than others, for our approach have more discrimination ability among clusters.

Table 1. Comparisons of classification accuracy (%) in the CAVIAR dataset.

Method	Number of training samples per class					
	5	15	25	35	45	50
HKM [13]	19.8	28.4	35.9	36.7	38.9	45.9
PSC [29]	21.9	23.6	37.1	38.6	47.4	68.3
GMMs [5]	-	-	-	-	-	38.6
Our approach	**38.7**	**58.4**	**65.5**	**67.3**	**70.1**	**72.3**

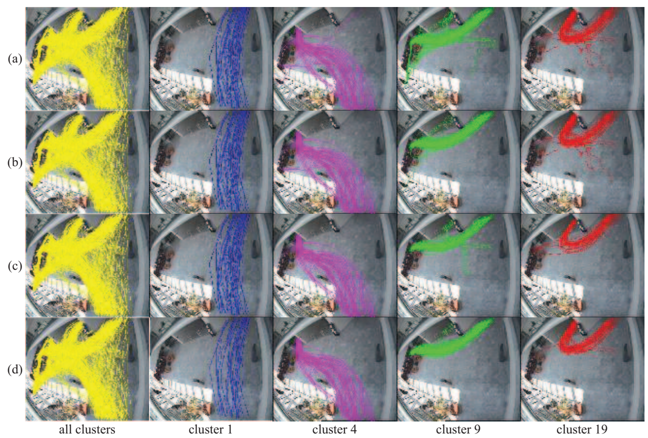

Fig.3. Examples of classified testing trajectories in the CAVIAR dataset. The results of: (a) HKM, (b) PSC, (c) GMMs, (d) our proposed approach. (Better View in Color)

Different from the experimental settings in CAVIAR, we use the LCSCA-based trajectory feature representation with eleven control points in Carpark. α and β are set to 1 in the experiment. We also evaluate the classification ability of our approach compared with three methods, where a test trajectory is classified by the most similar cluster. We use 5, 10, and 15 training samples per category, respectively. The experimental results are summarized in Table 2. It is shown that our approach consistently outperforms other competitive methods. Moreover, we show some samples of classified categories in Fig.4. The trajectories are well classified to cluster 1, while the cluster 3, cluster 6 and cluster 7 have some misclassified trajectories, for the purple (cluster 3), green (cluster 6) and red (cluster 7) trajectories pass by the crossroad. All methods are difficult to identify the purple trajectories from east to west-south and from west to west-south and from west to west-south, because both of them have long segments with similar positions and shapes. Nevertheless, the results show that the proposed approach outperforms the other methods in cluster 3, cluster 6 and cluster 7.

Table 2. Comparisons of classification accuracy (%) in the Carpark dataset.

Method	Number of training samples per class		
	5	10	15
HKM [13]	17.9	33.7	37.5
PSC [29]	18.5	36.9	45.8
GMMs [5]	-	-	38.4
Our approach	**38.1**	**66.7**	**69.2**

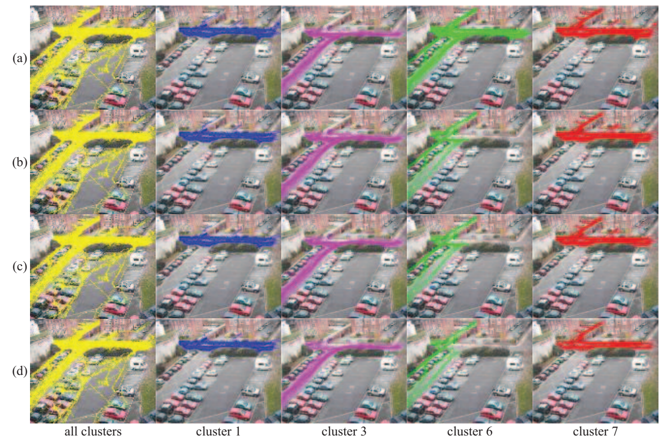

Fig.4. Examples of classified testing trajectories in the Carpark dataset. The results of: (a) HKM, (b) PSC, (c) GMMs, (d) our proposed approach. (Better View in Color)

5. CONCLUSION

We have proposed the approach of Cluster-based Dictionary Learning (CDL) and Locality-constrained Sparse Reconstruction (LSR) to classify the trajectories in surveillance videos. By introducing label consistency constraint and label updating strategy in the dictionary, the incremental CDL approach can learn the dictionary that explores the importance of the label consistency constraint and classification optimization. On the learned dictionary, we obtain a multiple-category classifier based on LSR that explores both sparsity and local adaptability for robust trajectory classification. Experimental results on two public datasets show the good performance of our approach.

Future work includes exploring better optimization algorithms to train the automatic parameter setting.

6. ACKNOLEGEMENTS

The authors would like to thank anonymous reviewers and Prof. Jianbin Jiao, A.P. Qixiang Ye, and A.P. Zhenjun Han of UCAS for helpful comments and discussions.

7. REFERENCES

[1] D. Weinland, R. Ronfard, and E. Boyer, “A survey of vision-based methods for action representation, segmentation and recognition,” *CVIU*, vol. 115, no. 2, pp. 224-241, 2011.

[2] J.C. Nascimento, J.S. Marques, and J.M. Lemos. “Modeling and classifying human activities from trajectories using a class of space-varying parametric motion fields,” *TIP*, vol. 22, no. 5, pp. 2066-2080, 2013.

[3] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz, “Robust real-time unusual event detection using multiple fixed-location monitors,” *TPAMI*, vol. 30, no. 3, pp. 555-560, 2008.

[4] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, “Anomaly detection in crowded scenes,” in *CVPR*, 2010, pp. 1975-1981.

[5] R.R. Sillito, and R.B. Fisher, “Semi-supervised learning for anomalous trajectory detection,” in *BMVC*, 2008, pp. 1035-1044.

[6] W. Hu, X. Li, G. Tian, S. Maybank, and Z. Zhang, “An incremental DPMM-based method for trajectory clustering, modeling, and retrieva,” *TPAMI*, vol. 35, no. 5, pp. 1051-1065, 2013.

[7] A. Naftel, S. Khalid, “Classifying spatiotemporal object trajectories using unsupervised learning in the coefficient feature space,” *Multimedia Systems*, vol. 12, no. 3, pp. 227-238, 2006.

[8] R.R. Sillito, and R.B. Fisher, “Parametric trajectory representations for behavior classification,” *Multimedia Systems*, vol. 12, pp.227-238, 2006.

[9] L. Chen, M.T. Özsu, and V. Oria, “Robust and fast similarity search for moving object trajectories,” *in Proc. of the 2005 ACM SIGMOD international conference on management of data*, 2005, pp. 491-502.

[10] F.I. Bashir, A.A. Khokhar, and D. Schonfeld, “Real-time motion trajectory-based indexing and retrieval of video sequences,” *IEEE Transactions on Multimedia*, vol. 9, pp. 58-65, 2007.

[11] M. Vlachos, G. Kollios, and D. Gunopulos, “Discovering similar multidimensional trajectories,” *in Proc. of IEEE International Conference on Data Engineering*, 2002, pp. 673-684.

[12] B.T. Morris, and M.M. Trivedi, “A survey of vision-based trajectory learning and analysis for surveillance,” *TCSVT*, vol. 18, no. 8, pp.1114-1127, 2008.

[13] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, and S. Maybank, “A system for learning statistical motion patterns,” *TPAMI*, vol. 28, no. 9, pp. 1450-1464, 2006.

[14] X. Wang, K.T. Ma, G.W. Ng, and W.E.L. Grimson, “Trajectory analysis and semantic region modeling using nonparametric hierarchical bayesian models,” *IJCV*, vol. 95, no. 3, pp. 287–312, 2011.

[15] D. Vasquez, T. Fraichard, and C. Laugier, “Growing hidden markov models: an incremental tool for learning and predicting human and vehicle motion,” *The International Journal of Robotics Research*, 2009.

[16] C. Li, Z. Han, S. Gao, L. Pang, Q. Ye, and J. Jiao, “Locality-constrained Sparse Reconstruction for Trajectory Classification,” in *ICPR*, 2014, pp. 2602-2606.

[17] Q. Zhang, and B. Li, “Discriminative k-svd for dictionary learning in face recognition,” in *CVPR*, 2010, pp. 2691-2698.

[18] Y. Cong, J. Yuan, J. Liu, “Sparse reconstruction cost for abnormal event detection,” in *CVPR*, 2011, pp. 3449-3456.

[19] C. Li, Z. Han, Q. Ye, J. Jiao, “Visual abnormal behavior detection based on trajectory sparse reconstruction analysis,” *Neurocomputing*, vol. 119, pp. 94-100, 2013.

[20] W. Dong, X. Li, D. Zhang, and G. Shi, “Sparsity-based image denoising via dictionary learning and structural clustering,” in *CVPR*, 2011, pp.457-464.

[21] Z. Jiang, Z. Lin, and L.S. Davis, “Label consistent k-svd: learning a discriminative dictionary for recognition,” *TPAMI*, vol. 35, no. 11, pp. 2651-2664, 2013.

[22] R. Rubinstein, T. Peleg, and M. Elad, “Analysis k-svd: A dictionary-learning algorithm for the analysis sparse model,” *IEEE Transactions on Signal Processing*, vol. 61, no. 3, pp. 661-677, 2013.

[23] J.A. Tropp, and A.C. Gilbert, “Signal recovery from random measurements via orthogonal matching pursuit,” *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655-4666, 2007.

[24] D.S. Pham, and S. Venkatesh, “Joint learning and dictionary construction for pattern recognition,” in *CVPR*, 2008, pp. 1-8.

[25] J. Mairal, F. Bach, and J. Ponce, “Task-driven dictionary learning,” *TPAMI*, vol. 34, no. 4, pp. 791-804, 2012.

[26] D.M. Bradley, and J.A. Bagnell, “Differential sparse coding,” *Conf. Neural Information Processing Systems*, 2008.

[27] CAVIAR: http://homepages.inf.ed.ac.uk/rbf/CAVIAR

[28] H.M. Dee, and D. Hogg, “Detecting inexplicable behavior,” in *BMVC*, 2004, pp.1-10.

[29] W.Y. Chen, Y. Song, H. Bai, C.J. Lin, and E.Y. Chang, “Parallel spectral clustering in distributed systems,” *TPAMI*, vol. 33, no. 3, pp. 568-586, 2011.

[30] Z. Han, J. Jiao, B. Zhang, Q. Ye, and J. Liu, "Visual object tracking via sample-based adaptive sparse representation (AdaSR)," *Pattern Recognition*, no.44, pp. 2170–2183, 2011.