Multimed Tools Appl (2016) 75:2453–2472
DOI 10.1007/s11042-015-2876-y

A bag-of-regions representation for video classification

Min-Kook Choi[1] · Ziyu Wang[1] · Hyun-Gyu Lee[1] · Sang-Chul Lee[1]

Received: 25 November 2014 / Revised: 22 April 2015 / Accepted: 10 August 2015 /
Published online: 17 September 2015

Abstract A bag-of-regions (BoR) representation of a video sequence is a spatio-temporal tessellation for use in high-level applications such as video classifications and action recognitions. We obtain a BoR representation of a video sequence by extracting regions that exist in the majority of its frames and largely correspond to a single object. First, the significant regions are obtained using unsupervised frame segmentation based on the JSEG method. A tracking algorithm for splitting and merging the regions is then used to generate a relational graph of all regions in the segmented sequence. Finally, we perform a connectivity analysis on this graph to select the most significant regions, which are then used to create a high-level representation of the video sequence. We evaluated our representation using a SVM classifier for the video classification and achieved about 85 % average precision using the UCF50 dataset.

Keywords Video segmentation · Region tracking · Bag-of-regions · Video classification

1 Introduction

Video representation is a key issue in high-level vision systems, as it underpins many processes, including video classification, action recognition, and content-based video retrieval systems (CBVRs). The video representation algorithm for these processes is a key technology to establish practical systems such as in motion capturing, video editing, unusual activity detection, multimedia searching, etc. The recognition or classification of a video sequence in human terms requires an autonomous extraction of semantically plausible

✉ Sang-Chul Lee
sclee@inha.edu

Min-Kook Choi
mkchoi@inha.edu

[1] Department of Computer and Information Engineering 100 Inha-ro, Inha University, Yonghyun-dong, Nam-gu, Incheon, Republic of Korea

information from the video. Current research on the semantic analysis of video sequences is generally considered to have four layers: the frame, shot, scene, and video itself, where human perception is related most directly to the shot and scene layers [45].

There are several efficient ways to analyze the shot or scene layer of a video sequence based on the model of human perception, but they all have a difficulty in dealing with the huge amount of redundant data, and therefore it is necessary to use techniques such as feature representation, dimensionality reduction, and the elimination of duplicated data to cope with this problem. The use of keyframes may be a possible starting point for a semantic analysis. Raptis and Sigal [34] proposed an algorithm for group activity analysis using an action model that is constructed by local discriminative keyframes. The keyframes are collected from the partial information of key-poses of the actor(s) and key states in action sequences. They provided a learning metric that uses latent variables with keyframes in a max-margin discriminative framework. Although the keyframe based model works well in certain applications in classification and recognition, it has its own problems, as it has to extract meaningful keyframes, consider temporal information, and handle detailed data in an efficient manner.

Further, several lines of research aiming to describe video semantics have been conducted. The time interest point technique [9, 20, 22, 42] provides a point-based representation with certain dominant features [12, 13] to recognize an action within a video sequence. Although it achieves acceptable results on the most distinctive features of motion, a point-based representation provides only very low-level of information to bridge the semantic gap. Other key approaches for semantic analysis for video classification have been proposed to obtain efficient semantic cues in streaming information such as pyramid matching [11, 31], parts and attributes based model [33, 44], motion and trajectory analysis [17, 18, 39], and holistic features of text and video [2, 7], and etc. Many of them have used region-based approaches to represent the semantics of a video volume. Although region-based approaches are typically noisier than point-based approaches, the region-based approaches can represent much wider spectrum of semantics since a region is considered as an effective aggregation of similar information. Bilen and Namboodiri [6] proposed a robust region-tracking algorithm to recognize actions in a video sequence. They used a motion segmentation method to obtain connected components that correspond to moving objects. Banerjee and Sengupta [3] implemented unsupervised learning to create a video model, and used a novel approach for building a graph-matching model for tracking objects based on temporally stable regions (TSRs). A stable method of motion segmentation was used to obtain a coarse delineation of the area where actions occur, and homogeneous regions were also segmented on the basis of color information using a mean-shift technique.

The bag-of-regions (BoR) approach is an extension of the bag-of-features (BoF) representation [23] and has been used by G. Demir and A. Selim [14] to classify outdoor scenes using a Bayesian classifier. Their recognition process took account of the spatial distributions of regions that are salient in a given image. To achieve this goal, they segment an image in two steps: The initial classification is used to obtain regions homogeneous in color and texture by applying Gaussian classifiers to a scene. In the subsequent structure detection step, regions outside of the initial segmentation that have similar structural properties are identified. They generated a region codebook using the HSV vector, and used this codebook to make probabilistic measurements based on spatial distributions. Our work is motivated by this approach by considering three-dimensional spatio-temporal space, and uses more robust features and a graph representation to distinguish representative regions.

For a bag-of-regions and graph representation in order to constitute a sparse video representation, video object segmentation (VOS) has to be performed in advance. One of the

most difficult challenges in VOS is a change in the contours of regions in successive fames over time. To cope with this problem, Brendel and Todorovic [8] proposed an unsupervised segmentation method in which temporal information is used to identify candidate regions corresponding to static and moving objects. The authors used a low-level (e.g., mean-shift) segmentation to extract regions on a local basis in each frame, and then applied the Lucas-Kanade optical flow estimation method to recognize similar areas within frames. Next, regions are merged and split through a contour-matching method, which is an extension of dynamic time warping (DTW) to cyclic sequences. Deng [15] also implemented an unsupervised segmentation of video objects to extend the previous JSEG method into a spatio-temporal volume. This latter segmentation algorithm is based on the tracking of homogeneous color and texture by region growth.

Another important approach toward understanding a scene or action, i.e., classification and recognition, is the extraction and selection of useful features in the spatio-temporal domain. Turcot and Lowe [38] selected a small subset of training features for recognition using term frequency – inverse document frequency (TF-IDF) ranking in a BoW framework. The best TF-IDF matches are geometrically validated using a pre-computed affine model with an epipolar geometry. This approach has achieved significantly improved recognition results without expanding the set of required features [38]. A tracking method can also be used to organize a feature representation and enhance the results of feature extraction and selection. Makadia [28] addressed the problem of large-scale image retrieval in a wide-baseline setting within a bag-of-virtual-words framework. He achieved the unsupervised tracking of millions of features using a weighted graph embedded in a quantized feature space to deal with very large datasets. Sivic and Schaffalitzky [36] also used a tracking method to extract spatio-temporal objects from a video sequence by representing them as affine covariant regions.

Our approach was inspired by the bag-of-regions [14] method and TSRs [3]. To achieve robust division and tracking of regions, especially when they should be detected over the majority of frames within a shot, we use the JSEG video object segmentation algorithm to pre-process a video into semantically promising regions. JSEG is a region-growing method that uses color and texture properties, and is similar to the embedded seed tracking method used by Brendel and Trodorovic [8] for video segmentation. Since JSEG determines the homogeneity of the color and texture properties by tracking the overlapping regions, a seed tracking method based on region growth is likely to be more efficient in this context than alternative segmentation methods such as watershed segmentation. However, when a motion is excessively violent, as is in an action video, discarding the seeds in each frame using a temporally extended JSEG can cause several regions to become too small to track. This fact motivated us to use top-down tracking to match the irregular regions that belong to the same object after the segmentation using the JSEG algorithm. Finally, the candidate regions are selected from the tracking results, that are stored as a relational graph, in which each vertex is a region, and the edges connecting vertices corresponding to such regions are assigned to the same object. In this graph, we can then identify a bag-of-regions that contains objects and scenes similar to those recognizable by human viewers.

The main contributions of this paper are as follows: (1) an alternative representation of determining bag-of-regions representations in spatio-temporal domain and (2) lower computational complexity for full frame video processing with comparable performance in classification accuracy, when compared with other state-of-the-art methods.

The remainder of this paper is as follows. In the next section, we present a bag-of-region representation using region tracking with a relational graph in more detail. The experimental

results using the SVM classifier [41] are presented in Section 3. Finally, concluding remarks regarding the proposed method are given in Section 4.

2 Bag-of-regions using region-graph tracking

Our method consists of the following steps: First, regions are extracted from each video frame using JSEG segmentation. The segmentation results may be irregular because of the splitting and merging of regions caused by illumination changes, occlusions, shadows, and inhomogeneous textures [15]. To deal with this problem, in the next step, each segmented region is tracked in a top-down manner. Irregularities in the segmentation are then dealt with using a matching metric determined from a relational graph, which itself is extracted from the segmented video. Regions that have larger areas and more connections with other regions in the graph are selected to become a part of the bag-of-regions representation. Finally, features such as the color, shape, and local interest points descriptors are extracted from the representative regions for the subsequent video classification. Figure 1 shows the overview of the proposed method applied to a video sequence for classification.

In the following sections, we describe the two main stages of our method: sparse region segmentation using the JSEG algorithm and the creation of a bag-of-region representation using region tracking with a relational graph.

2.1 Preprocessing and JSEG segmentation

Since JSEG segmentation uses the color class and texture property with color variances, it tends to over-segment regions due to the noisy structure of color and texture, by nature. To reduce the response of JSEG against the excessive sensitivity of the color distribution and complex texture, we apply a preprocessing step to the original video frames using mean-shift smoothing. We selected 15 parameters of spatial coordinates h_s and color space h_r in the joint domain of $K(\mathbf{x}) = C \cdot k_s(\|\mathbf{x}^s / h_s\|) \cdot k_r(\|\mathbf{x}^r / h_r\|)$, where C is the normalization constant, and k is the multivariate kernel density estimation computed at point $\mathbf{x}$. Figure 2 shows the differences in the segmentation results between the original and preprocessed images.

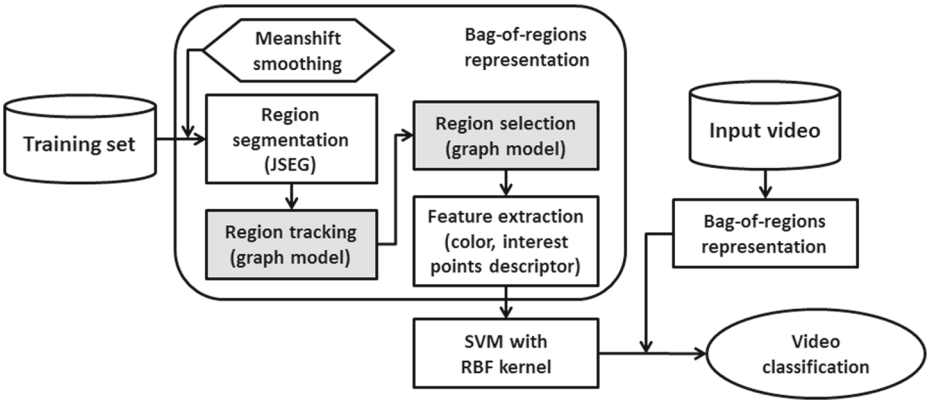


Fig. 1 Overview of the proposed method. The shaded steps show the main idea of this work. The method provides an alternative sparse representation of a video sequence

Fig. 2 **a** An image of a 'baseball pitch' video clip. **b** JSEG segmentation results of the original image. **c** Preprocessed image using mean shift smoothing. (d) JSEG segmentation results of the preprocessed image. **d** The improved segmentation results

Many segmentation algorithms use visible image properties (e.g., color, edge, and texture) and methods (e.g., graph cut, belief propagation, and mean shift). In previous work, k-means and mean-shift segmentations [3, 8, 40, 45] have been employed to obtain regions that can then be tracked to find the TSRs in a scene. However, it is difficult to choose segmentation algorithms that are based on low-level features to produce satisfactory results. Instead, we use JSEG segmentation to extract regions that have semantic significance since they are homogeneous in terms of properties such as color and texture. The boundaries between neighboring regions should then contain the boundaries of objects in a frame. In the preprocessing step, regions are selected based on their color and shape information, and during the subsequent tracking we pursue an ideal segmentation in which each region corresponds to a single semantic entity. These regions then take part in the whole process of segmentation and region tracking. The segmentation algorithm consists of two stages: color quantization, in which the colors in a frame are quantized into several representative classes, and each pixel is replaced by a label identifying the corresponding color class; and robust spatial segmentation through the construction of multi-scale 'J-images' using the criterion of data homogeneity, which are then used for seed detection [15].

Each pixel is then labeled with its class, creating a color-class map, which can be viewed as a 2D image of the height data. Usually, a coherent region in this image will contain pixels with similar color properties from a small subset of the color classes. After the color-class map has been generated, the criteria used for segmentation can be defined as follows: Let Z be a set of all N data-points in the class map, and let $z(x, y)$, $z \in Z$, and m be the mean of z. Suppose Z is classified into C classes, $Z_i, i = 1, ..., C$, and m_i is the mean of

the N_i data-points of class Z_i. Further, let $S_T = \sum_{z \in Z} \|z - m\|^2$ and $S_W = \sum_{i=1}^{c} S_i = \sum_{i=1}^{c} \sum_{z \in Z_i} \|z - m_i\|^2$.

The measure of J is then defined as $J = \frac{S_B}{S_W} = \frac{S_T - S_W}{S_W}$ and the average value of J can be expressed as follows: $\bar{J} = \frac{1}{N} \sum_k M_k J_k$, where J_k is the value of J calculated over region k, M_k is the number of points in k, N is the total number of points in the class map, and the summation includes all regions in the class map. A better segmentation tends to have a lower value of $\bar{J}$ through the minima of local J values [15]. Since a J-image is generated by calculating J in small local windows, it represents both region interiors and their boundaries. The pixels corresponding to locally minimum values of J are selected as seeds for region growth. Starting from these seeds, the regions are grown and merged forming the initial JSEG segmentation.

2.2 Bag-of-regions (BoR) representations

The JSEG segmentation can segment regions that undergo a rigid-body or affine transformation between frames. Nevertheless, occlusion by other objects, shadowing, or changes in illumination, can cause the color and texture properties of all or parts of a region to change suddenly. It is a big challenge for most feature extraction methods to track objects using the features that have been changed. Unless the source of a change is understood and can be accommodated explicitly, regions of objects may vanish, or new regions may appear between consecutive frames. The appearance of a new region between frames can be handled by the relations between regions, but existing algorithms cannot detect occlusions effectively in terms of overlapping regions. This and other similar issues have to be handled before a bag-of-regions representation is constructed. An ideal example of this combination of regions is shown in Fig. 3.

In our system, we combine regions based on size, color, and texture. First, all the frames in the sequence are segmented into regions using the JSEG algorithm, and then we extract the candidate regions from the spatio-temporal volume of the video sequence. Regions are typically over-segmented by JSEG, and thus several regions actually belong to the same video object. This disparity between regions and objects is caused by a change in viewpoint and sudden differences between frames. Regions that should be assigned to the same object can be viewed as a subset of the segmented sequence. To create a meaningful bag-of-regions representation, it is necessary to carefully select from the initially generated regions. Our region selection method is based on a relational graph, which is generated using

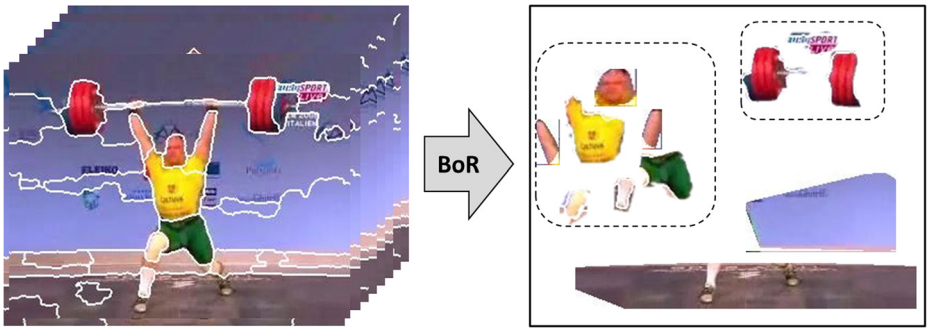


Fig. 3 An ideal example of bag-of-regions (BoR) representations; the dotted rectangles are meaningful combinations of regions, while the remaining regions make up the background

region-based tracking to construct the relations for each shot. A region is represented by a vertex in the graph, and the edges of the graph connect the vertices corresponding to regions that are part of the same object in consecutive frames. The connected components of the graph then correspond to bag-of-regions representations of the video objects.

The main challenge in the tracking process based on color and texture is in matching regions with an irregular shape and size. To maintain a consistent JSEG segmentation, the J measure is used to match pairs of regions, and the color consistency applied during region merging is used to create new objects. Regions are matched from frame to frame based on the location and size, as these quantities are not normally subject to extreme changes between frames.

The regions extracted from a video can be divided into two different types: segments of the background, which might be a natural scene, which usually has a reasonably uniform color and texture; and segments of the foreground objects. When the camera is stationary, both types of regions are likely to appear in most frames; however, when the camera follows the main objects as they move across the scene, almost all the TSRs come from the objects being tracked by the camera. Both region types can be expected to contribute to the recognition of events [25] in a video sequence, as TSRs appearing in several frames can be assumed to represent significant objects. The use of a bag of TSRs reduces the redundancy of the representation when both objects and the background appear in several successive frames, and is therefore suitable for the classification and understanding of tasks. This approach results in a good segmentation with large regions. Large regions are much more likely to encapsulate the information on video objects that are important to the viewer. The tracking algorithm and selection policy are described in more detail in the following section.

2.2.1 Assumptions for robust tracking

The main goal of tracking [30] is to construct a relational graph that allows the TSRs to be associated with the candidate representations. During tracking, a region may appear, vanish, or separate, or two regions may merge. To deal with these events effectively, we make two major assumptions:

Locality assumption Regions belonging to the same object will not move far apart in consecutive frames.

Parent relation assumption A single region will not participate in splitting and merging at the same time, and therefore each region can be linked to its parent node in the graph.

We exploit the locality assumption by attaching a 2D bounding box to each region, which provides a boundary on the search for matches. The parent relation assumption reflects the requirement that each region has a single meaning and its own unique properties. This assumption can be met by directing the search toward large regions in consecutive frames, and then excluding these regions from the matching process.

While tracking, we use the time indices of the frames within a shot to provide the sequential information for the tracking: each region corresponds to a vertex in a graph structure. The set of regions with the same time index can be considered as an independent graph, g_j, which is mapped to the bipartite graph. A segmented video is expressed as a set of graphs, $G = \{g_i | i = 1, , I\}$, where I denotes the number of frames in the video, and each g_j is a graph that is initially disconnected. We then combine g_j and g_{j+1} by adding edges using the tracking model, until all versions of g in G are processed and we have a relational graph that identifies the objects in each component for each frame.

2.2.2 Details of the tracking algorithm

The tracking algorithm is based on matching regions from frame to frame. Given a pair of frames, g_j and g_{j+1} are the corresponding subsets of the bipartite graph, with n_j and n_{j+1}, which are total number of the segmented regions from given consecutive frames, respectively. The total number of vertices is $N = n_j + n_{j+1}$, and following notations are used in the matching of each vertex:

1. Area $A(v)$ is the number of pixels inside the region;
2. Bounding box, $B(v)$, is the smallest rectangle surrounding the region;
3. Region $R(v)$ is an image made up of segmented regions.

The indices of a segmented region in a first frame, $A(v_i)$, $Box(v_i)$ and $R(v_i)$, $i = 1, ..., n_j$, are stored as attributes of each vertex in the list of vertices. We sort this list using $A(v_i)$ in descending order. Algorithm 1 describes tracking details with given notations.

Algorithm 1 Region Tacking form Two Consecutive Frames

Inputs:
$R(v_{i,j})$: i^{th} segmented regions in j^{th} frame
T: pixel threshold
d: color threshold
Outputs:
$\hat{V}_i$: list of vertices
J_i: color variance of $\hat{V}_i$
Initialize:
$A(v_{i,j}) \leftarrow$ # of pixel in $R(v_{i,j})$
$B(v_{i.j}) \leftarrow$ bonding box from $R(v_{i,j})$
$V_j \leftarrow \{v_1, ..., v_{n_j}\}$ sort listed of vetices by $A(v_{i,j})$ in descending order
for i = 1 to n_j **do**
  for k = 1 to n_{j+1} **do**
    $O(v_{i,k}) \leftarrow \#(B(v_i) \bigcap B(v_k))$
    if $|A(v_i) - O(v_{i,k})| < T$ **then**
      $\hat{V}_i \leftarrow v_k$
    end if
  end for
  $m = \text{length}(\hat{V}_i)$
  $S = \sum_{v_t \in \hat{V}_i} A(v_t)$
  for t = 1 to m **do**
    $D_t = \| P_i - P_t \|$, where $P_i = Hist(R(v_i))$
    if $D_t > d$ and $|(A(v_i) - S| > T$ **then**
      pop out v_t from $\hat{V}_i$
    end if
  end for
  $l = \text{length}(\hat{V}_i)$
  $S_{\hat{V}_i} = \sum_{v_t \in \hat{V}_i} S_{v_t}$
  $J_i = \frac{S_{v_t} - S_{v_t}}{S_{\hat{V}_i}}$
end for

The combination with the smallest J_i is considered as the best match. Matching vertices are then removed from the list. The matching terminates when all the vertices in the current and following frames are considered. When all the frames in the sequence have been processed, the construction of a relational graph is complete.

2.2.3 Details of matching

Our matching method uses a rejection scheme with three steps, from coarse to fine, which can accommodate irregular regions of video objects. The properties of the region corresponding to a vertex, such as its homogeneity and size, are used for matching, as we previously discussed. The most suitable regions in each frame undergo the following steps:

1. Nearby regions that have satisfied the locality assumption are rejected on the basis of the color property. We use a threshold of color distance from the region merging step of the JSEG algorithm to reject regions with very different colors. However, if an object with a different color appears between frames, the new object has to be segmented individually using the locality assumption.
2. The size of the region is calculated to help dealing with the irregular shapes of segmented regions. Although most of the segmented regions share boundaries with video objects, and these boundaries change over time, they will not usually change too much between consecutive frames. Using this assumption, we find all the combinations of regions in V_j. The total time required for this step depends on the number of vertices in V_j, where the total number of combinations in V_j is 2^n.
3. Combinations of regions with a total size similar to that of the current region, and in turn each combination of vertices, are split from v_i. The threshold for matching should be set to provide similar bounding boxes used in constructing $\hat{V}_i$. A higher threshold makes the matching more robust, but requires more values of J_i to be computed in the next step. In our experiments, we used a threshold of 15%. Finally, we determine J_i for each region of v_i and $\hat{V}_i$ in Algorithm 1. The smallest J_i corresponds to the region with its vertex in $\hat{V}_i$, which is most likely to split from region v_i. Therefore, region v_i and the regions corresponding to the vertices in $\hat{V}_i$ are assigned to the same object in different frames.

When new objects appear with a color or texture similar to nearby objects, the segmentation may be poor. In addition, complicated overlaps and occlusions occasionally split a region into several smaller regions, which are too small to be useful for a bag-of-regions representation. Figure 4 shows a graph model using the proposed region tracking metric with segmented regions extracted from a temporal sequence of frames.

2.2.4 Region selection

The region tracking process described in the previous section allows us to construct a relational region graph from an image sequence. Many connected components in the graph correspond to meaningful objects. We first discard isolated vertices and components with the longest path that is shorter than the threshold length. These fragments are unlikely to make a meaningful contribution to a video object. The remaining connected components are stored as TSRs.

We assume that region tracking will produce an acceptable segmentation of every object with a homogeneous color and texture. However, incorrect connections of tracked regions can occur in the relational graph, since over-segmentation can cause part of a homogeneous

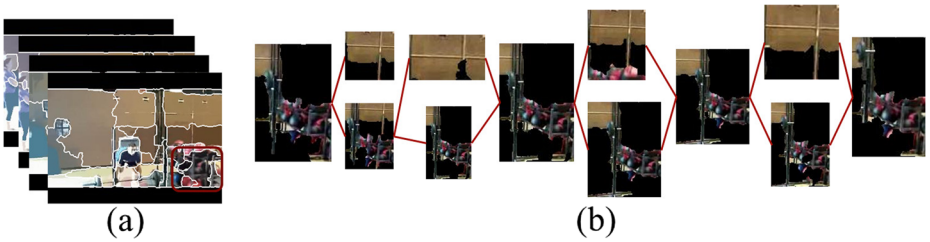


Fig. 4 **a** Segmented sequence using JSEG segmentation with mean-shift smoothing. **b** A graph model using the proposed region tracking metric around the red rectangle. Segmented regions are extracted from the temporal sequence of the frame. Two major assumptions are applied to achieve robust region tracking

region to be allocated to other components. Therefore, we choose regions in each connected component that have the largest area, and correspond to vertices with more connectivity, as shown in Fig. 5. The selected regions are likely to split and have a high probability to correspond to an object or part of a specific object in a video sequence.

Region selection interprets a video as a bag-of-regions, forming a jigsaw of semantically plausible pieces of each frame. The experimental results confirm that the regions selected usually encapsulate important features on the shot layer. Since a bag-of-regions representation has a greatly reduced redundancy, it is suitable for a subsequent understanding of tasks and content-based retrieval. Figure 6 shows an example of a final sparse representation using the proposed BoR representation with a video clip.

3 Experimental results

3.1 Dataset for video classification and action recognition

We applied the BoR method to the video classification problem. Several kinds of benchmark datasets have been released to evaluate action classification or recognition and video classification algorithms. These datasets are mainly divided into two classes depending on their recording properties, where each video clip typically contains a single video shot [5,

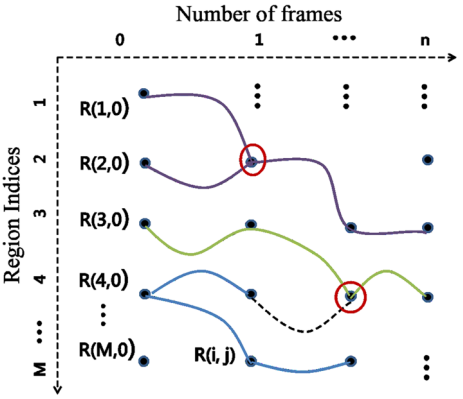


Fig. 5 Region selection policy. Each region is identified by a time index. The dotted line is an incorrect connection between two components. We chose the regions corresponding to the circled vertices

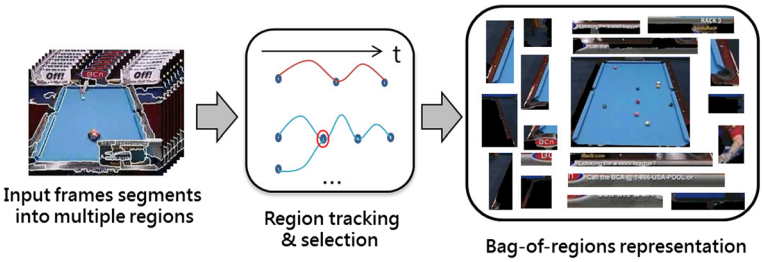


Fig. 6 An example of a bag-of-regions representation with a given video clip (UFC50, 'Billiards')

21, 26, 29, 32, 35, 43]. The first datasets consist of a single staged actor with no occlusions and less clutter, such as in KTH [35], Weizmann [5], and IXMAS [43] (around 611 actions each). The other datasets consist of more challenging recording environments such as large variations in camera motions, object appearance, object positions, scales, and the viewpoints of the actors with complex cluttered backgrounds, such as in Hollywood [29], UCF50 [26], Olympics [32], and HMDB51 [21].

Since most of these are intended for evaluating human action recognition, and not for a sparse representation of a video sequence, we selected ten classes from the UCF50 datasets for evaluation. These chosen classes are suitable for evaluating sparse video representations that present transparent semantic information in either the foreground or background. In our work, we focused on a sparse video representation using a bag-of-representations (BoR) scheme, and then compared our classification results with other techniques for ten classes of classification from UCF50, as reported in [21].

3.2 Video classification results using UCF50 dataset

We used ten classes of videos from UCF50: Baseball Pitch (BP), Basketball (BB), Billiards (Bi), Breast Stroke (BS), Fencing (FE), Military Parade (MP), Punch (PU), Rock Climbing Indoor (RCI), Skiing (SK), and Tennis Swing (TS). These videos contain background and foreground regions with different properties, and specific regions of these shots provide significant clues to what is happening on screen. Some of the videos, such as Billiards, Breast Stroke, and Skiing, have distinct colors or textures that make them suitable for JSEG segmentation. However, other videos, such as Military Parade and Rock Climbing Indoor, tend to become over-segmented under the same parameters. Figure 7 shows sample frames from these videos.

- ***Feature extraction.***

  To separate regions with different semantic contents and achieve robust classification, we extract features based on color and local interest point descriptors from each region in the BoR. Although there are many types of local descriptors [24, 37], we selected most widely used local descriptors such as PCA-SIFT, SIFT, and SURF [4, 19, 27]. To handle a large number of points of interest, we used a grid-based approach in each extracted region. This grid-based approach is a feature-selection method that selects the most dominant points of interest represented by the largest magnitude of descriptor vectors. We tested six grid sizes from 9 to 144, and then chose 36 as the optimal size. To effectively extract the color features and overcome various illumination conditions, we used the histogram of the hue value in a region based on the HSV color

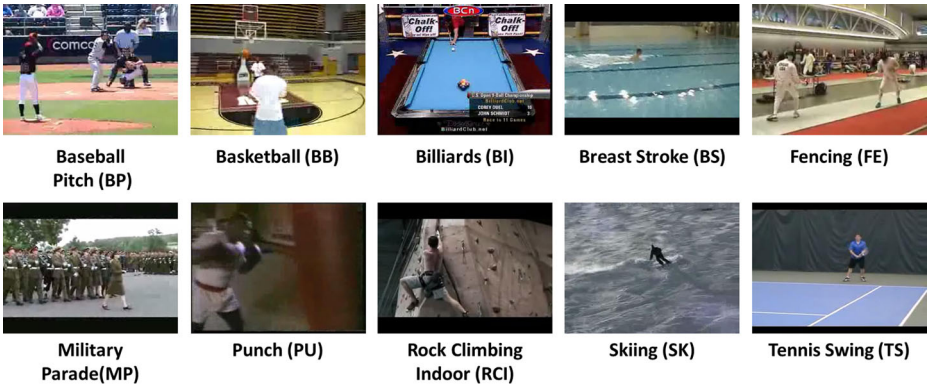


Fig. 7 Sampled frames from 10 classes videos of the UCF50 dataset

space [16]. We tested bin sizes of 8 to 100 to find the optimal size for the histogram, which was determined to be 32 bins.

- ***Selection of SVM parameter and weight of features.***

  We used the SVM classifier based on the VC dimension theory [41]. Given a training set of instance-label pairs $(x_i, y_i), i = 1, 2, ..., l,$, where $x_i \in \mathbb{R}^n$ and $y \in \{-1, 1\}^l$, the soft margin SVM model $f(x) =< w, \phi(x) > +b$ is obtained by solving the following optimization problem:

$$\min_{w,b,\xi} \frac{C}{2} w^T w + \frac{1}{l} \sum_{i=1}^{l} \xi_i, \quad s.t.\ y_i f(x_i) \geq 1 - \xi_i,\ \xi_i \geq 0, \tag{1}$$

where the feature x_i are mapped to a higher-dimensional space by function ϕ and ξ is slack variable for admitting marginal error of a support vector [10]. We can generate multiple support vector model from a each feature from the given SVM model, and then we linearly combine decision result of each feature:

$$Z = \gamma \cdot Color + (1 - \gamma) \cdot LIPD, \tag{2}$$

where $Color$ and $LIPD$ are decision results from given SVM model in (1), and γ is combination weight for final decision. Finally, we tuned the weights on tuned SVM with RBF kernel between the features, local interest point descriptor (LIPD), and color histogram, then we selected the weight as $\gamma = 0.67$. Figure 8 shows classification accuracy for tuning combination weight.

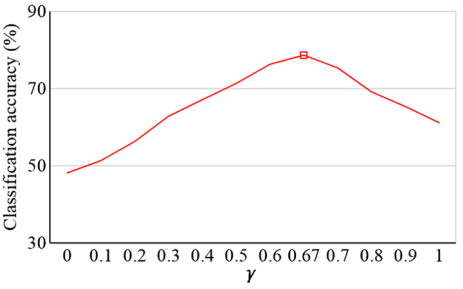


Fig. 8 Tuning result of the combination weight γ using classification decisions between color histogram and local descriptor

- ***Classification evaluation*** The following precision and recall measures facilitate an evaluation of the classification results: (a) classification error, $fp/(tp+fp)$; (b) precision, $tp/(tp+fp)$; (c) recall, $tp/(tp+fn)$; (d) true negative rate, $tn/(tn+fp)$; and (e) accuracy, $tp+tn/(tp+tn+fp+fn)$, where tp, fp and fn are the true-positive, false-positive, and false-negative rates, respectively.

We trained the classifier using 70 shots from each of the ten selected classes of UCF videos (700 shots in total), and a further 30 shots in each class were used for testing (300 shots in total). To verify advantages of the proposed method, we compared the classification results of the BoR based method with two baseline methods; the keyframe based method uses whole features in each keyframe after extraction proposed by [1], and the largest-region-sampling (LRS) method uses features from the largest segmented region with similar color histogram distance, which is defined as $D_i < d$ in Algorithm 1. The LRS uses a

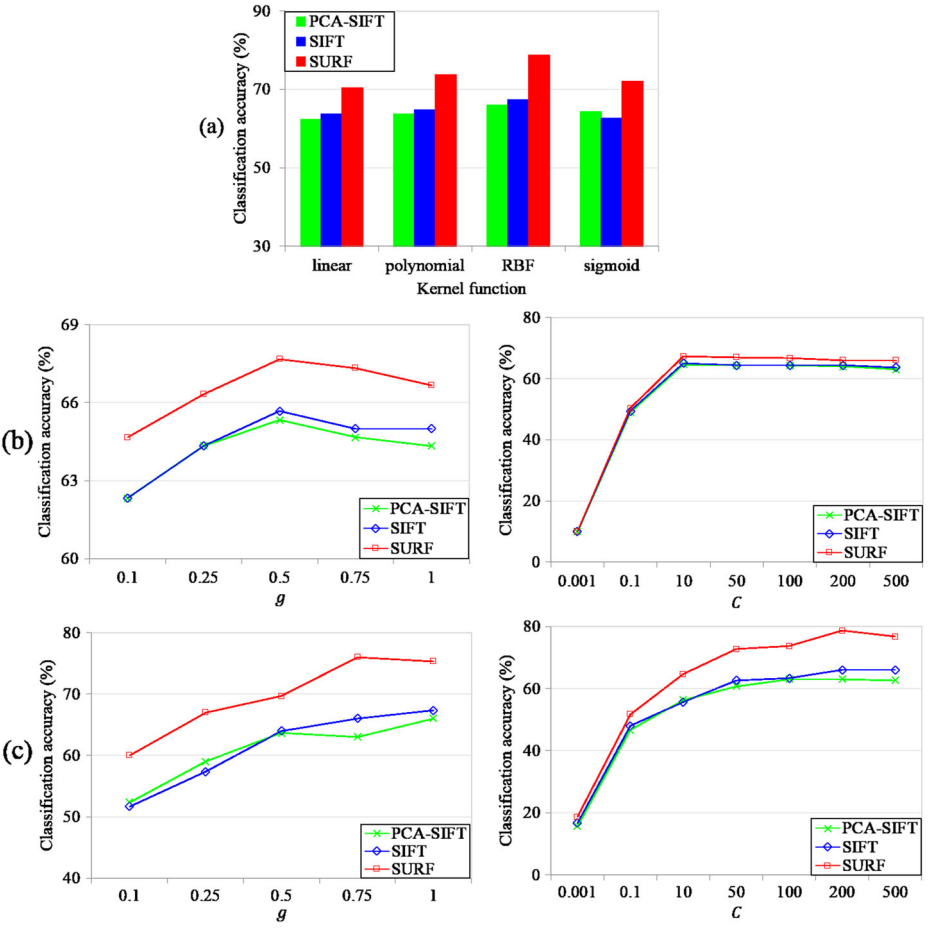


Fig. 9 Tuning parameters of the SVM classifier for BoR and baseline methods: **a** g for RBF kernel function. **b** Regularization cost C. **c** Choice of kernel function. Note that, based on this analyses, we selected the SVM parameters as follows: the gamma in the kernel function $g = 0.75$ and the cost value $C = 200$ with the radial basis kernel function for BoR based method; the gamma in the kernel function $g = 0.5$ and the cost value $C = 10$ with the radial basis kernel function for keyframe based method

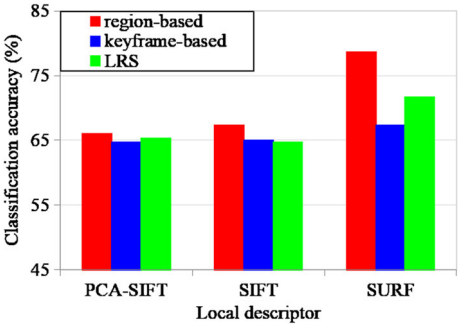


Fig. 10 Comparison of classification accuracy with tuned SVM parameters using different types of local descriptors

simple sampling technique instead of the top-down tracking to reject redundant regions from segmented video.

We performed comparison using three different types of local interest points descriptors such as PCA-SIFT, SIFT and SURF, and tuned the SVM parameters of kernel function. We chose RBF kernel obtaining best performance (shown in Fig. 9a) and also tuned gamma g as a kernel parameter, and regularization cost C for the given SVM model (shown in Fig. 9b and c).

Figure 10 and Table 1 show the comparisons of classification accuracy between the BoR and the baseline methods, using three different types of features. The BoR achieves the best results for all types of features, and the BoR outperforms the baseline method by 7 % when using SURF descriptors. Table 2 shows the confusion matrices of the best accuracy for the BoR, using only the color features (a), only the SURF descriptors (b), and a combination of color features and SURF descriptors (c). The final precision and recall rates of the classification were 69.67 % and 75.64 %, respectively, when we used only the color features based on the BoR. The use of SURF descriptors with the BoR achieved precision and recall rates of 49.33 % and 69.35 %, respectively, while combining the SURF descriptors and color features with the BoR after tuning weight achieved rates of 78.67 % and 81.53 %.

Finally, we compared the classification results of the proposed method using the BoR representations with those of the interest point based method using HOG/HOF features [21]. The authors of [21] performed classification using their selection of ten classes from the UCF50 dataset consisting of the following categories: Basketball, Biking, Diving, Fencing, Golf Swing, Horse Riding, Pull-up, Push-up, Rock Climbing Indoor, and Walking with Dog. The results of their classification ratio based on the points of interest using the HOG/HOF features achieved a rate of 66.3 %. In the selected categories from the UCF50 dataset, such as Baseball Pitch, Basketball, Billiards, Breast Stroke, Fencing, Military Parade, Punch, Rock Climbing Indoor, Skiing, and Tennis Swing, the proposed method shows increased classification accuracy by approximately 12 % in comparison to [21]. One can notice that our selection categories tend to capture more information from the foreground and the background characteristics rather than motion patterns, as used in [21].

Table 1 Classification accuracy of BoR and other baseline based methods with three types of local descriptors

	PCA-SIFT+color	SIFT+color	SURF+color
keyframe	64.67	65	67.33
LRS	65.33	64.67	71.67
BoR	66	67.33	78.67

Table 2 Precision and recall rates for the UCF50 dataset: (a) color features only, (b) SURF descriptors only, and (c) combination of color and SURF descriptors with tuning weight (p, precision; r, recall)

	BP	BB	BI	BS	FE	MP	PU	RC	SK	TS	*precision*
BP	27	3	0	0	0	0	0	0	0	0	90.0
BB	6	18	0	0	1	0	3	2	0	0	60.0
BI	0	0	30	0	0	0	0	0	0	0	100.0
BS	0	1	0	25	2	0	0	1	1	0	83.33
FE	4	0	0	1	25	0	0	0	0	0	83.33
MP	8	2	1	0	2	8	6	2	1	0	26.67
PU	3	0	2	0	1	0	23	1	0	0	76.67
RC	2	2	0	0	0	0	0	26	0	0	86.67
SK	0	7	0	5	1	0	0	2	15	0	50.0
TS	7	5	0	0	1	1	0	3	1	12	40.0
recall	47.37	47.37	90.90	80.65	75.76	88.89	71.88	70.27	83.33	100.0	**75.64(r)/66.67(p)**

(a) Color features only

	BP	BB	Bl	BS	FE	MP	PU	RC	SK	TS	*precision*
BP	1	0	5	0	0	12	5	5	2	0	3.33
BB	0	2	6	1	1	5	6	3	6	0	6.67
BI	0	0	29	0	0	0	1	0	0	0	96.67
BS	0	0	0	10	0	1	1	6	12	0	33.33
FE	0	0	8	0	5	10	1	5	1	0	16.67
MP	0	0	0	0	0	27	3	0	0	0	90.0
PU	0	0	1	0	0	1	26	1	1	0	86.67
RC	0	0	1	0	0	0	2	26	1	0	86.67
SK	0	0	4	2	0	0	2	1	21	0	70.0
TS	0	0	1	0	0	6	9	4	9	1	3.33
recall	100.0	100.0	52.73	76.92	83.33	43.55	46.43	50.98	39.62	100.0	**69.36(r)/49.33(p)**

(b) SURF descriptors only

	BP	BB	BI	BS	FE	MP	PU	RC	SK	TS	*precision*
BP	26	2	0	0	0	2	0	0	0	0	86.67
BB	1	12	1	1	0	2	5	6	2	0	40.0
BI	0	0	30	0	0	0	0	0	0	0	100.0
BS	0	1	0	28	0	0	0	0	1	0	93.33
FE	1	0	0	1	21	2	0	5	0	0	70.0
MP	0	0	0	0	0	28	2	0	0	0	93.33
PU	0	0	0	0	0	0	28	2	0	0	93.9
RC	0	0	0	0	0	0	0	30	0	0	100.0
SK	0	1	1	3	0	0	2	2	21	0	70.0
TS	1	2	0	0	0	2	6	3	4	12	40.0
recall	89.66	66.67	93.75	84.85	100.0	77.78	65.12	62.5	75.0	100.0	**81.53(r)/78.67(p)**

(c) Combination of color features and SURF descriptors with tuning weight ($\gamma = 0.67$)

The BoR based method not only outperforms the keyframe and other state-of-the-art methods in terms of classification accuracy, but also achieves lower computational complexity in full frame video processing. In traditional methods, for a video clip with N number of frames the processing time typically requires $O(N)$. However, the proposed method reduces the computational complexity up to $O(R)$ by only considering representative regions for classification purpose, where R is the number of selected regions. Therefore, the computational cost would be further improved for a video shot with larger number of frames.

Table 3 Classification accuracy of combined features. Note that we set the weights of color, SURF, STIP/HoG and STIP/HoF to 0.26, 0.35, 0.1, and 0.29, respectively

Method(features)	precision
BoR(PCA-SIFT+color)	66
keyframe(SURF+color)	67.33
BoR(SIFT+color)	67.33
LRS(SURF+color)	71.67
STIP(HoG/HoF) [42]	72.67
BoR(SURF+color)	78.67
BoR(SURF+color)+STIP(HoG/HoF)	**84.67**

Table 4 Precision and recall rates for the UCF50 dataset after combining the BoR and STIP features with turning weight as shown in Table 3.3 (p, precision; r, recall)

	BP	BB	BI	BS	FE	MP	PU	RC	SK	TS	*precision*
BP	25	0	0	0	0	0	2	3	0	0	83.33
BB	0	19	0	1	1	0	4	5	0	0	63.33
BI	0	0	30	0	0	0	0	0	0	0	100.0
BS	0	0	0	30	0	0	0	0	0	0	100.0
FE	0	0	0	0	22	0	6	2	0	0	73.33
MP	0	0	0	0	0	26	3	1	0	0	86.67
PU	0	0	0	0	0	0	30	0	0	0	100.0
RC	0	0	0	0	0	0	0	30	0	0	100.0
SK	0	0	0	2	0	0	1	2	25	0	83.33
TS	0	2	0	0	0	2	6	3	0	17	56.67
recall	100.0	90.48	100.0	90.9	95.65	92.86	57.69	65.22	100.0	100.0	**89.28(r)/84.67(p)**

3.3 Combining with motion features

Color and local interest point features based on the BoR reflect spatial information in the selected regions in temporal domain. In this section, we combine spatial features based on BoR with temporal features based on STIP (spatio-temporal interest points) [42]. Table 3 shows the classification results with each combined features. The highest precision was achieved as the rate of 84.67 % when two methods are linearly combined as shown in (2):

$$Z = \gamma \cdot Color + (1 - \gamma) \cdot LIPD + \alpha \cdot SHoG + \beta \cdot SHoF. \tag{3}$$

Table 4 shows the confusion matrix of the best accuracies for the BoR using the combined features. Use of STIP features with BoR achieved increased precision by 12 % and 6 % over STIP features only and BoR with the SURF descriptor, respectively. Figure 11 shows the precision/recall and F1 score by comparing different methods in each class of the UCF50 with tuned parameters and weights. Note that the F-score is calculated as follows with $\beta = 1$:

$$F_\beta = (1 + \beta^2) \cdot precision \cdot recall / (\beta^2 \cdot precision + recall)$$

Based on our experimental evaluation, the combination of two methods, BoR and STIP, achieved the best accuracies comparing with other methods.

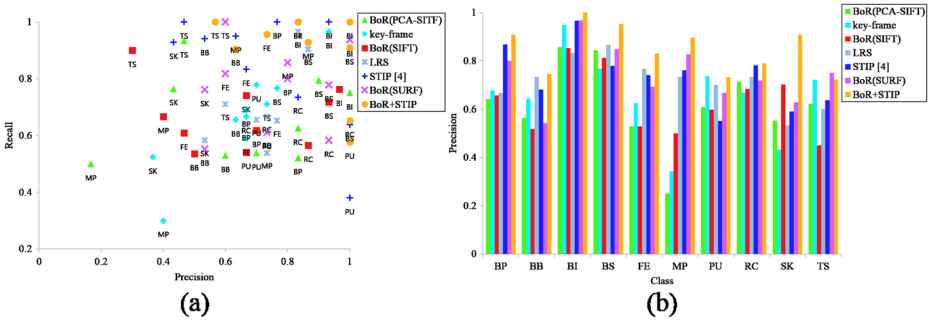


Fig. 11 **a** Precision/recall values of each class in the UCF50 dataset (the letters and the marks indicate classification categories and applied methods respectively). **b** The f1 scores in each class with tuned weight and SVM parameters

4 Conclusions

We introduced a bag-of-regions representation for high-level sparse representation of a video sequence. To represent all distinct objects appearing as TSRs within a scene, we performed region-based tracking to generate a relational graph of a video sequence, and TSRs to construct the representation of the video content. This result is a jigsaw-like representation of the video objects in each frame. Then, the propsed method adjusts to gradual changes in the illumination of the video. Even when a video clip has been recorded by a moving camera, our method can extract the main regions with semantic significance. Our test results show that this region-based classification is competitive with other state-of-the-arts and has potential application to a semantic understanding of tasks such as in motion capturing, video editing, unusual activity detection, multimedia searching, etc.

The main contributions of this paper include the following: a novel way of determining a bag-of-regions representation from a video for high-level applications; an alternative representation for existing CVBRs; and a way to reduction in the 'semantic gap' through the creation of regions with a semantic significance. In the future research, we will investigate the use of semantically significant regions in high-level applications, such as a bag-of-visual-words framework. The proposed method can be extended to project low-level features into a semantic space to support the semantic retrieval of video sequences. Other features of the representative regions, such as 3D motion blob and parts model, as well as the use of other classifiers, may also offer significant applications of this work.

Acknowledgments This work was supported by (1) Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (2012M3C4A7032781), (2) the ICT R&D program of MSIP/IITP. [2014(10047078), 3D reconstruction technology development for scene of car accident using multi view black box image], and (3) INHA UNIVERSITY Research Grant.

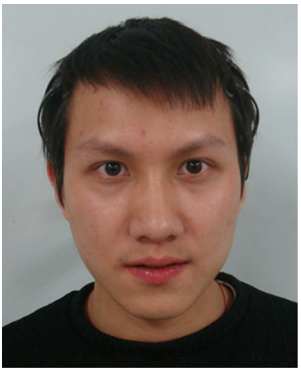
References

1. Abd-Almageed W (2008) Online, simultaneous shot boundary detection and key frame extraction for sports videos using rank tracing. In: Proceedings of International Conference on Image Processing
2. Bandla S, Grauman K (2013) Active learning of an action detector from untrimmed videos. In: Proceedings of International Conference on Computer Vision
3. Banerjee P, Sengupta S (2008) Model generation for robust object tracking based on temporally stable regions. In: IEEE Workshop on Motion and Video Computing
4. Bay H, Ess A, Tuytelaars T, Gool LV (2008) Surf: Speeded up robust features. Comp Vision Image Underst 110(3):346–359
5. Black M, Gorelick L, Shechtman E, Irani M, Basri R (2005) Actions as space-time shape. In: Proceedings of International Conference on Computer Vision
6. Blilen H, Gool LV (2011) Action recognition: A region based approach. In: IEEE Workshop on Applications of Computer Vision
7. Bojanowski P, Bach F, Laptev I, Ponce J, Schmid C, Sivic J (2013) Finding actors and actions in movies. In: Proceedings of Computer Vision and Pattern Recognition
8. Brendal W, Todorovic S (2009) Video object segmentation by tracking regions. In: Proceedings of International Conference on Computer Vision
9. Chakraborty B, Holtre MB, Moesl TB, Gonzalez J (2012) Selective space-time interest points. Comput Vis Image Underst 116:396–410
10. Chang CC, Lin CJ (2011) Libsvm: A library for support vector machine. ACM Transactions on IST 2(27):1–27. Software available at http://www.scie.ntu.edu.tw/cjlin/libsvm
11. Choi J, Wang Z, Lee SC, Jeon WJ (2013) A spatio-temporal pyramid matching for video retrieval. Comp Vision Image Underst 117:660–669

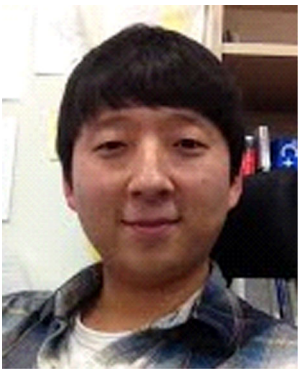
12. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: Proceedings of Computer Vision and Pattern Recognition
13. Dalal N, Triggs B, Schmid C (2006) Human detection using oriented histograms of flow and appearance. In: Proceedings of Computer Vision and Pattern Recognition
14. Demir G, Selim A (2007) Scene classification using bag-of-regions representations. In: Proceedings of Computer Vision and Pattern Recognition
15. Deng Y, Manjunath BS (2011) Unsupervised segmentation of color-texture regions in images and video. IEEE Transactions on Pattern Analysis and Machine Intelligence 23(8):800–810
16. Gonzalez RC, Woods RE Digital image processing. Pearson Prentice Hall
17. Jhuang H, Gall J, Zuffi S, Schmid C, Black MJ (2013) Towards understanding action recognition. In: Proceedings of International Conference on Computer Vision
18. Junejo IN, Dexter E, Laptev I, Perez P (2010) View-independent action recognition from temporal self-similarities. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(1):172–185
19. Ke Y, Sukthankar R (2004) Pca-sift: A more distinctive representation for local image descriptors. In: Proceedings of Computer Vision and Pattern Recognition
20. Kovashka A, Grauman K (2010) Learning a hierarchy of discriminative space-time neighborhood features for human action recognition. In: Proceedings of Computer Vision and Pattern Recognition, pp 2046–2053
21. Kuehne H, Jhuang H, Garrote E, Poggio T, Serre T (2011) Hmdb: A large video database for human motion recognition. In: Proceedings of International Conference on Computer Vision
22. Laptev I (2005) On space-time interest points. Int J Comput Vis 108:207–229
23. Lazebnik S, Schmid C, Ponce J (2006) Finding actors and actions in movies. In: Proceedings of Computer Vision and Pattern Recognition
24. Leutenegger S, Chli M, Siegwart RY (2011) Brisk: Binary robust invariant scalable keypoints. In: Proceedings of International Conference on Computer Vision
25. Li LJ, Fei LF (2007) What, where and who? classifying events by scene and object recognition. In: Proceedings of International Conference on Computer Vision
26. Liu J, Luo J, Shah M (2009) Recognizing realistic actions from videos “in the wild”. In: Proceedings of Computer Vision and Pattern Recognition
27. Lowe DG (2005) Object recognition from local scale-invariant features. In: Proceedings of International Conference on Computer Vision
28. Makadia A (2010) Feature tracking for wide-baseline image. In: Proceedings of European Conference on Computer Vision, pp 310–323
29. Marszalek M, Latev I, Schmid C (2009) Action in context. In: Proceedings of Computer Vision and Pattern Recognition
30. Masoud O, Papanikolopoulos NP (2001) A novel method for tracking and counting pedestrians in real-time using a single camera. Veh Technol 50(5):1267–1278
31. McCandless T, Grauman K (2013) Object-centric spatio-temporal pyramids for egocentric activity recognition. In: Proceedings of British Machine Vision Conference
32. Niebles J, Chen C, Fei LF (2010) Modeling temporal structure of decomposable motion segments for activity classification. In: Proceedings of European Conference on Computer Vision
33. Pirsiavash H, Ramanan D (2012) Detecting activities of daily living in first-person camera view. In: Proceedings of Computer Vision and Pattern Recognition
34. Raptis M, Sigal K (2013) Poselet key-framing: A model for human activity recognition. In: Proceedings of Computer Vision and Pattern Recognition
35. Schuldt C, Laptev I, Caputo B (2004) Recognizing human action: A local svm approach. In: Proceedings of International Conference on Pattern Recognition
36. Sivic J, Schaffalitzky F (2006) Object level grouping for video shots. Int J Comput Vis 67(2):189–210
37. Sivic J, Zisserman A (2003) Video google: A text retrieval approach to object matching in videos. In: Proceedings of International Conference on Computer Vision
38. Turcot P, Lowe DG (2009) Better matching with fewer features: The selection of useful features in large database recognition problems. In: International Conference of Computer Vision Workshops, pp 2109–2116
39. Ullah MM, Laptev I (2012) Actlets: A novel local representation for human action recognition in video. In: Proceedings of International Conference on Image Processing, pp 777–780
40. Umamakeswari A, Rajaraman A (2007) Object based video analysis, interpretation and tracking. J Comput Sci 3(10):818–822

41. Vapnik V (1998) Statistical learning theory. Wiley, Hoboken
42. Wang H, Ullah MM, Klaser A, Laptev I, Schmid C (2009) Evaluation of local spatio-temporal features for action recognition. In: Proceedings of British Machine Vision Conference
43. Weinland D, Boyer E, Ronfard R (2007) Action recognition from arbitrary views using 3d exemplars. In: Proceedings of International Conference on Computer Vision
44. Yao B, Jiang X, Khosla A, Lin AL, Guibas L, Fei LF (2011) Human action recognition by learning bases of action attributes and parts. In: Proceedings of International Conference on Computer Vision
45. Ying L, Der WT, Neng HJ (2003) Object-based analysis and interpretation of human motion in sports video sequences by dynamic bayesian networks. Comp Vision Image Underst 92:196–216

Min-Kook Choi is a Ph.D. candidate in the Department of Computer and Information Engineering at Inha University, Incheon, Korea. Choi's research interest is in machine learning for computer vision and video processing.



Ziyu Wang was a graduate student in the Department of Computer and Information Engineering at Inha University, Incheon, Korea. Wang's research interest is in image and video processing.



Hyun-Gyu Lee is a Ph.D. candidate in the Department of Computer and Information Engineering at Inha University, Incheon, Korea. Lee's research interest is in medical image analysis, computer vision and image processing.

Sang-Chul Lee is an associative professor in the Department of Computer and Information Engineering at Inha University, Incheon, Korea. Prof. Lee's scientific interest is in computer vision, image processing and multimedia information retrieval.