Abstract—This paper considers an unsupervised learning algorithm that can automatically discover key behavior patterns (bPs) to characterize a complex video scene. For behavior features (bFs) extracted at multiple spatial-temporal scales, an optimization problem is formulated to cluster bFs in their scales while establishing a collaborative nonlinear relationship in the form of a kernel regression function among clustered bFs across different spatial scales. This relationship allows features extracted in one scale to be considered as contextual information in the analysis of another scale. This optimization problem is solved using linear programming to reduce computational complexity. The proposed algorithm is evaluated on four crowded traffic scenes and two sports video datasets. Experimental results show that the proposed algorithm achieves better performance compared to current state-of-the-art algorithms in terms of video segmentation accuracy.

Index Terms—Complex video scene analysis, Kernelized collaborative pattern learning, Temporal video segmentation

I. INTRODUCTION

The task of complex video scene analysis (VSA), often referred to as discovering object behavior patterns (bPs), involves eliciting representative bP present in a complex scene amid noise and occlusion. With the advancement in information technology, this task can be automated based on learning, and the current research trend attempts to perform this learning task without resorting to a large volume of expensive labeled data.

Various frameworks for VSA have been proposed based on machine learning techniques: probabilistic topic models (PTMs) [1]–[6], hierarchical models [1], [4], [6]–[9], multi-scale models [5], [10], convex optimization [11] and sparse matrix factorization [12]. Their performances on experimental data are encouraging but still leaves much to be desired. One main reason might be attributed to the difficulty in determining an appropriate spatial-temporal scale at which to perform the analysis. Collection of features extracted from large spatial-temporal regions, which tend to be heterogeneous, may carry information difficult to decode while that from small spatio-temporal regions may carry no significant information. To avoid making difficult decision on scale selection, many VSA frameworks perform multi-scale analyses. However, without a collaborative and collective effort in performing a consolidated analysis, a sensible bP prediction that makes sense at all scales is not guaranteed.

This paper tries to address this issue by extracting features at multiple scales, but rather than performing independent analyses across different scales, a kernel function is incorporated to form a collaborative hierarchical nonlinear relationship among the bPs across different spatial scales while collectively discovering a bP that can represent the video clip. The algorithm referred to as kernelized collaborative behavior pattern learning (K-CBPL) is considered to automatically discover representative bPs in an unsupervised manner. In order to meet its objective, a convex optimization problem is formulated to discover the bPs. The optimization problem is solved with...
low computational complexity by formulating the problem as a linear programming (LP) problem. The intuition behind the use of a nonlinear hierarchical relationship is to relate patterns across different scales in such a way that they collaboratively and collectively assist one another in discovering the representative bP: a feature extracted in one scale provides contextual information for bP analysis in other scales.

Although bF extraction at multiple scales has been previously considered, the explicit relationship among bF spaces across different scales was never formulated, and therefore, previous algorithms could not make use of the available contextual information. A kernel regression (KR) function [13] is employed to model the relationship among clusters of bFs across different scales. The proposed algorithm is evaluated on four crowded traffic scenes and two sports video datasets, and the result is given in terms of how well video is segmented temporally as shown in Fig. 1. The advantage of considering the proposed hierarchical relationship and collaborative behavior pattern learning is reflected in all experiments conducted.

The rest of this paper is organized as follows. Section II reviews various related literatures. Section III describes preliminaries and gives an overview of the proposed algorithm. Section IV describes the proposed K-CBPL algorithm in detail. Section V presents experimental results comparing and analyzing various state-of-the-art algorithms. Finally, Section VI concludes the paper.

II. RELATED WORKS

Various algorithms for VSA have been proposed in the last decade, and these algorithms can be classified depending on whether or not object-tracking features are used. There are advantages and disadvantages to consider.

In the object-tracking feature based algorithms [14]–[19], moving objects are detected from foreground, and their trajectories are extracted as bFs in order to discover the representative bPs. These algorithms focus on constructing bP models based on individual object movements over time and space, and the algorithms proposed under this category are as follows: hierarchical trajectory learning [14], long-term observation based semantic scene model [15], statistical trajectory features and Hidden Markov Models (HMM) [16], robust motion tracking algorithm [17], bag-of-particle trajectories [18], and multi-camera networks based joint behavior learning model [19]. These algorithms are all susceptible to tracking error that are often caused by occlusion and noise.

Recently, VSA algorithms that are based on non-tracking features [6]–[8], [20]–[22] have shown promising results and are gaining increased popularity. Without tracking individual object movements, BoW-like features, which implicitly reflect the dependencies of object bFs as histogram of feature co-occurrences, are extracted and applied to a probabilistic framework based either on HMM to model temporal dependency of bP or on PTM which does not but considers only the overall frequency. It should be reminded that probabilistic frameworks require considerable amount of data for training. Some notable HMM-based algorithms are as follows: Markov Clustering Topic Model [7], Dependent Dirichlet Processes (DDP)-HMM [8], multivariate nonparametric probability density function [20], sequential relationships based unified model selection [21], and Multi-Observation-Mixture+Counter-HMM [22]. Unfortunately, these algorithms that model temporal dependency require a well-defined relationship among the object’s bPs with pre-specified rules to capture the temporal order of object’s bPs wherein the relationship is often represented by a complex hierarchical structure, which levies high computational cost during inference.

In order to overcome some of the weaknesses involved in the use of object-tracking features and temporal modeling, as mentioned above, algorithms based on PTM such as probabilistic latent semantic analysis (pLSA) and Latent Dirichlet Allocation (LDA) [1]–[6], [9]–[12] using non-tracking BoW-like features, have also been proposed for predicting bPs that can represent a video clip. However, the appropriate spatio-temporal scale at which the bFs are extracted has always been difficult to determine and so critical to the end result. Features extracted from a large, spatial-temporal region, which is likely to be heterogeneous, may carry inexplicable information while features extracted from small, spatial-temporal region may just carry information about noise.

In this paper, we employ a multi-scale BoW-like feature but do not rely on any PTMs for predicting the representative bPs. Instead, we model the task of discovering representative bPs using the K-CBPL algorithm, which is described in detail in Section IV. The proposed K-CBPL algorithm is partially motivated by [11], where a convex optimization with earth mover’s distance (EMD) based objective function is considered. Here, the EMD is used to measure the cross-bin distance between two BoW features. In contrast to [11], the proposed K-CBPL algorithm explicitly focuses on discovering representative bPs by adopting a hierarchical relationship among clusters of bFs at different scales with a kernel regression function while trying to reduce the computational complexity of the algorithm as much as possible via the use of linear programming. The incorporation of the nonlinear hierarchical relationship provides contextual information across different scales by interrelating the different bF spaces, and, as an overall effect, information from different scales are used collectively and collaboratively in discovering the representative bP.

III. PRELIMINARIES

In this paper, the representative bP of a video clip is discovered in three stages: (1) atomic-event extraction, (2) behavior feature construction and (3) kernelized-collaborative pattern learning as shown in Fig. 2. For a given input video, dense optical flows of object movements are extracted at multiple spatial-temporal scales from the foreground pixels which are obtained after performing background subtraction. Henceforth, without loss of generality and for the sake of simplicity, only two spatial-temporal scales of small and large will be considered. From densely sampled optical flows, small and large-scale atomic events which are optical flow descriptors defined in the following section, are clustered into “dominant atomic events” in their respective scales. Object bFs which are constructed as histograms of the dominant atomic events are obtained from each video clip as shown in Fig. 3. Based on bFs
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2016.2539540, IEEE Transactions on Circuits and Systems for Video Technology

Fig. 2. The overall process of the proposed K-CBPL algorithm to discover representative behavior patterns for complex video scene analysis. (vertical representative bPs (blue) and horizontal representative bPs (green)).

Fig. 3. The example of the extracting atomic events and constructing behavior features in small-scale behavior feature domain.

A. Atomic Events Extraction

For capturing local behavior information of objects, atomic events are extracted from various object movements in a scene. Given a video \( V \) consisting of many frames, it is divided into non-overlapping spatial-temporal grid with size \( \Delta_S \times \Delta_S \times \Delta_T \) and \( \Delta_L \times \Delta_L \times \Delta_T \) for extracting a set of small and large-scale atomic events \( E_S \) and \( E_L \) respectively. Here, \( \Delta_S \) and \( \Delta_L \) are the width and height of small-scale spatial grid size (pixels) while \( \Delta_T \) is the small-scale temporal grid size (frames). Large-scale spatial-temporal grids \( \Delta_L \) are similarly defined. Both sets of atomic events \( E_S \) and \( E_L \) are constructed using 5-dimensional features which represent the location of the spatial-temporal grids, direction and magnitude of optical flows, and object foreground ratio. As a side note, prior to obtaining visual features using the Lucas-Kanade algorithm, background subtraction is performed. Finally, \( E_S \) is clustered into \( D_S \) number of dominant small-scale atomic events using K-medoids clustering [23] with a Kanade algorithm, background subtraction is performed. For capturing local behavior information of objects, atomic events are extracted from various object movements in a scene.

B. Multi-scale Behavior Feature Construction

In this paper, object bF is represented as occurrence or histogram of dominant atomic events (HOA). Given \( V \), it is divided into \( N \) number of small video clips \( \{ V_i \}_{i=1}^N \), and small and large-scale bF pairs \( (h_i^S, h_i^L) \in \mathbb{R}^{D_S \times 1} \times \mathbb{R}^{D_L \times 1} \) are obtained as histograms of dominant small and large-scale events of \( V_i \) for \( i = 1, \ldots, N \).

IV. COLLABORATIVE BEHAVIOR PATTERN LEARNING

Given \( H = \{(h_i^S, h_i^L)\}_{i=1}^N \), the CBPL discovers the representative small and large-scale behavior pattern (bP) pairs \( P = \{(p_i^S, p_i^L)\}_{i=1}^N \) while trying to satisfy the hierarchical relationship between the two bPs. In fact, \( p_i^S \in \mathbb{R}^{D_S \times 1} \) and \( p_i^L \in \mathbb{R}^{D_L \times 1} \) are respectively the small and large-scale clustered bFs, and each bF is assigned to the closest representative bP of its scale. In the following section, collaborative behavior pattern learning (CBPL) algorithm is used to determine both linear and nonlinear hierarchical relationship between small and large-scale bFs. The nonlinear relationship is formulated as a kernel regression function.

A. Collaborative behavior pattern learning using linear relationship

As described above, the bP learning algorithm can be formulated as the following optimization problem:

\[
\min_{P} \Phi(H, P) + \lambda^R \Psi(P),
\]

\[
\text{s.t. } p_i^S, p_i^L \geq 0, \sum_{q=1}^{D_S} p_{i,q}^S = 1, \sum_{u=1}^{D_L} p_{i,u}^L = 1, \forall i, j = 1, \ldots, N,
\]

where

\[
\Phi(H, P) = \sum_{i=1}^{N} d_i \left( \begin{bmatrix} h_i^S \\ h_i^L \end{bmatrix}, \begin{bmatrix} p_i^S \\ p_i^L \end{bmatrix} \right),
\]

\[
\Psi(P) = \sum_{i,j, i \neq j} \eta_{ij} d_2 \left( \begin{bmatrix} p_i^S \\ p_i^L \end{bmatrix}, \begin{bmatrix} p_j^S \\ p_j^L \end{bmatrix} \right).
\]

Here, \( d_i(a, b) \) for \( i = 1, 2 \) indicates a pre-defined distance function between \( a \) and \( b \). In this paper, the EMD and Manhattan distance (L1 distance) are considered. The first loss function \( \Phi(\cdot) \) in Eq.(1) enforces \( P \) to be close to \( H \). The second function \( \Psi(\cdot) \) which acts as a regularizer enforces smoothness among related bPs such that the difference between the \( i^{th} \) and \( j^{th} \) bPs is minimized. The L1 regularization is known to induce sparsity which leads to a small number of bPs. To group the bPs explicitly, bPs that are within a certain distance in the L1 sense are sequentially grouped as in [24] (for our purpose \( 10^{-4} \) was used). Here \( \lambda^R \) controls the strength of the regularization. When \( \lambda^R = 0 \), \( p_i^S \) can
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2016.2539540, IEEE Transactions on Circuits and Systems for Video Technology

Algorithm 1 Optimization procedure for CBPL using linear relationship.

**Input:**

\( H = \{ (h_{i}^{S}, h_{i}^{L}) \}_{i=1}^{N} \in \mathbb{R}^{D_{S} \times N} \times \mathbb{R}^{D_{L} \times N} \): a set of small and large-scale bFs, 
\( \lambda^{R} \): regularization constant, MaxIter : the maximum iterations.

**Output:**

Optimized \( P_{\tau}^{S} \in \mathbb{R}^{D_{S} \times N} \): a final set of representative bPs, 
Optimized \( W_{\tau} \in \mathbb{R}^{D_{L} \times D_{L}} \): a linear transformation matrix.

**Procedure:**

1. set \( \tau = 0 \),
2. initialize \( P_{\tau=0}^{S} \) and calculate \( C_{\tau=0} \) (the objective function value) using following optimization problem:
   \[
   \min_{P_{\tau}^{S}} \sum_{i=1}^{N} d(h_{i}^{S}, P_{\tau}^{S}).
   \]
3. repeat
   4. **Step 1**: calculate \( W \) with fixed \( \hat{P}_{\tau}^{S} \) using following optimization problem according to Eq.(2):
      \[
      (\text{set} \lambda^{R} = 0 \text{ and discard } d_{1}(H^{S}, P_{\tau}^{S}) \text{ in Eq.(2)})
      \min_{W} \sum_{i=1}^{N} d_{1}(h_{i}^{S}, W^{T} \hat{P}_{\tau}^{S}) + \lambda^{R} \sum_{i=1}^{D_{S}} ||w_{i}||, \]
   5. **Step 2**: calculate \( \hat{P}_{\tau}^{S} \) and \( \hat{C} \) with fixed \( W \) and \( \lambda^{R} \) using following optimization problem according to Eq.(2):
      \[
      (\text{set} \lambda^{W} = 0 \text{ in Eq.(2) } \)
      \min_{P_{\tau}^{S}} \sum_{i=1}^{N} d_{1}(h_{i}^{S}, P_{\tau}^{S}) + d_{1}(h_{i}^{L}, W^{T} \hat{P}_{\tau}^{S}) \]
      \[
      + \lambda^{R} \sum_{i \neq j} \eta_{ij} d_{2}(P_{\tau}^{S}, P_{\tau}^{S}) + d_{2}(W^{T} \hat{P}_{\tau}^{S}, W^{T} \hat{P}_{\tau}^{S}) \]
   6. Grouping \( \hat{P}_{\tau}^{S} \) using sequential clustering scheme based on \( L_{1} \) distance between the \( i^{th} \) and \( j^{th} \) bPs.
7. Update \( P_{\tau+1}^{S} \leftarrow \hat{P}_{\tau}^{S} \), \( W_{\tau} \leftarrow W \), \( C_{\tau} \leftarrow \hat{C} \).
8. \( \tau \leftarrow \tau + 1 \).
9. until Convergence: \( |C_{\tau+1} - C_{\tau}| \leq 10^{-2} \& \tau \leq \text{MaxIter} \)
10. Return \( P_{\tau_{\text{opt}}}^{S} \), \( W_{\tau_{\text{opt}}} \).

**Fig. 4.** The example of linear relationship between multi-scale bPs.

take on \( N \) different representative bPs, while \( \lambda^{R} \rightarrow \infty \), \( \hat{P}_{\tau}^{S} \) can take only one representative bP (all bPs are grouped as one representative bP). Essentially, different \( \lambda^{R} \) will lead to different \( K (1 \leq K \leq N) \) number of representative bPs. In the experiment, \( \lambda^{R} \) is initialized to 0 then increased until the number of discovered representative bP reaches \( K \) as in [11], [31]. Also, \( \eta_{ij} \) indicates whether \( P_{\tau}^{S} \) and \( \hat{P}_{\tau}^{S} \) should be merged based on the similarity between \( h_{i}^{S} \) and \( h_{j}^{S} \), in terms of clustering bPs. For computational efficiency, the \( \eta_{ij} \) are pre-determined as a set of \( N_{p} \subseteq N - 1 \) non-zero nearest-neighbor pairs of \( h_{i}^{S} \); thus, \( \eta_{ij} = 1 \) when \( h_{j}^{S} \) is in the nearest-neighbor of \( h_{i}^{S} \) based on the EMD.

Firstly, a simple linear hierarchical relationship between the small and large-scale bPs is considered as shown in Fig. 4. For understanding the relationship between small and large-scale bPs, the following examples are considered: small-scale vehicle movements can define large-scale traffic flow in a traffic scene and small-scale movement of an athlete can lead to large-scale team play in a sports scene. The bP learning algorithm of Eq.(1) can be reformulated as discovering \( \hat{P}_{\tau}^{S} \) and \( W \) from \( H \) with \( \hat{P}_{\tau}^{L} = W^{T} \hat{P}_{\tau}^{S} \). To end, this, the CBPL algorithm reformulated as the following optimization problem:

\[
\min_{P_{\tau}^{S}, W} \Phi(H, P_{\tau}^{S}) + \lambda^{R} \Psi(P_{\tau}^{S}) + \lambda^{W} \Gamma(W),
\]

\[
\text{s.t } P_{\tau}^{S} \geq 0, \quad \sum_{q=1}^{D_{S}} \hat{P}_{i,q}^{S} = 1, \quad \forall i = 1, \ldots, N,
\]

where

\[
\Phi(H, P_{\tau}^{S}) = \sum_{i=1}^{N} d_{1}(h_{i}^{S}, \hat{P}_{i}^{L}),
\]

\[
\Psi(P_{\tau}^{S}) = \sum_{i,j} \eta_{ij} d_{2}(\hat{P}_{i}^{S}, \hat{P}_{j}^{S}),
\]

\[
\Gamma(W) = \sum_{i=1}^{D_{S}} ||w_{i}||.
\]

Note that Eq.(2) is a non-convex constrained optimization problem due to the cross terms of \( W^{T} \hat{P}_{\tau}^{S} \) and \( \hat{P}_{\tau}^{S} \). For mathematical tractability, this problem is converted into two convex optimization problems which are solved iteratively based on LP. The two convex optimization problems are given by Eq.(4) and Eq.(2). After solving for \( W \) in Eq.(4), the final set of representative bPs \( P_{\tau}^{S} \) are obtained by solving Eq.(2) with the solution of \( W \) as shown in Algorithm 1.

When the hierarchical relationship is assumed linear, the problem can be solved using a simple linear transformation matrix \( W \) as mentioned above. However, a linear relationship is limited in explaining various hierarchical relationship between multi-scale object bPs, and may not be effective in the use of a more complex scene. It should be noted that bP learning algorithm of Eq.(2) is non-convex and different initial conditions and iterative optimization process will affect its performance.

B. Kernelized Collaborative behavior pattern learning using nonlinear relationship

In contrast to previous works on behavior pattern learning, this paper formulates the task of discovering representative bPs as a kernelized collaborative behavior pattern learning (K-CBPL) problem that simultaneously learns multi-scale bPs from multi-scale bFs while considering a hierarchical nonlinear relationship among multi-scale bPs. To overcome the inherent limitations dictated by the linear relationship considered above, this section learns a nonlinear relationship among multi-scale bPs, \( P_{\tau}^{S} \) and \( \hat{P}_{\tau}^{S} \), using a kernel regression (KR) function which is a non-parametric technique that predicts the
Fig. 5. The example of proposed K-CBPL algorithm and temporal video segmentation result using two types of discovered representative behavior patterns.

The label \( y \) of a given data \( x \) as the weighted sum of the labels of other data. Here, the weights are assigned according to the distance defined by a kernel between \( x \) and the other data. Given data and corresponding label instances \((x, y_j)\), prediction on \( y \) based on \( x \) is given as

\[
\hat{y} = \sum_j y_j \frac{K_{j,j}}{\sum_j K_{j,j}},
\]

where \( K_{j,j} = K(x, x_j) \) is referred as a kernel function (e.g., RBF-kernel \( K_{RBF}(x, x_j) = \exp(-||x - x_j||^2/2\sigma^2) \)) and the value of the kernel function can be determined based on the distance between \( x \) and its corresponding neighbor \( x_j \).

This paper assumes a nonlinear relationship between small and large-scale bPs in the form the kernel distance function, and it is incorporated into the optimization problem discussed above. The \( i \)-th large-scale bP \( p_i^L \) associated with the large-scale bP \( p_j^L \) is given as a weighted average of all small-scale bPs associated with all other large-scale bPs except \( p_j^L \). Here, the weight is defined by a kernel function \( K(\cdot) \) which measures the similarity between bPs. Thus, prediction on \( p_i^S \) based on \( p_j^L \) using kernelized collaborative function \( f^S(\cdot) \) can be defined as:

\[
f^S(p_j^L) = \sum_{j \neq i} \frac{p_j^S K(p_i^L, p_j^L)}{\sum_{j \neq i} K(p_i^L, p_j^L)}.
\]

Similarly, prediction on \( p_i^L \) based on \( p_j^S \) using \( f^L(\cdot) \) can be defined as:

\[
f^L(p_i^S) = \sum_{j \neq i} \frac{p_j^L K(p_i^S, p_j^S)}{\sum_{j \neq i} K(p_i^S, p_j^S)}.
\]

Incorporating \( f^L(\cdot) \) and \( f^S(\cdot) \) into the optimization leads to both \( p_i^S \) and \( p_i^L \) is simultaneously affected each other by \( f^L(\cdot) \) and \( f^S(\cdot) \) while discovering both a set of representative bPs \( P^S \) and \( P^L \).

To this end, the proposed K-CBPL can be formalized as the following optimization problem:

\[
\min_{P} \Phi(H, P) + \lambda^C \Omega(P) + \lambda^R \Psi(P),
\]

s.t. \( p_i^S, p_i^L \geq 0, \sum_{q=1}^{D_S} p_i^S_q = 1, \sum_{u=1}^{D_L} p_i^L_u = 1, \forall i, q, u, \)

where

\[
\Phi(H, P) = \sum_{i=1}^{N} d_i \left( \begin{bmatrix} h_i^S \end{bmatrix}, \begin{bmatrix} p_i^S \end{bmatrix}, \begin{bmatrix} p_i^L \end{bmatrix} \right),
\]

\[
\Omega(P) = \sum_{i=1}^{N} d_2 \left( \begin{bmatrix} p_i^S \end{bmatrix}, \begin{bmatrix} f^S(p_i^L) \end{bmatrix} \right),
\]

\[
\Psi(P) = \sum_{i,j, i \neq j} \eta_{ij} d_3 \left( \begin{bmatrix} \alpha p_i^S \end{bmatrix}, \begin{bmatrix} \alpha p_j^S \end{bmatrix} \right).
\]

Here, the objective function consists of three different functions. The first loss function \( \Phi(\cdot) \) encourages both \( P^S \) and \( P^L \) to be close to \( H^S \) and \( H^L \), respectively. As a measure of distance, the EMD measure can be used for \( d_1(\cdot) \) which leads to a convex function for a single scale case. The second function \( \Omega(\cdot) \) encourages accurate prediction between target bP and prediction using bPs from another scale \((P^S, f^S(P^L))\) and \((P^L, f^L(P^S))\). The main idea of the proposed algorithm is discovering representative bPs based on collective and collaborative sharing of information provided by bFs extracted at different scales. Intuitively, bF at one scale can provide contextual information to other bF at a different scale. The third function \( \Psi(\cdot) \) enforces smoothness among related bPs within a particular scale. This \( \Psi(\cdot) \) encourages similar bPs to merge thus has a clustering effect. In the experiment, \( L_1 \) norm is adopted for the \( \Omega(\cdot) \) and \( \Psi(\cdot) \). A binary indicator \( \eta_{ij} \in \{0, 1\} \) selects bP pairs to encourage merging. The bP pairs with \( \eta_{ij} = 1 \) can be identified easily by finding pairs close to one another. The trade-off parameter \( \lambda^C \) and \( \lambda^R \) control the strength of each term. Without resorting to any collaboration (e.g., \( \lambda^C = 0 \)), Eq.(9) becomes Eq.(1). To group the bPs explicitly, we use the same sequential clustering scheme used in CBPL. When \( \lambda^R = 0 \), \( P^S \) and \( P^L \) can take on \( N \) different values while for \( \lambda^R \to \infty \) enforces the number of unique bP to be one. Here, \( \alpha \) control the importance of small-scale bPs with respect to the large-scale bPs. By scanning \( \lambda^C \) and \( \lambda^R \) over their respective ranges \( 0 \leq \lambda^C, \lambda^R \leq \infty \), \( N \) to 1 number representative bPs can be obtained.

C. Convex Optimization for K-CBPL with EMD and KR

To evaluate the similarity between two histograms, conventional bin-to-bin distance measure such as \( L_p \)-norm distance
are appropriate when the domains of the two histograms are well aligned. When this is not the case, EMD [33] can be used to determine the cross-bin distance between two histograms. The EMD is defined as the minimal cost that must be paid to transform one histogram into another. The EMD between two histograms is appropriate when the domains of the two histograms are well aligned. When this is not the case, EMD [33] can be used to determine the cross-bin distance between two histograms. The EMD is defined as the minimal cost that must be paid to transform one histogram into another. The EMD between two histograms.

The above convex optimization problem cannot be solved and large-scale feature histograms as mentioned in Eq.(10), \( \gamma \).

Here, the variable \( \gamma_{qu} \) indicates a flow representing the amount transported from the \( q^{th} \) element of histogram \( h_1 \) to the \( u^{th} \) element of histogram \( h_2 \) and \( d_{qu} \) is the ground distance between \( h_1 \) and \( h_2 \) (e.g., \( d_{qu} = |h_{1,q} - h_{2,u}| \)). However, in the case of high dimensional histograms, solving Eq.(10) is computationally expensive with a large number of flow variables. Several methods have been proposed to accelerate the EMD computational cost of \( O(D^2) \) to \( O(D) \) where \( D \) is the dimension of histograms. As proposed in [34], every possible positive flows between faraway histogram bins can be replaced by sequences of flows between adjacent bins to calculate EMD with \( L_1 \)-norm (EMD-\( L_1 \)) as ground distance. This implies that Eq.(10) can be simplified as follows:

\[
\min_{\gamma_{qu}} \sum_{q=1}^{D} \sum_{u=1}^{D} d_{qu} \gamma_{qu}, \quad \text{s.t.} \quad \sum_{q=1}^{D} \gamma_{h_1,q} = 1, \sum_{u=1}^{D} \gamma_{h_2,u} = 1. \tag{10}
\]

Minimize \( \sum_{q=1}^{D} \sum_{u=1}^{D} d_{qu} \gamma_{qu} \) subject to \( \sum_{q=1}^{D} \gamma_{h_1,q} = 1, \sum_{u=1}^{D} \gamma_{h_2,u} = 1 \).

Algorithm 2 Optimization procedure for K-CBPL using non-linear relationship.

**Input:**
- \( H = \{ (h_1^S, h_1^T) \}_{i=1}^N \in \mathbb{R}^{D_S \times N} \times \mathbb{R}^{D_L \times N} \) : a set of small and large-scale bFs,
- \( \lambda^C \) : collaborative constant, \( \lambda^R \) : regularization constant, \( \delta_c \) : step size, MaxIter : the maximum iterations.

**Output:**
- Optimized \( P^S_\tau = \{ (P_1^S, P_1^T) \}_{i=1}^N \in \mathbb{R}^{D_S \times N} \times \mathbb{R}^{D_L \times N} \) : a final set of representative bPs.

**Procedure:**
1. set \( \tau = 0, P_0^S = 0, P_0^L = 0, C_0 = 0 \)
2. repeat
3. calculate \( D^S_\tau, P^L_\tau = \gamma_\tau \) and calculate \( C_\tau = 0 \) (the objective function value) according to Eq.(9).
4. Grouping \( P^S_\tau \) and \( P^L_\tau \) using sequential clustering scheme based on \( L_1 \) distance between the \( i^{th} \) and \( j^{th} \) bPs at each scale.
5. Update \( P^S_{\tau} \leftarrow \hat{P}^S_\tau, P^L_{\tau} \leftarrow \hat{P}^L_\tau, C_\tau \leftarrow \hat{C} \),
6. \( \tau \leftarrow \tau + 1 \), \( \lambda_R \leftarrow \lambda_R + \delta_c, \lambda_C \leftarrow \lambda_C + \delta_c \)
7. until Convergence: \( |C_{\tau+1} - C_\tau| \leq 10^{-2} \) & \( \tau \leq \) MaxIter
8. Return \( P^S_\tau \).

Generalized to discovering multi-scale bPs. For this, optimization problem in Eq.(9) can be extended and reformulated as the following optimization problem:

\[
\min_{P^S_\tau} \Phi^n(H^n, P^n) + \lambda^C \Omega^n(P^n) + \lambda^R \Psi^n(P^n)^\tau, \tag{13}
\]

subject to \( P_i^S, P_i^L \geq 0, \sum_{q=1}^{D_q} P_i^S, P_i^L = 1, \forall i, q, n \),

where

\[
\Phi^n(H^n, P^n) = \sum_{n=1}^{N} \sum_{i=1}^{S} \Phi_h(h^n_i, p^n_i),
\]

\[
\Omega^n(P^n) = \sum_{n=1}^{N} \sum_{i=1}^{S-1} d\left( \left[ P_i^{n+1} \right], \left[ f^n(P_i^{n+1}) \right] \right),
\]

\[
\Psi^n(P^n) = \sum_{n=1}^{N} \sum_{i=1}^{S} \eta_{i,j} \Psi(p_i^S, p_j^L).
\]

Here, the overall objective function consists of three different functions as in Eq.(9). The first loss function \( \Phi^n(\cdot) \) encourages the \( n^{th} \)-scale bPs to be close to the \( n^{th} \)-scale bFs. The second function \( \Omega^n(\cdot) \) discourages discord between adjacent bPs. The third function \( \Psi^n(\cdot) \) enforces smoothness among related object bPs in the same scale. Other parameters such as \( \eta_{ij}, \lambda^C, \lambda^R \) are also used for the same reason as mentioned above. The regularization constant \( \lambda^R \) in Eq.(9), (12) and (13) controls the different possible number of representative bP. A large value of \( \lambda^R \) leads to small possible number of representative bPs and vice versa for a small value of \( \lambda^R \). In VSA, a large value of \( \lambda^R \) leads to a concise interpretation with a small number of representative bP types while smaller \( \lambda^R \) can describe the object movements in detail. As the values of \( \lambda^R \) decreases, the degree of details is enhanced. In scene segmentation (e.g., video clustering), specific number of representative bPs is
required— which requires $K$ to be fixed to a certain number of classes— the K-CBPL can discover the desired $K$ number of representative bPs by scanning $\lambda^R$ and $\lambda^C$ across a range of values. Different values of $\lambda^R$ lead to different number of representative bPs, thus for a given application, $\lambda^R$ can be automatically scanned to discover the appropriate number of bP in accordance with the desired number of classes for segmentation.

V. EXPERIMENTS

In this section, various comparative experiments are conducted to evaluate the proposed K-CBPL algorithm in terms of video segmentation accuracy.

A. Datasets

The proposed K-CBPL algorithm is evaluated using six publicly available datasets: three from QMUL [1] dataset, two from CVBASE’06 [32] dataset and all of the MIT-traffic [2, 35] dataset. These datasets have been extensively used in previous works [1], [2], [5], [7], [8], [11], [12], [35]. The QMUL, MIT-traffic and CVBASE’06 datasets contain complex object movements from various vehicles at crossroads and athletes at sports stadiums, respectively. In the datasets, all vehicles and pedestrians in the QMUL and MIT-traffic datasets follow the regular traffic sign while the players in the CVBASE’06 follow the game rules. Examples of these datasets are shown in Fig. 6.

The test datasets consist of 39, 59 and 50 video clips for the Junction-1, Roundabout and Junction-2 in the QMUL dataset, respectively. In the MIT-traffic, 70 video clips are provided. While in the CBASE’06, 50 video clips of Basketball-A and Handball-B are provided. The aim of the proposed algorithm is to discover a set of $K$ different number of representative bPs using $N$ number of segmented video clips with pre-fixed time length. Thus, the algorithm extracts one bF histogram per video clip, and $N$ number of bFs are clustered into $K$ number of representative bP using the proposed K-CBPL algorithm. Each video clip could include multiple types of representative bPs. In our experiment, all test video clips of the benchmark datasets are assigned to one of two representative bPs (e.g., vertical or horizontal flow in load traffic). Hence, only one of the representative bPs is assigned to each segmented video clip for evaluating video segmentation accuracy. For a fair comparison between the proposed K-CBPL and the state-of-the-art algorithms, the same datasets and ground truth labels [1], [11], [32] are used to evaluate video segmentation accuracy. Additionally, we conduct multi-class temporal video segmentation with $K$ different number of representative bPs using manually labeled ground truth. The sample demonstration code for K-CBPL is available at http://slsp.kaist.ac.kr/software/Demo_KCBPL.zip.

B. Setup

For each video clip, HOA-based small and large-scale bFs are independently constructed using the same method as mentioned in Sec. III-A and Sec. III-B. The dimensionalities of small-scale bF $D_S$ and large-scale $D_L$ are set to 16 and 9, respectively. Detailed descriptions of the datasets and parameters for a two-scale bF extraction are given in Table I.

We initially set the minimum spatial-temporal grid size to $12 \times 12 \times 10$ to extract object bFs (HOA) by roughly considering the object size and spatial resolution based on parameters used in [11]. After that, we construct various set of 16-dimensional object bF histogram by gradually increasing the spatial-temporal grid size and we conduct 10 trials of video segmentation using these sets of bFs based on EMD-$L_1$ [11] algorithm to determine initial parameter size and baseline performance. The average accuracies are represented in Table II. As shown, the performance gain as a function of spatial-temporal grid size is relatively constant.

We set two different spatial-temporal grid sizes for extracting multi-scale bFs. The smallest scale would be small enough to be used for monitoring local movements such as cars and players, and the largest scale would be large enough for monitoring traffic flow and team movements. For all the experiments conducted in this paper, the grid size for extracting large-scale features is considered to be larger than that for extracting small-scale features. Thus, the grid size for obtaining large-scale bFs $H_L$ is set to be double the spatial size of that of the small-scale bFs $H_S$ and triple the temporal size of that of small-scale bFs, as shown in Table I.

The K-CBPL is valid for more than two scales. We conducted video segmentation experiment using three-scale bFs on QMUL datasets (in Table III). For this, we use 9, 16 and 25 dimensional HOA. In terms of complexity and performance gain trade-off, two-scale grid gives better trade-off: three-scale grid does not offer significant performance gain over two-scale but large complexity increase. Using both cases of two and three-scale bFs showed better...
three-scale multi-scale optical flows are independently extracted using an algorithm, which provides best trade-off between performance and computational complexity.

The proposed K-CBPL algorithm is implemented in MATLAB using GLPK (GNU linear programming kit), while multi-scale optical flows are independently extracted using an OpenCV library. For all video clips, we extracted the maximum number of nearest-neighborhood \(N_p\) to calculate \(n_{ij}\) as set to 3. For each \(i\), \(n_{ij}\) takes non-zero value for maximum of 3 values of \(j\). To evaluate benchmark video segmentation accuracy, the desired number of representative bPs, \(K\), is set in accordance with ground truth label. For multi-class segmentation, we set \(K=3\), and 5 in accordance with manually assigned ground truth label.

C. Comparison with different behavior feature representation

In this subsection, the details of our implementation are discussed. The current state-of-the-art performance in the VSA task is based on BoW feature representation, using local feature descriptors. For VSA, interest point based local feature descriptors and densely sampled patch based feature representations have been the two most popular methods for video feature representation. Among the existing feature descriptors for VSA, such as action recognition, the combination of HOG (Histograms of Oriented Gradients) [26] and HOF (Histograms of Optical Flow) [27] has achieved excellent results on a variety of datasets [27], [28]. The HOG focuses on the static appearance information, whereas the HOF captures the local motion information. Additionally, motion boundary histograms (MBH) [29], which rely on differential optical flow are introduced for the problem of human detection in video. In [30], STIP (Spatial-Temporal Interest Point) is introduced by extending the Harris detector.

In this paper, the densely sampled HOG, HOF, MBH, MBHy and STIP based BoW feature representations are used for comparison purposes. For BoW representation of object bFs, we construct a codebook for each feature descriptor (HOG, HOF, MBH, MBHy and STIP) separately and using the K-means clustering algorithm. For this, we extract descriptors on two different scales where one block consists of 12 by 12 pixels by 10 frames for constructing small-scale bFs, and the other block consists of 24 by 24 pixels by 30 frames for constructing large-scale bFs.

For both HOG and HOF, orientations are quantized into 8 bins. An additional zero bin is added for HOF (i.e., 9 bins). It accounts for pixels whose optical flow magnitudes are lower than a threshold. Both descriptors are normalized with their \(L_2\)-norm. The final dimension of each descriptor is 96 for HOG and 108 for HOF from one grid.

The MBH descriptor separates the optical flow into its horizontal and vertical components. Spatial derivatives are computed for each component, and orientation information is quantized to be incorporated in the histograms. The magnitude is used for weighting. We obtain an 8-bin histogram for each component (i.e., MBHx and MBHy). Both histogram vectors are normalized separately in the \(L_2\)-norm sense. The final dimensions of descriptors are 96 for both MBHx and MBHy.

The STIP descriptors are extracted based on local interest points that are detected using the extended space-time Harris detector from a fixed set of multiple spatial-temporal scales. For each interest point, we compute two alternative patch descriptors, HOG and HOF, which are computed on a 3D video patch in the neighborhood of each detected STIP. The patch is partitioned into a grid with \(3 \times 3 \times 2\) spatial-temporal blocks: 4-bin HOG descriptors and 5-bin HOF descriptors are then computed for all blocks and are concatenated into a 72-element and a 90-element descriptor, respectively. The final dimension of the STIP descriptor is 162. Finally, all descriptors are normalized, and PCA is performed to reduce the dimensionality by 50%. After that, we quantize each feature descriptor using K-means over the set of all feature vectors from the entire video. We chose \(K=16\) for small-scale bFs and \(K=9\) for large-scale bFs, which is used in Segmentation 3. A and III-B.

We conduct video segmentation experiments using various behavior feature representations on the junction-1 dataset. The segmentation results are shown in Table IV. As shown, the proposed K-CBPL shows better results compared to other algorithms. As we expected, using multi-scale bFs working collaboratively in discovering representative bFs leads to performance improvement, and the nonlinear relationship of K-CBPL leads to better performance than the linear relationship in most cases. We investigate video segmentation using different bF combinations (different feature type and dimension) to construct a set of small and large-scale bF pairs (\(H^S, H^L\)) for K-CBPL on Junction-1 dataset. For this, we select a set of feature representations known to perform poorly on single-scaled algorithms and then obtain performance results on various pairs. We observed that the performance improvement of K-CBPL algorithm is not limited to specific feature combinations. This is shown in Table V.

D. Atomic Events Clustering and Spatial Scene Segmentation

Atomic Events Clustering: In Sec. III-A, all atomic events that occur in the video clip are clustered into \(K\) dominant atomic events at each scale as shown in Fig. 3. After that, both small-scale and large-scale bFs are constructed using the distribution of both \(K\) number of small-scale and large-scale dominant atomic events independently. Fig. 7 shows an
Fig. 7. Different number of small-scale dominant atomic events with $D_s=9$.

**TABLE IV.** COMPARISON OF SEGMENTATION ACCURACIES (%) USING VARIOUS FEATURES AND USE OF DIFFERENT DIMENSIONS ON Junction-1.

<table>
<thead>
<tr>
<th>Feature</th>
<th>$K_{CR}$</th>
<th>PLSA</th>
<th>LDA</th>
<th>EM</th>
<th>LDA (CC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOA ($D_s=0$)</td>
<td>79.49</td>
<td>87.17</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
</tr>
<tr>
<td>HOA ($D_s=1$)</td>
<td>84.92</td>
<td>92.31</td>
<td>92.31</td>
<td>92.31</td>
<td>92.31</td>
</tr>
<tr>
<td>HOA ($D_s=2$)</td>
<td>87.17</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
</tr>
<tr>
<td>HOA ($D_s=3$)</td>
<td>92.31</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
</tr>
<tr>
<td>HOA ($D_s=4$)</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
</tr>
</tbody>
</table>

*Fig. 8(a). The numbers (yellow in color) in Fig. 8(a) are the indices of the colored regions of dominant small-scale atomic events. Fig. 8(b) shows small-scale atomic events (a) event clustering on Junction-1. The numbers (yellow in color) illustrate the indices of dominant small-scale atomic events. Direction of dominant atomic events and arrows indicate direction of dominant movements. As mentioned above, large $K$ value can describe object movements in detail while $K$ value can capture objects that are noisy whereas a small $K$ value captures simplified movement patterns. A similar trend is observed with large-scale atomic events event clustering. In Table V, only when feature #1, #4, #5 and #8 are related to upward traffic flow while #2, #3, #6 and #7 are related to rightward traffic flow.

**E. Collaborative Behavior Pattern Learning with W**

Each large-scale atomic events can be represented as linear combination of small-scale atomic events. As an example, the}

**TABLE V.** COMPARISON OF SEGMENTATION ACCURACIES (%) USING VARIOUS FEATURES AND USE OF DIFFERENT DIMENSIONS ON Junction-1.

<table>
<thead>
<tr>
<th>Feature</th>
<th>$K_{CR}$</th>
<th>PLSA</th>
<th>LDA</th>
<th>EM</th>
<th>LDA (CC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOA ($D_s=0$)</td>
<td>79.49</td>
<td>87.17</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
</tr>
<tr>
<td>HOA ($D_s=1$)</td>
<td>84.92</td>
<td>92.31</td>
<td>92.31</td>
<td>92.31</td>
<td>92.31</td>
</tr>
<tr>
<td>HOA ($D_s=2$)</td>
<td>87.17</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
</tr>
<tr>
<td>HOA ($D_s=3$)</td>
<td>92.31</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
</tr>
<tr>
<td>HOA ($D_s=4$)</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
<td>94.7</td>
</tr>
</tbody>
</table>

*Fig. 8(a). The numbers (yellow in color) in Fig. 8(a) are the indices of the colored regions of dominant small-scale atomic events. Fig. 8(b) shows small-scale atomic events (a) event clustering on Junction-1. The numbers (yellow in color) illustrate the indices of dominant small-scale atomic events. Direction of dominant atomic events and arrows indicate direction of dominant movements. As mentioned above, large $K$ value can describe object movements in detail while $K$ value can capture objects that are noisy whereas a small $K$ value captures simplified movement patterns. A similar trend is observed with large-scale atomic events event clustering. In Table V, only when feature #1, #4, #5 and #8 are related to upward traffic flow while #2, #3, #6 and #7 are related to rightward traffic flow.

**E. Collaborative Behavior Pattern Learning with W**

Each large-scale atomic events can be represented as linear combination of small-scale atomic events. As an example, the
a frame from Junction-1 dataset is shown in Fig. 9. Here, the figures on the top show (a) small-scale and (b) large-scale dominant atomic events, and the figures on the bottom depict (c) initial and (d) optimized W matrices.

The learned W matrix shows that each large-scale atomic event is related to one or two dominant small-scale atomic events. Remarkably, in terms of direction and location, each large-scale atomic event is similar to the small-scale atomic events that are related by W. For example, in Fig. 9, the large-scale atomic event #8, a horizontal flow at the right corner, is related to the small-scale atomic events #13 and #14, which corresponds to the horizontal flow at a similar location. Likewise, the large-scale atomic event #7, a vertical flow in the middle of the right lane, is closely related to the small-scale atomic events #9 and #15, which are also vertical flow in the right lane. It is obvious that there is a strong relationship between the behavior occurring in small and large-scale regions that are overlapping. However, we also notice a certain relationship between non-overlapping regions. In Fig. 9, although the location of the large-scale atomic event #3 is far from the small-scale atomic event #1 and #10, the algorithm has found a strong linear relation between the atomic events. These small and large-scale atomic events are all closely related to vertical traffic flow. Based on these results, the CBPL algorithm can potentially discover the hierarchical structure represented by the W matrix.

### F. Multi-class Temporal Video Segmentation

In this subsection, multi-class temporal video segmentation as a function of K and λ^R is discussed. The different number of representative bPs are discovered by varying λ^R. In the experiment, λ^C = 0.1 × λ^R. As shown in Fig. 10(a), temporal video segmentation results are shown for different λ^R and K, and Fig. 10(b) shows example of representative bPs for K = 6 and λ^R = 4.

When K = 2, two different representative bPs, which correspond to two different modes of traffic flow, are discovered: horizontal traffic flow (green bar) and vertical traffic flow (blue bar). When K = 6, a more detailed segmentation result as shown in Fig. 10(b) is obtained. Here, six different colored rectangles depict different representative bPs (e.g., patterns of traffic flows). When K = 3, the green bar in Fig. 10(a) and the green rectangle in Fig. 10(b) depict leftward traffic flow while the yellow bar and the yellow rectangle depict rightward traffic flow. When K = 2, leftward traffic flow and right traffic flow are grouped into the same class (green bar) as horizontal traffic flow as shown in Fig. 10(a).

To evaluate the effectiveness of the proposed K-CBPL algorithm, a multi-class temporal video segmentation experiment that involves clustering N video clips \{V_i\}^N_{i=1} into K classes based on the discovered representative bPs is conducted using HOA-based bPs on Junction-1, Roundabout and MIT-traffic datasets. For fair comparison, we construct ground truth labels for K = 3 and K = 4 for Junction-1 and Roundabout dataset. Also, we construct new ground truth labels for K = 2, 3 and 5 for MIT-traffic dataset.

The quantitative results of the K-class temporal video segmentation accuracies on the Junction-1 and Roundabout datasets are represented in Table VI. Experimental results show that the proposed K-CBPL algorithm achieves best performance for K = 3 and K = 4.

Examples of multi-class temporal segmentation on Junction-1 and Roundabout are given in Fig. 11 and 12, respectively.
and Fig. 12(b), the proposed K-CBPL algorithm achieves segmentation results when $K=5$ is shown in Fig. 13(a). As shown in Fig. 13(b), the proposed K-CBPL algorithm achieves best segmentation performance.

G. Benchmark Two-class Temporal Video Segmentation

To determine the effectiveness of the proposed K-CBPL algorithm, temporal video segmentation experiment that involves clustering video clips into two classes is conducted. The proposed multi-scale bF based algorithm is compared with single-scale bF based convex optimization algorithms using only small-scale bFs (EMD-$L_1(\mathbb{H}^S)$), only large-scale bFs (EMD-$L_1(\mathbb{H}^L)$) and concatenated (small+large)-scale bFs (EMD-$L_1((\mathbb{H}^S+\mathbb{H}^L))$). The quantitative and qualitative experimental results on QMUL and CVBASE’06 datasets are discussed below.

1) The QMUL Load Traffic Datasets: Quantitative results of the two-class temporal video segmentation accuracies on the QMUL datasets are shown in Table VIII. Experimental results show that the proposed K-CBPL algorithm achieves best performance compared to other the state-of-the-art algorithms. For example, in the temporal video segmentation result on Roundabout in Table VIII, the K-CBPL algorithm achieved 95.91% accuracy, whereas other algorithms achieved lower accuracy: PTMs (pLSA: 72.30%), hierarchial methods (HpLSA:72.30% and DDP-HMM:85.14%), multi-scale method (Cas-pLSA:76.20%), convex optimization methods (EMP: 86.40%, sparse-EMD:90.0%, EMD-$L_1(\mathbb{H}^S)$:88.97%, EMD-$L_1((\mathbb{H}^S+\mathbb{H}^L))$:77.80% and EMD-$L_1((\mathbb{H}^S+\mathbb{H}^L))$:83.05%). As shown in Table VIII, EMD-$L_1((\mathbb{H}^S+\mathbb{H}^L))$, which is based on just concatenating small and large-scale bFs does not lead to performance improvement. However, the

<table>
<thead>
<tr>
<th>TABLE VII</th>
<th>MULTI-CLASS SEGMENTATION ACCURACIES(%) USING DIFFERENT $K$ NUMBER OF REPRESENTATIVE BPs ON MIT-traffic.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K=5$</td>
<td>51.43</td>
</tr>
<tr>
<td>$K=3$</td>
<td>64.43</td>
</tr>
<tr>
<td>$K=2$</td>
<td>68.57</td>
</tr>
</tbody>
</table>
TABLE VIII
SEGMENTATION ACCURACIES (%) FOR THE QMUL DATASETS

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Junction-1</th>
<th>Roundabout</th>
<th>Junction-2</th>
</tr>
</thead>
<tbody>
<tr>
<td>pLSA [1]</td>
<td>89.74</td>
<td>84.46</td>
<td>N.A</td>
</tr>
<tr>
<td>HpLSA [5]</td>
<td>76.92</td>
<td>72.30</td>
<td>N.A</td>
</tr>
<tr>
<td>LDA [5]</td>
<td>61.50</td>
<td>55.90</td>
<td>N.A</td>
</tr>
<tr>
<td>Cas-LDA [5]</td>
<td>87.20</td>
<td>74.60</td>
<td>N.A</td>
</tr>
<tr>
<td>Cas-pLSA [5]</td>
<td>89.70</td>
<td>72.60</td>
<td>N.A</td>
</tr>
<tr>
<td>Xu et al.-LDA [18]</td>
<td>92.31</td>
<td>60.17</td>
<td>N.A</td>
</tr>
<tr>
<td>LDA+Kmeans [18]</td>
<td>91.28</td>
<td>76.25</td>
<td>N.A</td>
</tr>
<tr>
<td>DDP-HMM [8]</td>
<td>87.18</td>
<td>85.14</td>
<td>N.A</td>
</tr>
<tr>
<td>MCTM [5], [7]</td>
<td>51.79</td>
<td>68.36</td>
<td>N.A</td>
</tr>
<tr>
<td>MOHMM [5], [36]</td>
<td>56.40</td>
<td>66.10</td>
<td>N.A</td>
</tr>
<tr>
<td>K-means [25]</td>
<td>76.92</td>
<td>74.58</td>
<td>79.60</td>
</tr>
<tr>
<td>EMP [11]</td>
<td>92.31</td>
<td>86.40</td>
<td>84.81</td>
</tr>
<tr>
<td>Sparse-EMD [12]</td>
<td>89.74</td>
<td>90.00</td>
<td>91.67</td>
</tr>
<tr>
<td>EMD-L1(H+) [11]</td>
<td>87.64</td>
<td>88.97</td>
<td>91.86</td>
</tr>
<tr>
<td>EMD-L1(H+H+) [11]</td>
<td>79.49</td>
<td>77.80</td>
<td>90.88</td>
</tr>
<tr>
<td>EMD-L1(H+H+H+) [11]</td>
<td>88.43</td>
<td>83.05</td>
<td>91.48</td>
</tr>
<tr>
<td>CBPL-(linear)</td>
<td>92.31</td>
<td>91.52</td>
<td>92.13</td>
</tr>
<tr>
<td>Proposed K-CBPL</td>
<td>94.87</td>
<td>95.91</td>
<td>95.20</td>
</tr>
</tbody>
</table>

Fig. 14. Temporal video segmentation result on Roundabout (Horizontal traffic flow (green), Vertical traffic flow (blue)).

(a) Horizontal traffic flow (small)  (b) Vertical traffic flow (small)
(c) Horizontal traffic flow (large)  (d) Vertical traffic flow (large)
(e) Qualitative result of temporal video segmentation

Fig. 15. Temporal video segmentation result on Junction-1 (Horizontal traffic flow (green), Vertical traffic flow (blue)).

(a) Horizontal traffic flow (small)  (b) Vertical traffic flow (small)
(c) Qualitative result of temporal video segmentation

Fig. 16. Temporal video segmentation result on Junction-2 (Horizontal traffic flow (green), Vertical traffic flow (blue)).

(a) Horizontal traffic flow (small)  (b) Vertical traffic flow (small)
(c) Qualitative result of temporal video segmentation

The proposed K-CBPL algorithm can discover the representative bPs more accurately by considering nonlinear relationships among small and large-scale bPs.

The qualitative results of two-class temporal segmentation on Roundabout, Junction-1 and Junction-2, are shown in Fig. 14, 15 and 16. Fig. 14(a) and (b) show respectively small-scale bPs (with small-scale dominant atomic events) classified as horizontal and vertical traffic flows. Fig. 14(c) and (d) show large-scale bPs (with large-scale dominant atomic events) classified as either horizontal or vertical. The proposed algorithm achieves best segmentation result, as shown in Fig. 14(e).

2) The CVBASE’06 Sports Datasets: To evaluate the effectiveness of the proposed K-CBPL, two-class temporal video segmentation was conducted using the CBASE’06 dataset. Two group behavior patterns provided in [32], were considered as representative bPs: (a) team defense and (b) team offense. Since there are no reported results for two-class temporal video segmentation on these datasets, the proposed algorithm is compared to the K-means clustering, pLSA and HpLSA.
TABLE IX
Segmentation accuracies (%) for the CVBASE’06 datasets

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Basketball-A</th>
<th>Handball-B</th>
</tr>
</thead>
<tbody>
<tr>
<td>K-means [25]</td>
<td>74.00</td>
<td>82.00</td>
</tr>
<tr>
<td>pLSA [1]</td>
<td>82.00</td>
<td>85.50</td>
</tr>
<tr>
<td>HpLSA [1]</td>
<td>86.00</td>
<td>82.80</td>
</tr>
<tr>
<td>EMD-L1(H\sup T) [11]</td>
<td>87.76</td>
<td>91.64</td>
</tr>
<tr>
<td>EMD-L1(H\sup -H\sup T) [11]</td>
<td>79.80</td>
<td>89.80</td>
</tr>
<tr>
<td>EMD-L1(H\sup +H\sup T) [11]</td>
<td>88.98</td>
<td>91.98</td>
</tr>
<tr>
<td>CBPL-(linear)</td>
<td>92.00</td>
<td>94.00</td>
</tr>
<tr>
<td>Proposed K-CBPL</td>
<td>93.80</td>
<td>97.14</td>
</tr>
</tbody>
</table>

Fig. 17. Temporal video segmentation result on Basketball-A (behaviors of the left team: defense (green) and offense (blue)).

The EMD-L1 based bP learning algorithm is investigated in a similar way in the experiments for the QMUL dataset. 

Quantitative results of the two-class temporal video segmentation on the CVBASE’06 dataset are shown in Table IX. Experimental results show that the proposed K-CBPL algorithm outperforms other state of the algorithms.

The qualitative results of two-class temporal segmentation on the Basketball-A and Handball-B are represented in Fig. 17 and Fig. 18. Fig. 17 shows the temporal video segmentation results on Basketball-A. Fig. 17(a) and (b) show two class of representative bPs from the left-hand side point of view: (a) the left team is on the “defense” and (b) the left team is on the “offense”. The numbers in yellow are the indices of the colored regions of dominant small-scale atomic events shown in Fig. 17(a) and 17(b) from Basketball-A. As shown in 17(c), the proposed K-CBPL algorithm achieves best segmentation results compared to other methods. Fig. 18 shows the temporal video segmentation results on Handball-B dataset. Fig. 18(a) and (b) show two class representative bPs from the right-hand side point of view: (a) the right team is on the “defense” and (b) the right team is on the “offense”. As shown in 18(c), the proposed K-CBPL algorithm achieved the best segmentation results compared to other algorithms.

VI. CONCLUSIONS

This paper proposes an unsupervised kernelized collaborative behavior pattern learning (K-CBPL) algorithm for VSA. The main idea of the proposed algorithm is discovering representative bPs using two different object bPs that can collaboratively assist one another in discovering representative bPs simultaneously and to enhance the accuracy where bPs are intrinsically correlated with each other from various object movements in the complex video scene. Although a hierarchical structure of bFs has been studied, each bP is independent learned. Hence, the hierarchical behavioral information does not reflect appropriately. In the proposed algorithm, we enforce that representative bPs should be discovered by sharing their multi-scale feature information simultaneously in the pattern learning framework. The proposed optimization problem can be solved efficiently by linear-programming based on multi-scale features collaboratively. The experimental results on various benchmark datasets show that the proposed algorithm achieves the best performance compared to the state-of-the-art algorithms. The benefit of considering the proposed hierarchical relationship and collaborative learning is shown in all the experiments. It is confirmed that the proposed algorithm can discover the representative bPs accurately by considering the correlation among multi-scale bPs in the complex video scene.

REFERENCES


Sanghyuk Park (S’11) the B.S. degree in electronics and communications engineering from kwangwoon university, Korea, in 2007, the M.S. degree in computer science from yonsei university, Korea, in 2009. His research interests are video scene analysis, behavior modeling, object detection and recognition, and machine learning.

Hyunsik Park (S’09) received the B.S., M.S. degrees from the Kobe university, in 2007 and 2009, respectively, all in computer science. His research interests are speech recognition, audio signal processing, and machine learning.

Chang D. Yoo (S’92-M’96-SM’11) received the B.S. degree in Engineering and Applied Science from the California Institute of Technology, the M.S. degree in Electrical Engineering from Cornell University and the Ph.D degree in Electrical Engineering from the Massachusetts Institute of Technology. From January 1997 to March 1999 he was Senior Researcher at Korea Telecom (KT). Since 1999, he has been on the faculty in the School of Electrical Engineering at the Korea Advanced Institute of Science and Technology (KAIST). He also served as Dean of the Office of Special Projects (2011-2013) and Dean of the Office of International Relations (2013-2015). He has been a consultant for many firms and is currently a consultant for the Korean Foundation for Advance Studies (one of the oldest and largest scholarship foundations in Korea). He is also Director of Korea Institute of Electrical Engineers (KIEE) and Director of the Acoustical Society of Korea (ASK). He is Member of Tau Beta Pi and Sigma Xi. He was on the technical committee member of IEEE machine learning for signal processing society from 2009 to 2011. He had also served as Associated Editor of IEEE Signal Processing Letters, IEEE Transactions on Information Forensics and Security, IEEE Transactions on Audio, Speech and Language Processing.

Dean of the Office of Special Projects (2011-2013) and Dean of the Office of International Relations (2013-2015). He has been a consultant for many firms and is currently a consultant for the Korean Foundation for Advance Studies (one of the oldest and largest scholarship foundations in Korea). He is also Director of Korea Institute of Electrical Engineers (KIEE) and Director of the Acoustical Society of Korea (ASK). He is Member of Tau Beta Pi and Sigma Xi. He was on the technical committee member of IEEE machine learning for signal processing society from 2009 to 2011. He had also served as Associated Editor of IEEE Signal Processing Letters, IEEE Transactions on Information Forensics and Security, IEEE Transactions on Audio, Speech and Language Processing.