

Received June 2, 2020, accepted June 29, 2020, date of current version July 16, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3007150

Multi-Sensor Integration for Key-Frame Extraction From First-Person Videos

YUJIE LI^{1,2}, (Member, IEEE), ATSUNORI KANEMURA^{2,3,4}, (Member, IEEE),
HIDEKI ASOH², (Member, IEEE), TAIKI MIYANISHI^{4,5}, AND MOTOAKI KAWANABE^{4,5}

¹School of Artificial Intelligence, Guilin University of Electronic Technology, Guilin 541004, China

²National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba 305-8560, Japan

³LeapMind Inc., Tokyo 150-0044, Japan

⁴Advanced Telecommunications Research Institute International (ATR), Kyoto 619-0288, Japan

⁵The RIKEN Center for Advanced Intelligence Project (AIP), Kyoto 619-0288, Japan

Corresponding author: Atsunori Kanemura (atsu-kan@leapmind.io)

This work was supported in part by Japan Society for the Promotion of Science (JSPS), KAKENHI under Grant 18K18083, in part by the New Energy and Industrial Technology Development Organization (NEDO), Japan, in part by the Dean Project of Guangxi Wireless Broadband Communication and Signal Processing Key Laboratory, and in part by National Natural Science Foundation of China, under Grant 61903090.

ABSTRACT Key-frame extraction for first-person vision (FPV) videos is a core technology for selecting important scenes and memorizing impressive life experiences in our daily activities. The difficulty of selecting key frames is the scene instability caused by head-mounted cameras used for capturing FPV videos. Because head-mounted cameras tend to frequently shake, the frames in an FPV video are noisier than those in a third-person vision (TPV) video. However, most existing algorithms for key-frame extraction mainly focus on handling the stable scenes in TPV videos. The technical development of key-frame extraction techniques for noisy FPV videos is currently immature. Moreover, most key-frame extraction algorithms mainly use visual information from FPV videos, even though our visual experience in daily activities is associated with human motions. To incorporate the features of dynamically changing scenes in FPV videos into our methods, integrating motions with visual scenes is essential. In this paper, we propose a novel key-frame extraction method for FPV videos that uses multi-modal sensor signals to reduce noise and detect salient activities via projecting multi-modal sensor signals onto a common space by canonical correlation analysis (CCA). We show that the two proposed multi-sensor integration models for key-frame extraction (a sparse-based model and a graph-based model) work well on the common space. The experimental results obtained using various datasets suggest that the proposed key-frame extraction techniques improve the precision of extraction and the coverage of entire video sequences.


INDEX TERMS Video summarization, multi-sensors, key-frame extraction, sparse estimation, graph model.

I. INTRODUCTION

First-person vision (FPV) videos captured by head-mounted wearable cameras are useful for understanding daily life activities [1], [2]. FPV videos often contain important scenes worth remembering in our daily lives. Summarizing such salient scenes is essential because FPV videos tend to be redundant [2]. However, FPV videos are unstable and noisy compared to third-person view (TPV) videos, and most existing methods of video summarization mainly focus on handling the stable scenes in a TPV video [3]–[19]. Moreover, the following differences between FPV and TPV videos substantially complicate summarizing FPV videos compared to summarizing TPV videos. (i) *Camera placement*: FPV videos are captured from the wearer's viewpoint (e.g., chest and head), whereas TPV videos are captured from a fixed point

of view. (ii) *Intention*: Although TPV videos are intentionally recorded by the photographer, FPV videos are recorded regardless of his/her intention. This unconstrained FPV video often contains insignificant objects, such as a ceiling or a floor. (iii) *Content*: TPV videos record experiences worth remembering through a manual operation that focuses on specific interesting scenes. FPV videos record natural scenes of life, which may contain repetitive video shots that are irrelevant to our interests. (iv) *Quality*: Whereas TPV videos contain stable frames, FPV videos tend to contain blurry and shaky frames due to the wearer's body motion. Therefore, TPV summarization techniques applied to noisy FPV videos perform inaccurately and even worse than uniform sampling [2]. To obtain high-quality FPV video summaries, we must address these issues.

In this paper, we present a key-frame extraction method for FPV videos with multi-sensor signals. To reliably select key frames, our method uses motion signals as the extra

The associate editor coordinating the review of this manuscript and approving it for publication was Cesar Vargas-Rosales .

sensor information beyond video frames, while most existing methods use only video information [3]–[19]. We assume that motion information expresses the detailed hand or head movement that visual information does not capture. To associate their features, we embed multi-sensor data into a common vector space [20]–[27] using probabilistic canonical correlation analysis (PCCA) [28]. The projection matrices learned by PCCA ensure that the relevant pairs of information are close. Moreover, we propose two key-frame extraction algorithms that are performed on this learned space. First, we use a sparse key-frame selection method based on a sparse measure, the l_1 -norm, and extend it with multi-sensor integration. Second, we use a key-frame extraction approach based on a probabilistic graphical model (referred to as a graph model) employing conditional random fields (CRFs) [29] for multi-sensor integration. We show that the proposed multi-sensor integration is effective for key-frame extraction from FPV videos under both sparse-based and graph-based models.

This paper is an extension of our conference publications [30], [31] with significant modifications. Two major differences are as follows: 1) We introduce a graph-model-based method and a sparse-model-based method to extract key frames from FPV videos. Therefore, the proposed multi-sensor integration can improve the key-frame extraction performance across different methods. 2) We expand the experimental results not only by adding more videos to the dataset used in the conference papers but also by introducing another new dataset and quantitative comparisons with the existing methods.

II. RELATED WORKS

Key frames are a group of frame images selected from different scenes in a video and presented in temporal order [4], [32]. Although there are several mathematical definitions of key frames, these definitions commonly model key frames as the most representative and informative frames, reflecting the most important contents in a video [4]–[6], [14]–[16].

1) KEY-FRAME EXTRACTION

Many key-frame extraction methods have been proposed in the literature [3]–[16]. Liu *et al.* [3] presented an algorithm based on the maximum a posteriori (MAP) method to detect key frames. Ejaz *et al.* [4] developed an integration scheme to combine the image features obtained from the correlation of RGB colour features, a colour histogram, and moments of inertia to select the key frames. Elhamifar *et al.* [6] proposed a sparse modelling representation selection (SMRS), which is an efficient algorithm for video classification and summarization. SMRS employs a sparse-coding-based framework with the l_1 -norm as a sparsity constraint. However, the direct utilization of SMRS estimated the null-information frames because of noise and instability in FPV videos [30], [31]. The proposed technique develops SMRS for better key-frame selection from FPV videos.

2) SPARSE REPRESENTATION

Sparse representation is undoubtedly a common model of sparse signals [33]. There are many applications, such as compressive sensing [34], denoising, sampling, classification, superresolution, inpainting, and deblurring, that employ the sparse representation theory and model as fundamentals. In the literature, sparse representation has been further proven to be an extremely powerful tool for representing, analysing, and compressing signals [33], [35]. Aiming for sparsity, most sparse representations employ the l_0 -norm [33], [35], [36] or the l_1 -norm as the sparsity constraint [37], [38]. In this paper, we also use the l_1 -norm as a sparse measure to reduce noise and detect salient activities in first-person videos.

3) GRAPH MODEL

A graph model is a probabilistic model for which a graph expresses the conditional dependence structure between random variables. They are generally used in pattern recognition and machine learning [39]. Two branches of graph models that are generally used are Markov random fields and Bayesian networks. An increasing number of publications in computer vision use graph-based energy minimization techniques for image processing applications, such as segmentation [40], [41], image restoration [42], stereo [43], [44], shape reconstruction [45], object recognition [46], texture synthesis [47], and socialized group photography [48]. For example, Ngo *et al.* [49] proposed video summarization methods and scene detection algorithms based on graph modelling. In their methods, a video is expressed as a complete undirected graph, and the normalized cut algorithm is applied to globally and optimally divide the graph into video clusters. Molino *et al.* [50] used a probabilistic approach based on active inference in CRFs, which is a type of discriminative undirected probabilistic graphical model, for active video summarization. In contrast to the existing graph-based approaches that only use video information, our approach additionally uses sensor information as well as video information for accurately modelling daily living activities.

III. MODEL AND FORMULATION

In this work, we propose a multi-sensor integration-based key-frame extraction method for FPV videos. First, we focus on applying the sparse model to select key frames using multi-sensor integration. Second, we use the graph model to select the key frames from FPV videos. Our proposed multi-sensor integration method can be applied to any key-frame extraction algorithm. However, in this paper, we choose two examples: sparse-model-based and graph-model-based algorithms.

A. VIDEO FEATURE SELECTION

First, we extract all the video frames from the raw video and learn the deep semantic features by adopting a pre-trained DNN (e.g., VGG). Inspired by previous work [51], we represent the input video frames as the deep semantic features of the semantic space, and every feature corresponds to a frame.

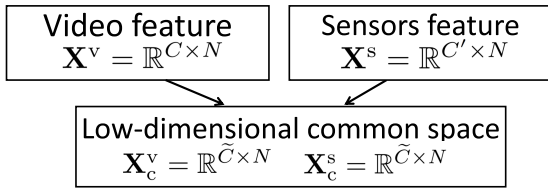


FIGURE 1. The graphical model for canonical correlation analysis.

This method encodes the semantic transition of videos. Thus, it is effective for many video processing applications, such as video description, video generation and video retrieval. Some video clusters can be estimated, each of which is predicted to involve similar frames. With this assumption, we estimate a cluster of frames by solving an optimization formulation of the video representation. We use frame deep features learned from the DNN rather than natural video frame images. For feature extraction, we employ a pre-trained VGG network [52], which produces discriminative visual features. Note that the video features in the following sections of this paper refer to the features extracted from VGG. In this way, we convert each video frame into a 1000-dimensional vector. After extracting the VGG features from all N frames, we construct a dictionary matrix, where each row represents N frames and each column contains the 1000-dimensional frame feature vector.

B. PROJECTION WITH MULTI-SENSOR INTEGRATION

We employ PCCA to embed the multi-sensor integration data (video and motion) into a common space (Fig. 1) [30]. Let $\mathbf{X}^v = \mathbb{R}^{C \times N}$ represent video data and $\mathbf{X}^s = \mathbb{R}^{C' \times N}$ represent motion sensor data, where N is the number of frames, C is the video feature dimensionality, and C' is the motion sensor feature dimensionality. In this paper, the sensor features consist three-axis acceleration (the rate of change in the velocity of an object) obtained by three-axis accelerometers, and rotational changes or maintaining orientation is obtained by gyroscopes. In general, motion data do not have units of frames; more often, they have units of seconds or another time unit. To fuse motion data with video data, we first synchronize the two modalities and then often perform a sampling of the motion data in units of video frames. Linear projections \mathbf{A}^v and \mathbf{A}^s from the video and sensor domains to a common space can be generated by the following formulation:

$$\min_{\mathbf{A}^v, \mathbf{A}^s} \|\mathbf{A}^v \mathbf{X}^v - \mathbf{A}^s \mathbf{X}^s\|_F^2, \tag{1}$$

where $\mathbf{A}^v \in \mathbb{R}^{\tilde{C} \times C}$ and $\mathbf{A}^s \in \mathbb{R}^{\tilde{C} \times C'}$ are linear projectors from the video feature domain and the sensor domain, respectively, to the common space with the same dimension, and $\|\cdot\|_F$ is the Frobenius norm [53]. The optimal projection matrices \mathbf{A}^v and \mathbf{A}^s are estimated from the solutions of an eigenvalue problem. After learning the projection matrices \mathbf{A}^v and \mathbf{A}^s , we can use them to project data vectors from the video and the sensor domains into the \tilde{C} -dimensional common space, where the corresponding sets of information are similar [30]. Noting that $\tilde{C} < C$ and $\tilde{C} < C'$, it realizes the reduced dimension of the common space after PCCA.

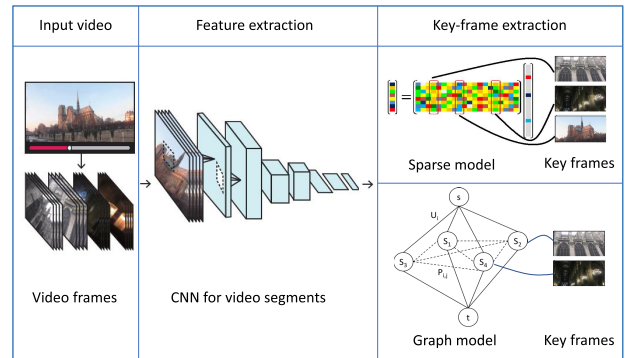


FIGURE 2. The framework of the sparse-model-based and graph-model-based video feature selection using only video information.

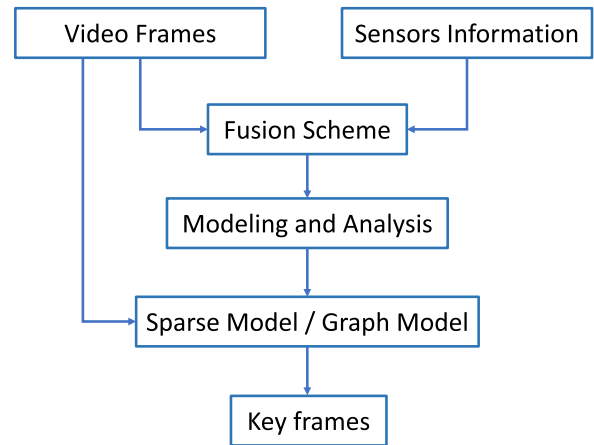


FIGURE 3. The framework of the video-summarization-based sparse model and graph model with multi-sensor integration.

The common space video features $\mathbf{X}_c^v = \mathbf{A}^v \mathbf{X}^v$ and the common space sensor features $\mathbf{X}_c^s = \mathbf{A}^s \mathbf{X}^s$ are spliced to be the integrated feature matrix $\mathbf{X} = [\mathbf{X}_c^v, \mathbf{X}_c^s]$.

C. SPARSE-MODEL-BASED KEY-FRAME EXTRACTION

First, we propose our multi-sensor integration model for extracting a key frame based on a sparse model. Fig. 2 shows the framework of our approach for video key-frame extraction based on the sparse model. A model for signals formulates a mathematical description of the group of signals, which allows them to be distinguished from the remaining signal space. A linear representation model has been developed and has recently received appreciable attention [54], [55]. Signals can be expressed as linear combinations of the representative signals. This can be formulated as a problem of finding the representative signals as a sparse multiple measurement matrix problem [6]. The sparse modelling method [33], [56]–[58] is the most effective representative methodology of all linear representation algorithms. The aim of sparse modelling is to approximate a natural signal by a linear expression of dictionary atoms. The signal is then represented as linear combinations of a few dictionary atoms. Fig. 3 shows the proposed framework of the video summarization.

To incorporate sparse representation into key-frame extraction, a modification was considered for to the dictionary learning problem, which first addresses the optimization of local minimum due to the generation of two

unknown matrices, namely, the sparse coefficient and the dictionary matrices. This enforces learning sparse representations from natural signals [6]. For this purpose, the formulation of sparse-representation-based key-frame extraction can be written as follows:

$$\min_{\mathbf{H}} \|\mathbf{X} - \mathbf{X}\mathbf{H}\|_F^2 \quad \text{s.t.} \quad \|\mathbf{H}\|_{0q} < s. \quad (2)$$

Here, the ℓ_{0q} norm is expressed as:

$$\|\mathbf{H}\|_{0q} = \sum_{i=1}^N \mathbf{I}(\|\mathbf{h}_i\|_q > 0). \quad (3)$$

Here, \mathbf{h}_i is the i th row of the matrix \mathbf{H} , and \mathbf{I} is the indicator function. Generally, $q = 2$ is the l_2 norm. $\|\mathbf{H}\|_{0q}$ counts the rows of nonzeros in the sparse coefficient matrix \mathbf{H} . The index of the nonzero rows of \mathbf{H} corresponds to the index of the columns of \mathbf{X} , which is selected as the signal representation. The indices of the zero rows of \mathbf{H} are redundant frames, which are neighbours of key frames. We select nonzero rows to represent key frames and discard the redundant and irrelevant frames. It is preferable that the extraction of the representation is invariant with respect to the global translation of the signal. Thus, we enforced the affine constraint $\mathbf{1}^T \mathbf{H} = \mathbf{1}^T$. Because the problem of the l_0 norm is NP-hard, we introduced the l_1 norm as a relaxation of this NP-hard problem. The l_1 norm is the sum of the elements of a vector. The proposed objective formulation can be written as:

$$\min_{\mathbf{H}} \|\mathbf{X} - \mathbf{X}\mathbf{H}\|_F^2 + \alpha \|\mathbf{H}\|_{1q} \quad \text{s.t.} \quad \mathbf{1}^T \mathbf{H} = \mathbf{1}^T. \quad (4)$$

Here, $\|\mathbf{H}\|_{1q}$ is expressed as:

$$\|\mathbf{H}\|_{1q} = \sum_{i=1}^N \|\mathbf{h}_i\|_q. \quad (5)$$

To normalize the rows of \mathbf{H} as the l_2 norm, we take $q = 2$. The final objective formulation can be written as:

$$\min_{\mathbf{H}} \|\mathbf{X} - \mathbf{X}\mathbf{H}\|_F^2 + \alpha \|\mathbf{H}\|_{1,2} \quad \text{s.t.} \quad \mathbf{1}^T \mathbf{H} = \mathbf{1}^T, \quad (6)$$

where

$$\|\mathbf{H}\|_{1,2} = \sum_{i=1}^N \|\mathbf{h}_i\|_2. \quad (7)$$

D. FACTOR-GRAPH-BASED KEY-FRAME EXTRACTION

Second, we propose our multi-sensor integration model for extracting key frames based on graph models. The frameworks using only video information and multi-sensor integration are shown in Fig. 2 and Fig 3. Methods that solve complex global functions of variables always employ the given function's factor as an output of "local" functions, and each function depends on a subset of the variables. This factorization can be expressed by a structure graph, which is called a factor graph [59].

Let $s = \{0, 1\}^N$ be a vector with binary values that represent the summary of frames from the FPV video, where s_i is equal to 1 when the i th frame is selected as a key frame

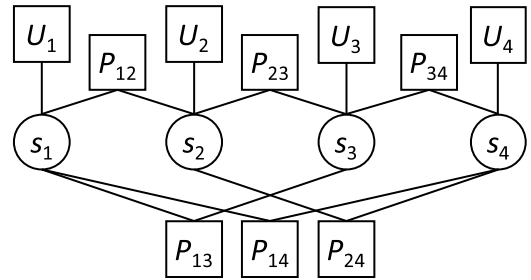


FIGURE 4. Energy interactions in the four-frame graph model. U_i and P_{ij} are shorthand for $U_{\theta}(s_i)$ and $P_{\theta}(s_i, s_j)$, respectively.

and 0 when the i th frame is NOT selected as a key frame. $p(s|\theta)$ is denoted as the probability density distribution of how likely the selected frame s is to be selected as a key frame. We select the frames of $s_i = 1$ and omit the frames of $s_i = 0$ to discard the redundant and irrelevant frames. We modelled this distribution by CRF, and $\theta = [\theta_0, \theta_1, \alpha, \gamma, \beta]$ are the values of its parameters to be defined later in this subsection.

A CRF models the probability density with a Gibbs distribution [50], [60]. Thus, $p(s|\theta)$ can be expressed as the normalized exponential of an energy function, which is denoted as $E_{\theta}(s): p(s|\theta) \propto \exp[-E_{\theta}(s)]$. The summary of the key frames, denoted as s^* , is generated by solving the MAP as follows:

$$s^* = \arg \max_s p(s|\theta) = \arg \min_s E_{\theta}(s). \quad (8)$$

We define the energy function as follows:

$$E_{\theta}(s) = \lambda \sum_i U_{\theta}(s_i) + \sum_{i,j} P_{\theta}(s_i, s_j). \quad (9)$$

Here, the unary potential $U_{\theta}(s_i)$ enforces the selection of static frames, the pairwise potential $P_{\theta}(s_i, s_j)$ encourages frames with diverse semantic content, and $\lambda > 0$ is a parameter that weights the unary and pairwise potentials. Taking four frames (s_1, s_2, s_3, s_4) as an example, we illustrate the unary and pairwise interactions as a graph in Fig. 4.

A directed weighted graph includes a group of nodes and a group of directed edges that connect the nodes. Generally, the nodes represent pixels, frames, or other features. A graph normally contains two special nodes referred to as the source s and the sink t ; thus, it is called an s - t graph. In the context of vision, terminals correspond to the group of labels that can be assigned to pixels [61]. In our situation, we will focus on the case of the graph with two terminals: the key frame and not a key frame, which is expressed in Fig. 5.

The unary potential, $U_{\theta}(s_i)$, defines the baseline to be selected as a key frame. We model

$$U_{\theta}(s_i) = \theta_q I[s_i = 1] + \theta_p I[s_i = 0], \quad (10)$$

where $I[Q]$ is an indicator function. $I[Q] = 1$ if Q is true and $I[Q] = 0$ otherwise, and θ_q and θ_p are constants that balance the ratio of key frames and other frames.

The pairwise potential, $P_{\theta}(s_i, s_j)$, is defined between each pair of similar frames and enforces selecting frames with diverse contents. Let $d(\psi_i, \psi_j)$ be the Euclidean distance

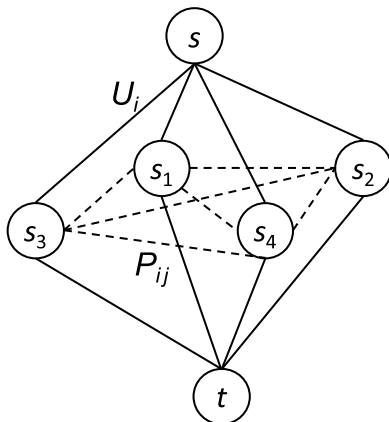


FIGURE 5. The s - t graph for key-frame selection by the min-cut/max-flow.

between the features of two frames i and j , expressed as follows:

$$d(\psi_i, \psi_j) = \|\psi_i - \psi_j\|_2. \quad (11)$$

The pairwise potential enforces that similar frames should not be selected for the summary. For this purpose, we define a potential that is weighted by the distance between features, shown as follows:

$$P_\theta(s_i, s_j) = \exp\{-d(\psi_i, \psi_j)\}P'_\theta(s_i, s_j). \quad (12)$$

Here, $P'_\theta(s_i, s_j)$ suggests that both frames s_i, s_j should not be selected at the same time, and the term $\exp\{-d(\psi_i, \psi_j)\}$ reduces the effort of $P'_\theta(s_i, s_j)$ when the frames are dissimilar. The value of the potential, $P_\theta(s_i, s_j)$, is smaller when the frames s_i, s_j are dissimilar. Specifically, $P'_\theta(s_i, s_j)$ is defined as follows:

$$P'_\theta(0, 0) = \theta_0\alpha, \quad P'_\theta(0, 1) = \gamma, \quad (13)$$

$$P'_\theta(1, 0) = \gamma, \quad P'_\theta(1, 1) = -\theta_0\beta. \quad (14)$$

Thus, the optimal solution for key-frame extraction can be obtained by minimizing the potential as follows:

$$\begin{aligned} s^* &= \arg \min_{\theta} E_\theta(s) \\ &= \arg \min_{\theta} \lambda \sum_i U_\theta(s_i) + \sum_{i,j} P_\theta(s_i, s_j) \\ &= \arg \min_{\theta} \lambda \sum_i (\theta_q I[s_i = 1] + \theta_p I[s_i = 0]) \\ &\quad + \sum_{i,j} \exp\{-d(\psi_i, \psi_j)\}P'_\theta(s_i, s_j) \end{aligned} \quad (15)$$

We use a general optimization framework of trust-region-based local submodular approximations (LSA-TR) [62] to solve problem (15). The local submodular approximations (LSA) approach constructs an approximation model without additional variables and uses a more accurate approximation. Trust region (TR) methods are a class of iterative optimization algorithms. The model is only accurate within a small region around the current solution called the “trust region”, and the approximate model is

Algorithm 1 Sparse-Model-Based Key-Frame Extraction (SMFE)

Require: Signal matrices \mathbf{X}^V from VGG and \mathbf{X}^S from sensors

- 1: Normalize the columns of the signal \mathbf{X}^V and \mathbf{X}^S to a unit l_2 -norm.
- 2: Embed into a multi-information matrix by PCCA.
- 3: Set the regularization parameters.
- 4: Initialize \mathbf{H} as a random matrix.
- 5: Execute SMRS [6] with ADMM to estimate the indices of the key frames from the FPV video.

Algorithm 2 Graph-Model-Based Key-Frame Extraction (GMFE)

Require: Data matrices \mathbf{X}^V from VGG and \mathbf{X}^S from sensors

- 1: Normalize the columns of the data \mathbf{X}^V and \mathbf{X}^S to a unit l_2 -norm.
- 2: Embed into a multi-information data matrix by PCCA.
- 3: Calculate the size of the $2 \times N$ array of unary terms (N is the number of frames in the video).
- 4: Calculate the size of the $M \times 6$ array, which is a list of M arbitrary pairwise potentials. Each row in this pairwise potential list is of the format $[i, j, P'_\theta(0, 0), P'_\theta(0, 1), P'_\theta(1, 0), P'_\theta(1, 1)]$, where i and j are neighbours and the four coefficients define the interaction potential.
- 5: Execute LSA-TR [62] to estimate the indices of the key frames of the FPV video.

then globally optimized within the trust region to obtain a candidate solution.

IV. ALGORITHMS

A. SPARSE MODEL

This section describes the proposed algorithm for summarization from FPV videos with multi-sensor integration based on SMRS [6]. The coding matrix of SMRS is computed using data self-representativeness (the dictionary is set by the video signals themselves) adopting block sparsity regularization. We employ the alternating direction method of multipliers (ADMM) optimization scheme. The corresponding algorithm is described in Algorithm 1: Sparse-model-based key-frame extraction (SMFE). We used the existing implementation of SMRS¹ for our method.

B. GRAPH MODEL

Next, we summarize the proposed algorithm for summarizing an FPV video through multi-sensor integration based on a graph model. We employ a min-cut/max-flow optimization framework to optimize the corresponding objective function. The corresponding algorithm is described in Algorithm 2: Graph-model-based key-frame extraction (GMFE). We use

¹<http://www.ccs.neu.edu/home/eelhami/codes.html>

Gorelick *et al.* [62]’s implementation of local submodular approximations-trust region (LSA-TR).²

V. EXPERIMENTAL SETTINGS

We evaluate our proposed key-frame extraction methods using human activity datasets captured in a house. In the following section, we present these datasets in detail.

A. DATASETS

1) CMU-MMAC

The Carnegie Mellon University Multimodal Activity (CMU-MMAC) database [63] is designed to overcome some of the previous limitations by collecting multi-modal (e.g., video, audio, motion capture, and accelerations) signals of human activity. To collect human activity in an environment that is as natural as possible, researchers have installed a nearly fully operable kitchen and collected the preparation of some meals from the beginning to the end. A Firewire camera, FL2-08S2C, is worn on the head of the subject. Accelerometer and gyroscope information is collected with MicroStrain’s 3DM-GX1 inertial measurement units.

There are five datasets that consist of cooking five different recipes in the CMU-MMAC database: brownie, salad, pizza, scrambled eggs, and sandwich. Because only the brownie dataset has labels, we use the brownie dataset in our paper. There are 13 videos in the brownie dataset, from B07 to B24.

2) DAILY ACTIVITIES

We used another non-public dataset collected by Miyanishi *et al.* [64], which we call the daily activities dataset in this paper. This dataset collects the daily activities of 8 persons (not the researchers), whose ages ranged from 21 to 26 years (mean = 23.13, SD = 1.69). These subjects wore wearable motion sensors containing a wearable camera, three-axis accelerometers, and gyroscopes. The subjects executed 20 daily actions at various locations following written instructions on a worksheet without direct supervision from the experimenters. For instance, he/she “washes dishes” in the kitchen and “drinks tea” in the living room. For each person, there are several sessions containing different actions performed. The recorded sensor signals consist of 17-h videos and motion data of approximately 20 actions. The proposed algorithm selects key frames from these FPV videos using not only video but also motion information [64].

The order of locations where subjects performed their daily activities is shuffled in each session. There is a room layout of the experimental environments and lists the 20 daily activities at each location performed by the subjects in each session of the with-object task. A single session averaged 10.86 min (SD = 1.14) among the subjects. The sessions were repeated 12 times (including two initial practice sessions), and short breaks were allowed. There was no researcher to supervise the subjects while collecting data under the semi-natural collection protocol. The researchers used the motion and video data from the 3rd to 12th sessions of the with-object task

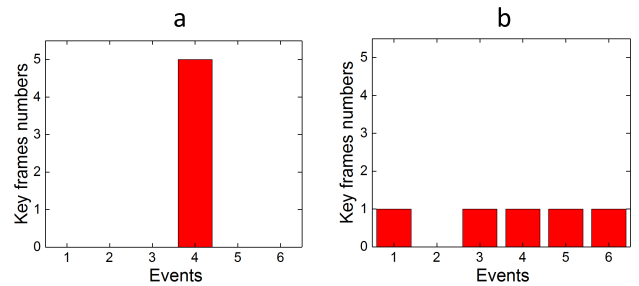


FIGURE 6. Two cases for the same accuracy value.

as the search target. After the with-object task, to collect gesture motions for retrieving past activities, the subjects were asked to remember and repeat 20 activities that they did in the with-object task experiments as gesture motions, which are used for queries. The second experiment is called a without-object task. Its activities are slightly different from the with-object task activities and required completing each activity during specified times. For example, the activity is to “pour hot water” and “stir a cup of coffee” rather than to “make coffee.” The subjects then repeated the 20 activities; at this time, there was no object, and they were in a new environment.

B. METRICS

To evaluate the key-frame extraction performances of different algorithms, we introduced two metrics: accuracy and entropy.

We use accuracy (A) to evaluate the effectiveness of the proposed methods for key-frame extraction from FPV videos with multi-sensor integration, which can be described as follows:

$$A = \frac{N_{\text{Correct}}}{N_{\text{Whole}}}, \quad (16)$$

where N_{Correct} denotes the number of selected key frames that are correctly selected with respect to the label and N_{Whole} is the total number of key frames selected by the methods. Note that the labels correspond to different actions in the video; there are start frames and end frames in each label. If the selected key frame is between the start frame and end frame of the label, we consider the key frame to be correctly chosen.

However, the metric of A cannot integrally measure the quality of a key-frame extraction. As shown in Fig. 6, cases (a) and (b) have the same accuracy value of 5/6. The results of the key-frame extraction are different. In the case of (a), the selected frames are all focused on event 4. However, in the case of (b), the selected key frames are dispersed (events 1, 3, 4, 5, and 6). Generally, the result of case (b) is better than that of case (a).

To evaluate the information content of different actions in the video, we introduced entropy as a metric for the experimental results, which can be described as follows:

$$S = - \sum_{i=1} p_i \log_2 p_i. \quad (17)$$

Here, p_i is the probability of each event extracted by the proposed algorithm. This metric will be maximum if all

²<http://vision.csd.uwo.ca/code/>

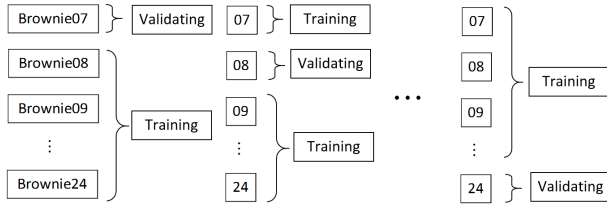


FIGURE 7. The framework of cross-validation.

Algorithm 3 Cross-Validation-Based Parameter Settings

Require: Video sequences $V_1, V_2, V_3, \dots, V_n$

- 1: **for** $i = 1$ to n **do**
- 2: Choose V_i as validation data and the others as training data.
- 3: **for** $\alpha = \alpha_1$ to α_M **do**
- 4: Apply SMRS to the training data.
- 5: Calculate the entropy over the training data.
- 6: Average the entropy.
- 7: **end for**
- 8: Draw the entropy curves versus the different values of α .
- 9: Choose the optimal value of α .
- 10: Apply SMRS to the validating data with the optimal α .
- 11: **end for**

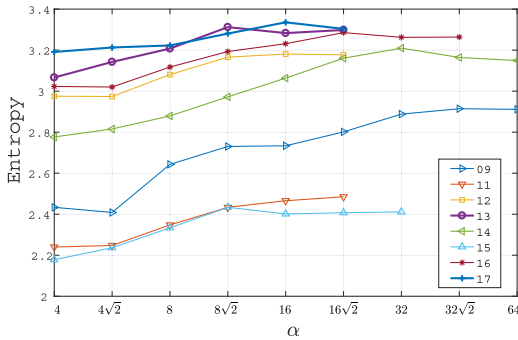


FIGURE 8. The average entropy values of the training dataset for various α (daily activities).

extracted events are equally likely. Thus, a higher entropy value means a better key-frame extraction result. In Fig. 6, the entropy of (a) is 0, and the entropy of case (b) is 2.3219. Thus, the entropy of (b) is higher than that of (a), which means that the key-frame selection result of (b) is better.

C. CROSS-VALIDATION-BASED PARAMETER SETTINGS

To obtain the appropriate parameter α in SMFE, we use cross-validation. We used all videos from the brownie dataset to determine the optimized parameter. At first, we took one brownie dataset video as validation data and the other videos as training data, which is illustrated in Fig. 7. Then, we changed α to different values and calculated the entropy of the training data and averaged the entropy results. From the curves of the average entropy versus the value of α , the optimal choice of α can be determined, which yields the highest entropy value. We describe the steps in Algorithm 3.

TABLE 1. The entropies and accuracies (%) for various types of information by SMFE (CMU-MMAC).

Dataset	Video	Motion	Multi
B07	2.99/51.85	3.26/56.67	3.20/52.94
B08	2.81/68.97	3.44/60.61	3.65/77.14
B09	3.37/ 72.41	2.91/60.71	3.44/70.59
B12	3.15/74.19	3.57/71.88	3.93/79.41
B13	3.87/78.79	3.85/56.67	3.54/56.76
B14	2.72/70.00	3.35/66.67	3.27/ 72.22
B16	3.15/68.75	3.92/75.00	3.10/ 84.21
B17	3.52/77.78	3.47/70.59	3.74/81.08
B18	3.83/ 65.71	3.89/57.58	3.65/58.33
B19	3.50/82.14	3.60/84.38	3.73/94.44
B20	3.27/73.33	3.61/72.73	3.78/77.14
B22	3.17/74.19	3.85/71.88	3.84/ 78.38
B24	2.67/72.00	3.04/73.33	3.19/79.41
Mean	3.23/71.55	3.52/67.59	3.54/74.00

TABLE 2. The entropies and accuracies (%) for various types of information by GMFE (CMU-MMAC).

Sess.	Video	Motion	Multi
B07	2.62/52.17	2.17/ 78.95	2.93/61.90
B08	3.54/60	3.33/ 64	3.61/61.54
B09	2.42/77.27	2.81/66.67	1.78/ 91.3
B12	3.16/70	2.95/75.41	2.61/ 79.03
B13	3.73/59.49	3.02/72.09	3.8/73.26
B14	2.55/63.01	2.53/66.67	2.4/ 69.33
B16	2.84/78.26	3.34/76.74	2.49/ 73.81
B17	3.69/72.22	3.16/62.5	2.53/ 80.28
B18	2.84/82.35	2.89/78.95	1.78/ 82.86
B19	2.13/70.83	3.01/88	2.13/ 92.59
B20	2.73/69.23	2.79/85.71	1.28/78.57
B22	3.01/45.45	2.95/55.17	2.6/ 57.58
B24	2.97/81.82	2.35/85.71	2.29/ 94.44
Mean	2.94/67.85	2.87/73.58	2.48/ 76.65

VI. EXPERIMENTAL RESULTS

We applied our proposed algorithms separately to the CMU-MMAC dataset and the daily activities dataset. The experimental results are presented in this section.

A. SPARSE MODEL

First, we conducted experiments using the sparse-model-based key-frame extraction algorithm. We performed experiments on the CMU-MMAC dataset with multiple information: video and motion information. To investigate the effects of different values of regularization parameter α on the quality of selected representatives, we considered the brownie dataset as political debate videos. We ran our proposed algorithm with $\alpha = 8, 8\sqrt{2}, 16, 16\sqrt{2}, 32, 32\sqrt{2}, 64, 64\sqrt{2}$ to investigate the optimal α with respect to different brownie dataset videos from 07, 08, ..., to 24. Fig. 9 displays the cross-validated entropy with various values of α . Then, we select the optimal α with the highest entropy value.

TABLE 3. Entropies and accuracies (%) for various information by SMFE (daily activities).

No.	α	Sess.	Video	Motion	Multi	No.	α	Sess.	Video	Motion	Multi
09	$32\sqrt{2}$	1	3.17/25.71	2/11.76	3/14.55	14	32	1	2.59/20.69	1/6.06	3.32/17.86
09	$32\sqrt{2}$	2	2/12.9	2/12.5	3/15.38	14	32	2	2/13.33	1.59/9.09	3.17/16.67
09	$32\sqrt{2}$	3	3/27.59	2/12.12	3.32/17.86	14	32	3	3/24.24	2/12.9	3.17/15
09	$32\sqrt{2}$	4	2.81/21.88	0/2.78	2.81/11.29	14	32	4	2/11.76	1.59/10.34	3.17/15.52
09	$32\sqrt{2}$	5	3.32/29.41	2.32/14.29	3.17/15	14	32	5	2/12.12	1.59/8.33	2.81/11.67
09	$32\sqrt{2}$	6	3.17/27.27	2.59/18.75	3.32/18.18	14	32	6	2.81/23.33	2.32/15.15	3.16/15
09	$32\sqrt{2}$	7	2/14.29	0/3.33	2.81/13.21	14	32	7	2.81/22.58	1/5.71	3.46/19.64
09	$32\sqrt{2}$	8	1/7.14	1/5.71	2.32/9.09	14	32	8	3.32/31.25	2.59/19.35	3.32/17.86
09	$32\sqrt{2}$	9	3.17/26.47	2.32/14.29	2.81/12.07	14	32	9	1.59/10	1.59/10.71	2.81/12.73
09	$32\sqrt{2}$	10	2.81/20.59	1.59/9.09	2.59/10.34	14	32	10	2.59/21.43	3/26.67	3.7/23.64
11	$16\sqrt{2}$	1	2.59/20.69	2.32/16.13	2/7.69	15	$8\sqrt{2}$	1	2/15.38	1.59/10.71	2.81/13.46
11	$16\sqrt{2}$	2	2.32/18.52	0/3.23	2/7.41	15	$8\sqrt{2}$	2	2.32/17.24	1.59/8.82	2.32/10
11	$16\sqrt{2}$	3	2.81/25	0/3.45	2.59/10.53	15	$8\sqrt{2}$	3	0/3.45	1/7.14	1.59/6.38
11	$16\sqrt{2}$	4	2.32/16.13	1/6.25	2.32/8.93	15	$8\sqrt{2}$	4	1.59/11.54	1/6.67	2.59/11.11
11	$16\sqrt{2}$	5	1/6.9	2/11.76	2.32/9.09	15	$8\sqrt{2}$	5	3.17/32.14	1.59/10.71	3/18.6
11	$16\sqrt{2}$	6	1/6.9	1/6.9	2.32/8.93	15	$8\sqrt{2}$	6	2.59/22.22	1/6.45	2.81/13.46
11	$16\sqrt{2}$	7	2.32/18.52	2/14.29	2.32/10.20	15	$8\sqrt{2}$	7	1.59/11.54	2/12.9	2.32/9.62
11	$16\sqrt{2}$	8	2.59/20	2.32/16.67	2.81/12.96	15	$8\sqrt{2}$	8	2.32/17.24	1.59/9.38	2.32/10.2
11	$16\sqrt{2}$	9	3/26.67	2/12.9	3.17/16.67	15	$8\sqrt{2}$	9	2/14.29	0/0	2/7.84
11	$16\sqrt{2}$	10	2.32/17.24	1.59/9.09	3/14.55	15	$8\sqrt{2}$	10	2/15.38	1.59/9.68	2.59/11.76
12	16	1	3.17/32.14	3.32/34.48	3.46/23.91	16	$16\sqrt{2}$	1	2.32/18.52	2.32/16.13	3.59/21.82
12	16	2	2.81/25.93	3/26.67	3.32/20	16	$16\sqrt{2}$	2	2.43/21.43	2.81/19.44	3.46/19.3
12	16	3	2.81/25	2.32/15.63	3.32/19.23	16	$16\sqrt{2}$	3	2.81/21.21	2/11.76	3.17/16.36
12	16	4	3.59/41.38	2.81/26.92	3.46/22.45	16	$16\sqrt{2}$	4	2/13.33	1.59/9.68	2.59/11.54
12	16	5	3/32	2.81/22.58	3.32/21.28	16	$16\sqrt{2}$	5	2.81/22.58	2.81/22.58	3.17/16.67
12	16	6	2.59/20	2.81/22.58	3/15.69	16	$16\sqrt{2}$	6	2.32/15.63	2.59/19.35	3.46/19.64
12	16	7	3.17/33.33	2.81/20.59	3.17/16.67	16	$16\sqrt{2}$	7	2.59/20.69	2.32/13.89	3.32/16.39
12	16	8	1.59/10.34	2.59/20.69	3/14.81	16	$16\sqrt{2}$	8	2.59/19.35	2/14.29	3.32/19.61
12	16	9	2.59/20	2.59/20.69	3.17/16.67	16	$16\sqrt{2}$	9	3.32/34.48	2/14.29	3.32/17.54
12	16	10	1.59/8.57	1/6.25	2.59/11.54	16	$16\sqrt{2}$	10	3.17/30	2/13.79	3.46/19.3
13	$8\sqrt{2}$	1	2.81/23.33	2/14.81	3.46/22.45	17	16	1	2.59/20.69	2.32/16.67	2.81/14
13	$8\sqrt{2}$	2	3/30.77	2.32/16.67	3.32/20.83	17	16	2	2.59/19.35	2.81/25	3.17/19.15
13	$8\sqrt{2}$	3	3/27.59	2.59/18.75	3.17/17.65	17	16	3	2.81/23.33	2.32/18.52	3.46/21.57
13	$8\sqrt{2}$	4	2.81/29.17	2.32/15.63	3.46/19.64	17	16	4	3.17/30	2.59/17.14	3.59/24
13	$8\sqrt{2}$	5	2.81/25.93	2.59/20	3.7/24.07	17	16	5	2.81/25	3/27.59	3.32/18.87
13	$8\sqrt{2}$	6	2.59/22.22	2.32/18.52	2.59/12.25	17	16	6	2/14.81	2.59/21.43	2.81/13.73
13	$8\sqrt{2}$	7	2/11.43	2.81/21.88	3.32/16.67	17	16	7	2.59/20	2.81/21.21	3.46/21.57
13	$8\sqrt{2}$	8	3/28.57	2.32/17.24	3.46/20.75	17	16	8	2/13.79	3.32/33.33	3.59/23.08
13	$8\sqrt{2}$	9	3/27.59	3.17/27.27	3.32/18.18	17	16	9	3.59/44.44	3.46/40.74	3.7/26.53
13	$8\sqrt{2}$	10	2.32/19.23	1.59/11.11	3.32/20.41	17	16	10	3.59/44.44	2.59/22.22	3.46/22

To evaluate the performance of our proposed multi-sensor integration, with the optimal $\alpha = 64$, we compared the entropies and accuracies using multi-sensor information and pure video and pure motion information. Table 1 presents the evaluation results, from which we can find that the performance using multi-sensor information is better than the ones using pure information in most cases.

Then, we also performed experiments on the daily activities dataset with multiple types of information: video and motion information. To obtain the optimal regularization parameter, α , in terms of the quality of selected key frames, we ran the proposed algorithm from $\alpha = 4$ to 64 with

a multiplicative step of $\sqrt{2}$ to investigate the optimal α with respect to different objects from 09, 11, ..., to 17. Each object has 10 repeated sessions. We averaged the results of each object and plot the results in Fig. 8, from which we can choose the optimal α with the highest entropy value.

With the optimal α , we compared the entropies and accuracies using multiple types of information and pure video and motion information. Table 3 displays the evaluation results, from which we can observe that multiple types of information achieve better results. Thus, our proposed multi-sensor integration achieved better performance.

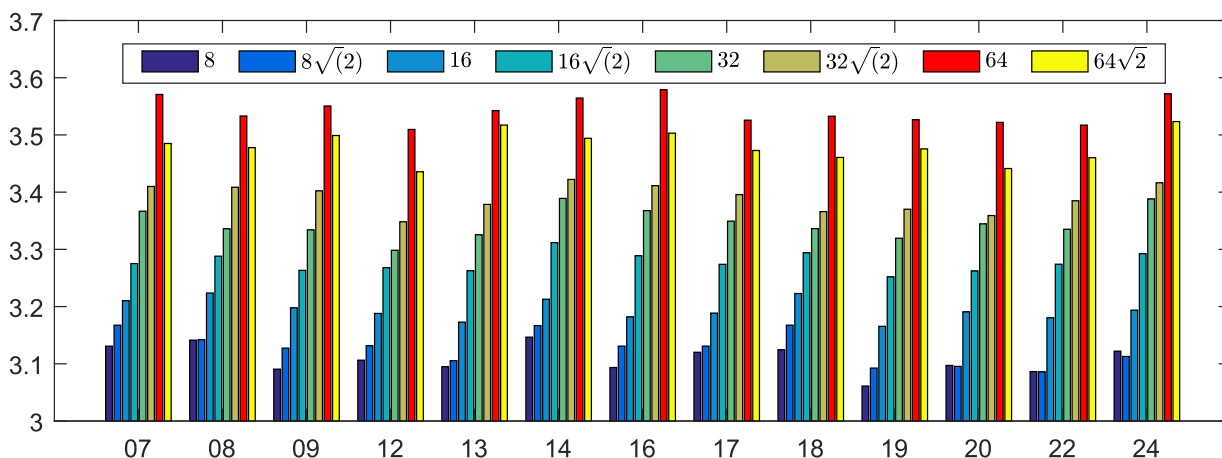


FIGURE 9. The average entropy values of the training dataset for various α (CMU-MMAC).

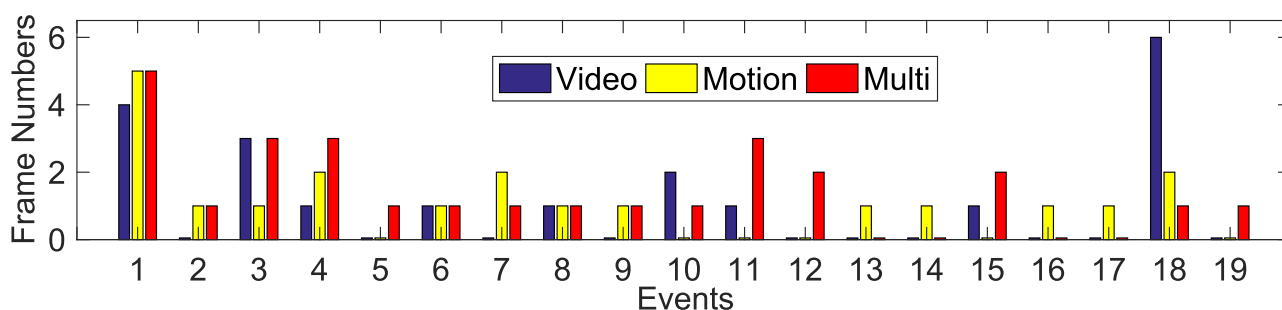


FIGURE 10. The number of each event by SMFE from brownie 08.

B. GRAPH MODEL

We conducted experiments using the graph-model-based key-frame extraction algorithm. We also applied our key-frame extraction algorithm based on the graph model to the CMU-MMAC and daily activities datasets. The parameter settings refer to those in [50]. Similar to the sparse model, we first presented results using the CMU-MMAC dataset, and we performed experiments with multiple information: video and motion information. The parameters were set to $\lambda = 1, \theta_1 = 20, \alpha = 5, \gamma = 1, \beta = 1$. The remaining parameter θ_1 controlled the number of selected key frames. Table 2 shows the results of the entropies and accuracies from various pure information and multiple information. From the experimental results, we inferred that multiple information performs better than only video or motion information.

Now, we will describe the experimental results achieved with the daily activities dataset. We took object 09 as a representative case. The parameters were set to $\lambda = 1, \theta_1 = 20, \alpha = 5, \gamma = 1, \beta = 1$. We adjusted θ_1 to control the number of selected key frames. As shown in Table 4, the experiments with multiple sensors achieved better results.

C. COMPARISON BETWEEN THE TWO MODELS

To compare the performances of SMFE and GMFE, we present the results of computational time consumption. The algorithms are run on a computer with an Intel Core i7 CPU under the Microsoft Windows 10 operating system. GMFE (averaged 700 s) costs us much less time than

TABLE 4. The entropies and accuracies (%) for various types of information by GMFE (daily activities).

Sess.	Video	Motion	Multi
1	3.71 /62.07	3.61/52.14	3.69/ 68.97
2	3.62/57.69	3.95 /38.3	3.68/ 60.33
3	3.21/69.01	3.11/40.28	3.44 / 70.59
4	3.32/50	3.22/47.5	3.37 / 56.96
5	3.27/57.3	3.69 /42.22	3.24/ 62.22
6	3.39/52.94	3.69 /50.59	3.49/ 61.63
7	3.34/51.65	3.27/47.19	3.61 / 73.63
8	3.33 /52.17	3.25/36.96	3.25/ 61.29
9	2.95/68.12	3.13 /49.3	2.71/ 70.42
10	3.3 /70.77	2.36/18.75	3.18/ 73.44

SMFE (average 2550 s). If we take the computational time consumption as a principal consideration, then GMFE will be a better choice than SMFE.

Then, we calculated the number of key frames selected by our methods for each event in the videos. Let us take brownie 08 as an example. Fig. 10 and Fig. 11 show the results, from which we can find that the key frames using proposed algorithms that use multi-information represent the events better than those using only pure information.

From the above discussion, the SMFE algorithm has fewer parameters (only one parameter) than the GMFE algorithm. Thus, SMFE is easier to adjust and more robust with respect to different videos. However, GMFE has less

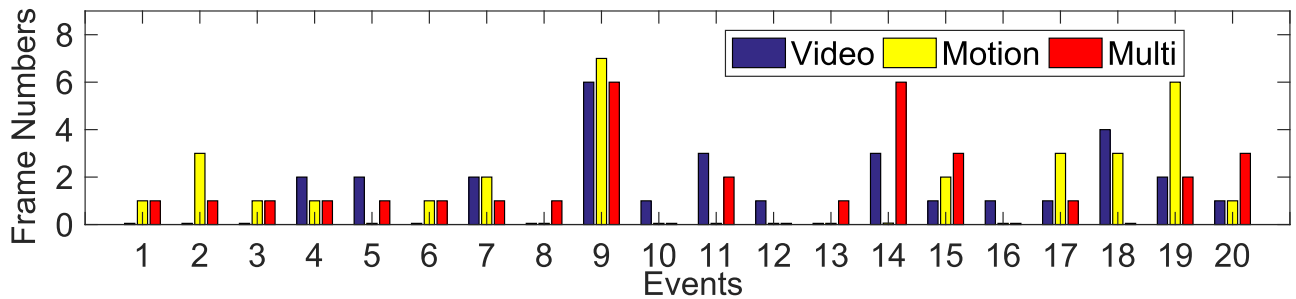


FIGURE 11. The number of each event by GMFE from brownie 08.

computational time consumption. Thus, GMFE is more efficient in high-dimensional situations.

VII. CONCLUSION

We proposed novel frameworks for key-frame extraction from FPV videos based on sparse modelling and graph modelling by multi-sensor integration. The deep features from a pre-trained DNN rather than raw video frames are used for key-frame extraction. The index of the key frame was then estimated by the proposed algorithms, which are proven to be more informative and elegant when extracting the key frames from FPV videos. The experimental results indicate that the proposed approaches can achieve a modest enhancement over pure video data. The accuracy and entropy results demonstrate the effectiveness of the proposed algorithms. Moving forward, we will develop our approach by incorporating other non-video information, including text, audio, electromyograms, and heart rate signals.

REFERENCES

- [1] M. Cornacchia, K. Ozcan, Y. Zheng, and S. Velipasalar, "A survey on activity detection and classification using wearable sensors," *IEEE Sensors J.*, vol. 17, no. 2, pp. 386–403, Jan. 2017.
- [2] A. Garcia del Molino, C. Tan, J.-H. Lim, and A.-H. Tan, "Summarization of egocentric videos: A comprehensive survey," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 1, pp. 65–76, Feb. 2017.
- [3] X. Liu, M. L. Song, L. M. Zhang, and S. L. Wang, "Joint shot boundary detection and key frame extraction," in *Int. Conf. Pattern Recognit. (ICPR)*, Tsukuba, Japan, 2012, pp. 2565–2568.
- [4] N. Ejaz, T. B. Tariq, and S. W. Baik, "Adaptive key frame extraction for video summarization using an aggregation mechanism," *J. Vis. Commun. Image Represent.*, vol. 23, no. 7, pp. 1031–1040, Oct. 2012.
- [5] Q. Xu, Y. Liu, X. Li, Z. Yang, J. Wang, M. Sbert, and R. Scopigno, "Browsing and exploration of video sequences: A new scheme for key frame extraction and 3D visualization using entropy based jensen divergence," *Inf. Sci.*, vol. 278, pp. 736–756, Sep. 2014.
- [6] E. Elhamifar, G. Sapiro, and R. Vidal, "See all by looking at a few: Sparse modeling for finding representative objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1600–1607.
- [7] X. Zhu, A. K. Elmagarmid, X. Xue, L. Wu, and A. C. Catlin, "InsightVideo: Toward hierarchical video content organization for efficient browsing, summarization and retrieval," *IEEE Trans. Multimedia*, vol. 7, no. 4, pp. 648–666, Aug. 2005.
- [8] C. Choudary and T. Liu, "Summarization of visual content in instructional videos," *IEEE Trans. Multimedia*, vol. 9, no. 7, pp. 1443–1455, Nov. 2007.
- [9] B.-W. Chen, J.-C. Wang, and J.-F. Wang, "A novel video summarization based on mining the story-structure and semantic relations among concept entities," *IEEE Trans. Multimedia*, vol. 11, no. 2, pp. 295–312, Feb. 2009.
- [10] Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z.-H. Zhou, "Multi-view video summarization," *IEEE Trans. Multimedia*, vol. 12, no. 7, pp. 717–729, Nov. 2010.
- [11] M. Wang, R. Hong, G. Li, Z.-J. Zha, S. Yan, and T.-S. Chua, "Event driven Web video summarization by tag localization and key-shot identification," *IEEE Trans. Multimedia*, vol. 14, no. 4, pp. 975–985, Aug. 2012.
- [12] F. Chen, C. De Vleeschouwer, and A. Cavallaro, "Resource allocation for personalized video summarization," *IEEE Trans. Multimedia*, vol. 16, no. 2, pp. 455–469, Feb. 2014.
- [13] Y. Zhang and R. Zimmermann, "Efficient summarization from multiple georeferenced user-generated videos," *IEEE Trans. Multimedia*, vol. 18, no. 3, pp. 418–431, Mar. 2016.
- [14] Y. Cong, J. Yuan, and J. Liu, "Sparse reconstruction cost for abnormal event detection," in *Proc. CVPR*, Colorado Springs, Jun. 2011, pp. 3449–3456.
- [15] L. Pan, X. Wu, and X. Shu, "Key frame extraction based on sub-shot segmentation and entropy computing," in *Proc. Chin. Conf. Pattern Recognit.*, Nanjing, China, Nov. 2009, pp. 1–5.
- [16] A. Kanemura, J. Yuan, and Y. Kawahara, "Finding structured dictionary representation by network-flow optimization," in *Proc. Workshop Data Discret. Segment. Knowl. Discov. (DDS)*, 2013, pp. 1–5.
- [17] H. Wang, Y. Kawahara, C. Weng, and J. Yuan, "Representative selection with structured sparsity," *Pattern Recognit.*, vol. 63, pp. 268–278, Mar. 2017.
- [18] P. Varini, G. Serra, and R. Cucchiara, "Personalized egocentric video summarization of cultural tour on user preferences input," *IEEE Trans. Multimedia*, vol. 19, no. 12, pp. 2832–2845, Dec. 2017.
- [19] E. A. Bernal, X. Yang, Q. Li, J. Kumar, S. Madhvanath, P. Ramesh, and R. Bala, "Deep temporal multimodal fusion for medical procedure monitoring using wearable sensors," *IEEE Trans. Multimedia*, vol. 20, no. 1, pp. 107–118, Jan. 2018.
- [20] J. Sang, J. Liu, and C. Xu, "Exploiting user information for image tag refinement," in *Proc. 19th ACM Int. Conf. Multimedia*, Scottsdale, AZ, USA, 2011, pp. 1129–1132.
- [21] C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning," 2013, *arXiv:1304.5634*. [Online]. Available: <http://arxiv.org/abs/1304.5634>
- [22] A. Lazaridou, E. Bruni, and M. Baroni, "Is this a Wampimuk? cross-modal mapping between distributional semantics and the visual world," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics*, Baltimore, MD, USA, 2014, pp. 1403–1414.
- [23] Y. Gong, Q. Ke, M. Isard, and S. Lazebnik, "A multi-view embedding space for modeling Internet images, tags, and their semantics," *Int. J. Comput. Vis.*, vol. 106, no. 2, pp. 210–233, Jan. 2014.
- [24] E. Bruni, N. K. Tran, and M. Baroni, "Multimodal distributional semantics," *J. Artif. Intell. Res.*, vol. 49, pp. 1–47, Jan. 2014.
- [25] D. Wang, P. Cui, M. Ou, and W. Zhu, "Learning compact hash codes for multimodal representations using orthogonal deep structure," *IEEE Trans. Multimedia*, vol. 17, no. 9, pp. 1404–1416, Sep. 2015.
- [26] X. Huang, A. Dhall, R. Goecke, M. Pietikainen, and G. Zhao, "Multimodal framework for analyzing the affect of a group of people," *IEEE Trans. Multimedia*, vol. 20, no. 10, pp. 2706–2721, Oct. 2018.
- [27] F. Wang and C.-W. Ngo, "Summarizing rushes videos by motion, object, and event understanding," *IEEE Trans. Multimedia*, vol. 14, no. 1, pp. 76–87, Feb. 2012.
- [28] F. R. Bach and M. I. Jordan, "A probabilistic interpretation of canonical correlation analysis," Dept. Statist., Univ. California, Berkeley, CA, USA, Tech. Rep. 688, 2005.
- [29] C. Sutton, K. Rohanimanesh, and A. McCallum, "Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data," in *Proc. 21st Int. Conf. Mach. Learn.*, Williamstown, MA, USA, 2004, pp. 693–723.

- [30] Y. Li, A. Kanemura, H. Asoh, T. Miyanishi, and M. Kawanabe, "Extracting key frames from first-person videos in the common space of multiple sensors," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Beijing, China, Sep. 2017, pp. 3993–3997.
- [31] Y. Li, A. Kanemura, H. Asoh, T. Miyanishi, and M. Kawanabe, "Key frame extraction from first-person video with multi-sensor integration," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Hong Kong, Jul. 2017, pp. 1303–1308.
- [32] A. G. Money and H. Agius, "Video summarisation: A conceptual framework and survey of the state of the art," *J. Vis. Commun. Image Represent.*, vol. 19, no. 2, pp. 121–143, Feb. 2008.
- [33] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. New York, NY, USA: Springer-Verlag, 2010.
- [34] Y. Li, S. Ding, and Z. Li, "A dictionary-learning algorithm for the analysis sparse model with a determinant-type of sparsity measure," in *Proc. 19th Int. Conf. Digit. Signal Process.*, Hong Kong, Aug. 2014, pp. 152–156.
- [35] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
- [36] R. Rubinstein, T. Peleg, and M. Elad, "Analysis K-SVD: A dictionary-learning algorithm for the analysis sparse model," *IEEE Trans. Signal Process.*, vol. 61, no. 3, pp. 661–677, Feb. 2013.
- [37] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via l_1 minimization," *Proc. Nat. Acad. Sci. USA*, vol. 100, no. 5, pp. 2197–2202, Mar. 2003.
- [38] T. Chen, Y. and Pock, and H. Bischof, "Learning ℓ_1 -based analysis and synthesis sparsity priors using bi-level optimization," in *Proc. NIPS*, Las Vegas, NV, USA, 2012, pp. 1–5.
- [39] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer-Verlag, 2006.
- [40] O. Veksler, "Image segmentation by nested cuts," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, New York, NY, USA, Jun. 2000, pp. 339–344.
- [41] H. Ishikawa and D. Geiger, "Segmentation by grouping junctions," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, New York, NY, USA, May 1998, pp. 125–131.
- [42] D. M. Greig, B. T. Porteous, and A. H. Seheult, "Exact maximum a posteriori estimation for binary images," *J. Roy. Stat. Soc., Ser. B Methodol.*, vol. 51, no. 2, pp. 271–279, Jan. 1989.
- [43] Y. Boykov, O. Veksler, and R. Zabih, "Markov random fields with efficient approximations," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, New York, NY, USA, Oct. 1998, pp. 648–655.
- [44] H. Ishikawa and D. Geiger, "Occlusions, discontinuities, and epipolar lines in stereo," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New York, NY, USA, Jun. 1998, pp. 232–248.
- [45] D. Snow, P. Viola, and R. Zabih, "Exact voxel occupancy with graph cuts," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, New York, NY, USA, Jun. 2000, pp. 345–352.
- [46] Y. Boykov and D. Huttenlocher, "A new Bayesian framework for object recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New York, NY, USA, May 1999, pp. 517–523.
- [47] V. Kwatra, A. Schodl, I. Essa, and A. Bobick, "Graphcut textures: Image and video synthesis using graph cuts," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 277–286, 2003.
- [48] Y. S. Rawat, M. Song, and M. S. Kankanhalli, "A spring-electric graph model for socialized group photography," *IEEE Trans. Multimedia*, vol. 20, no. 3, pp. 754–766, Mar. 2018.
- [49] C.-W. Ngo, Y.-F. Ma, and H.-J. Zhang, "Video summarization and scene detection by graph modeling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 2, pp. 296–305, Feb. 2005.
- [50] A. G. del Molino, X. Boix, J.-H. Lim, and A.-H. Tan, "Active video summarization: Customized summaries via on-line interaction with the user," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, San Francisco, CA, USA, vol. 2017, pp. 4046–4052.
- [51] D. DeMenthon, V. Kobla, and D. Doermann, "Video summarization by curve simplification," in *Proc. 6th ACM Int. Conf. Multimedia*, Bristol, U.K., 1998, pp. 211–218.
- [52] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Banff, Canada, 2014, pp. 1–14.
- [53] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural Comput.*, vol. 16, no. 12, pp. 2639–2664, Dec. 2004.
- [54] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 24, no. 2, pp. 227–234, Apr. 1995.
- [55] M. Huang, W. Yang, J. Jiang, Y. Wu, Y. Zhang, W. Chen, and Q. Feng, "Brain extraction based on locally linear representation-based classification," *NeuroImage*, vol. 92, pp. 322–339, May 2014.
- [56] M. S. Lewicki and T. J. Sejnowski, "Learning overcomplete representations," *Neural Comput.*, vol. 12, no. 2, pp. 337–365, Feb. 2000.
- [57] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T.-W. Lee, and T. J. Sejnowski, "Dictionary learning algorithms for sparse representation," *Neural Comput.*, vol. 15, no. 2, pp. 349–396, Feb. 2003.
- [58] Z. Li, S. Ding, and Y. Li, "A fast algorithm for learning overcomplete dictionary for sparse representation based on proximal operators," *Neural Comput.*, vol. 27, no. 1, pp. 1–32, 2015.
- [59] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Oct. 2001.
- [60] V. Kolmogorov and M. J. Wainwright, "On optimality properties of tree-reweighted message-passing," in *Proc. Conf. Uncertain. Artif. Intell. (UAI)*, Edinburgh, U.K., 2005, pp. 316–323.
- [61] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.
- [62] L. Gorelick, Y. Boykov, O. Veksler, I. B. Ayed, and A. Delong, "Submodularization for binary pairwise energies," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, America, Jun. 2014, pp. 1154–1161.
- [63] F. de la Torre, J. K. Hodgins, J. Montano, S. Valcarcel, A. Bargteil, X. Martin, J. Macey, A. Collado, and P. Beltran, "Guide to the Carnegie Mellon University Multimodal Activity (CMU-MMAC) Database," *Robot. Inst., Carnegie Mellon Univ., Pittsburgh, PA, Tech. Rep. CMU-RI-TR-08-22*, Apr. 2008.
- [64] T. Miyanishi, J. Hirayama, Q. Kong, T. Maekawa, H. Moriya, and T. Suyama, "Egocentric video search via physical interactions," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, Phoenix, AZ, USA, 2016, pp. 330–336.

•••