This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology

CAMHID: Camera Motion Histogram Descriptor and Its Application to Cinematographic Shot Classification

Muhammad Abul Hasan, Min Xu, *Member, IEEE,* Xiangjian He, *Senior Member, IEEE,* and Changsheng Xu, *Fellow, IEEE*

***Abstract*—In this paper, we propose a non-parametric camera motion descriptor called CAMHID for video shot classification. In the proposed method, a motion vector field (MVF) is constructed for each consecutive video frames by computing the motion vector of each macroblock (MB). Then, the MVFs are divided into a number of local region of equal size. Next, the inconsistent/noisy motion vectors of each local region are eliminated by a motion consistency analysis. The remaining motion vectors of each local region from a number of consecutive frames are further collected for a compact representation. Initially, a matrix is formed using the motion vectors. Then, the matrix is decomposed using singular value decomposition (SVD) technique to represent the dominant motion. Finally, the angle of the most variance retaining principal component is computed and quantized to represent the motion of a local region by using a histogram. In order to represent the global camera motion, the local histograms are combined. The effectiveness of the proposed motion descriptor for video shot classification is tested by using support vector machine (SVM). Firstly, the proposed camera motion descriptors for video shots classification are computed on a video dataset consisting of regular camera motion patterns (e.g., pan, zoom, tilt, static). Then, we apply the camera motion descriptors with an extended set of features to the classification of cinematographic shots. The experimental results show that the proposed shot level camera motion descriptor has strong discriminative capability to classify different camera motion patterns of different videos effectively. We also show that our approach outperforms state-of-the-art methods.**

***Index Terms*—video shot classification, motion analysis, singular value decomposition.**

I. Introduction

VIDEO is an important medium and it describes the visual content of a scene with the help of time domain to human eyes. Smooth visual information flow is achieved by capturing a significantly large number of sequential frames per second to comply with the human brain's cognition speed limit. As a result, video cameras have to capture highly redundant visual data. With the ever increasing production of video data, the demand of efficient indexing, retrieving and browsing of this redundant form of video data is a necessity. A video produced by a freely moving camera is a rich form of data which is heavily existed in today's multimedia contents. Apart from the visual information, interesting spatio-temporal information is also buried inside the raw video data. Using the dynamic spatio-temporal information, many methods have been proposed for video concept detection, content based video indexing and retrieval [1]–[7]. The approaches based on motion characterization have played an important role in this regard [8]. In videos captured by freely moving cameras, camera movements and object movements are two main sources of motion. By analyzing such motion information, video shots are classified to aid the tasks for content based video indexing and retrieval [9].

A robust shot classification technique can be useful in many applications such as shot indexing and retrieval [8], video summarization [10] and video data structuring [11]. In this regard, camera motion characterization based video shot classification techniques play an important role. In [12], a shot characterization technique is proposed to represent video shots by using histograms. In [13], [14], MPEG domain motion vectors (MVs) encoded in P- and B-frames were analyzed to characterize camera motion. A temporal slice based analysis was used to characterize camera motion by Ngo *et al.* [15]. Optical flow based methods were used to extract motion information and to model camera motion and object motion [16]–[19].

In this paper, we propose a novel camera motion characterization and description technique by identifying camera motion patterns from an inherent motion structure buried in raw video data. We deal with motion information in two main stages, motion characterization and motion description. Our intention is to exploit the redundant information of video data to characterize the camera motion patterns of video shots. Firstly, motion vectors of consecutive frames are extracted and inconsistent motions are suppressed by applying a statistical temporal motion analysis. Then, global motion patterns of each video shot is represented by using a number of local motion descriptors. The local motion descriptors are used to describe local camera motion patterns. Then, the extracted features are used in a statistical learning framework to recognize the qualitative camera motion patterns which have been categorized into directing semantic classes by video directors from the

Manuscript received month xx, xxxx; revised month xx, xxxx.

M. A. Hasan, M. Xu and X. He are with the Centre for Innovation in IT Services and Applications (iNEXT), School of Computing and Communications, Faculty of Engineering and Information Technology, University of Technology, Sydney, 2007, Australia, (e-mail: muhammadabul.hasan@student.uts.edu.au, min.xu@uts.edu.au, xiangjian.he@uts.edu.au). X. He is the corresponding author.

C. Xu is with the National Lab of Pattern Recognition. Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: csxu@nlpr.ia.ac.cn).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology

video directing point of view.

At the motion characterization stage, motion vector fields (MVFs) are constructed by extracting motion vectors from consecutive frames using a block matching (BM) technique. The MVFs are segmented into a number of equal sized local region. Then, the temporal gradient of the motion vectors of each MB of each local region is computed. Then, by using an effective statistical measure on the gradient of motion, the motion vectors of interest (MVIs) are identified. MVIs of each local region are then characterized by using the principal component analysis. The most variance retaining principal component is identified to represent the camera motion compactly. To do this, we accumulate the MVIs from a small number of frames of a local region and construct a matrix. Then, the matrix is decomposed using the SVD technique. Let us assume that we have t frames in a video shot. The matrix is formed by using MVIs of the corresponding local regions of n consecutive frames where ($n \ll t$). Finally, the oriented angle of the principle component is computed and quantized with a predefined step size. The consecutive quantized angles of each local region are used to characterize the local temporal motion. At the motion description stage, quantized angles of each local region are used to create a histogram. Each of the histogram is considered as local motion descriptor. By combining all the local histograms, the camera motion histogram descriptor is formed for an input video. Figure 1 shows the flow diagram of the proposed method.

In order to identify our work as a novel research effort, we study the state-of-the-art techniques on motion analysis and characterization. Our motivation is to develop a novel motion descriptor which has strong discriminative capability to distinguish different camera motions in a wide range of video shot types. The contributions of our work are as follows.

- A novel compact camera motion characterization technique is introduced to characterize the camera motion in a video.
- A novel shot level camera motion descriptor is proposed to represent the overall motion activity of a shot by using a histogram.
- We investigate the performance of the proposed camera motion descriptor along with a set of additional features in cinematographic shot classification.

The detail of our contribution is described in Sections III and IV.

The rest of the paper is organized as follows. In Section II, the related work is reviewed. Our proposed approach for camera motion characterization and description technique is described in Section III. Section IV describes the need of additional features for cinematographic shot classification and introduces a new set of features. In Section V, we show the effectiveness of the proposed camera motion descriptor in classifying different camera motion types. This paper is concluded in Section VI.

II. Related Work

In the context of motion analysis for video shot classification, indexing and retrieval, there are a great deal of research

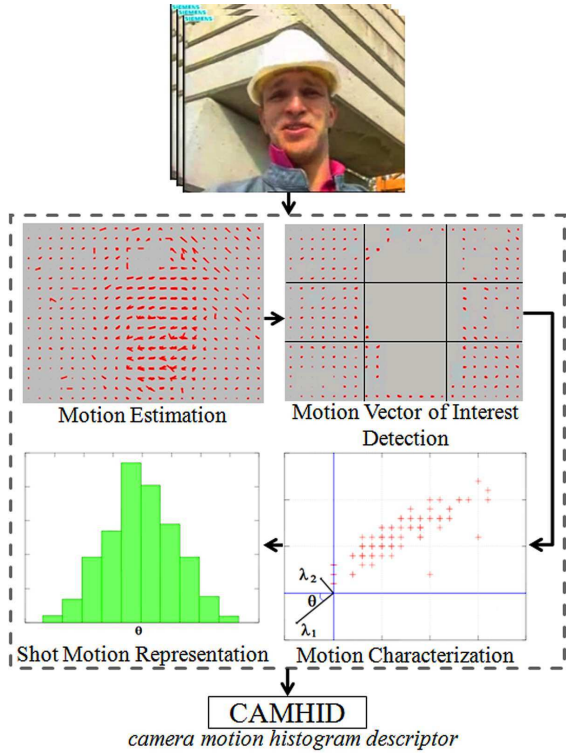


Fig. 1. Flow diagram of the proposed camera motion histogram descriptor technique.

work accomplished [9], [20]–[25]. In case of semantic analysis of video shot, camera motion patterns are often used as an important clue. While watching a video, human visual systems can perceive motions which can be described in terms of the motion quality: slow or fast motion. However, in a computer vision system, it is described by using activity descriptors. In [26], visual motion descriptors were organized into four categories: motion activity [27], camera motion [11], [28], mosaic [29] and motion trajectory [8]. Camera motion descriptors are used to represent the type of camera motion happened in a video shot. Mosaic is captured by using the parameters of the parametric motion model of a camera. Motion trajectory tells the object motion in time. Generally, the camera motion descriptors tell the generic inherent camera motion which is used to identify the directing semantics of video shots.

Many parametric methods for camera motion detection have been proposed. By using two consecutive video frames of a video, global motion models were proposed in [11], [29]–[32]. In each case, dominant motion patterns were determined using robust statistical techniques. Although global camera motion detection techniques are theoretically sound, it is considered to be less feasible to estimate correct parameters in wide variety of videos. The 2D parametric transformation suffers from a weak assumption that assumes the camera distance from the scene to be far. This assumption will lead to estimating wrong rotational and translational parameters. In [31], [32], the depth problem was handled by identifying the horizon lines. It is based on the assumption that there is a horizon line present in a typical outdoor scene video which can be identified by

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology

using a gradient analysis in gray scale images. This assumption will lead to many wrong parameter calculations where the horizon line is not obvious. None of the mentioned parametric models can be applied in classifying video shots based on directing semantic classes, as directing semantic video shots [33] consist of wide variety of shooting techniques in wide variety of shooting sets.

In contrast, nonparametric methods analyze the video data by using statistical methods to measure local or global camera motions. In [34], motion distribution of a shot was represented by using a histogram to analyze the camera motion to measure the video shot similarity. A template matching based approach was used to recognize camera motions in [22], [35]. In [9], a nonparametric spatio-temporal mode-seeking method was proposed in the motion space. The spatial distribution of the dominant motion modes was used to represent motion characteristics. Although this method is capable of learning the semantic concepts of video shots, it does not have the capability to model temporal motion patterns. Ma *et al.* [36] proposed another nonparametric generic motion pattern descriptor for video shot classification. A mapping technique based on a unit circle was applied to transform MV fields to a multidimensional motion pattern descriptor. Although it introduces a temporal information accumulating technique for statistical learning, it lacks an unified representation framework.

Template based camera motion detection was used in [22], [23]. Lan *et al.* proposed a framework for home video camera motion analysis in [23]. A template based background motion estimation (ME) technique was applied to characterize different camera motions. Lee *et al.* proposed an MPEG video stream shot classification technique [22]. A video was divided into shots and the shots were classified into six basic camera movements using templates. Template based techniques cannot be used in highly dynamic video sequences due to the high random motions captured from different objects and the motions from the camera.

In order to use motion information buried in the video data effectively, it is important to extract motion information from the frame sequences. In [24], [25], MPEG video MVs were directly used in a camera motion descriptor. Although a good classification accuracy was achieved, direct use of MVs for compressed videos could be misleading. For the purpose of best representation of a video frame, MVs in a MPEG frame were predicted either from its previous frame (i.e., P-frame) or bidirectionally predicted frame (i.e., B-frame). Bidirectionally predicted frames are much complex in nature as they use both previous and future frames for motion compensation. Thus, it can be concluded that the MVs in MPEG videos do not represent the optimal optical displacement of a macroblock (MB) with respect to time. In [24], backward predicted MVs were mapped to a forward predicted one. Moreover, I-frame's MVs were estimated by interpolation of two nearest P-frames. The overall procedure may produce misleading motion information and eventually may identify a wrong camera motion. In [25], MVs were only estimated from P-frames for characterization of camera motion. This technique of extracting motion information may suffer from lack of information and hence the estimated camera motion may not be accurate. Due to this problem, frames may suffer from the wrong representation during reconstruction and frame-rate-up conversation. Many research efforts were made to address this problem in the compressed domain [37], [38].

In conclusion, although many research effort have been accomplished to characterize camera motions of video shots, there is still a room to improve it. Efficient motion representation and effective motion description techniques can represent a video shot for semantic indexing. In our proposed method, we apply a BM based ME technique to estimate MVs from video frames and further represent the MVs compactly in the temporal direction to overcome the above-mentioned problems. The detail of our approach is shown in Sections III and IV below.

III. Proposed Camera Motion Descriptor

In this section, the proposed technique for camera motion descriptor is described in detail. We name the proposed descriptor the CAmera Motion HIstogram Descriptor (CAMHID). The CAMHID is constructed in four steps. Firstly, BM based ME technique is used to estimate the local motion of each MB. Then, by analyzing the local motion of each MB in the temporal direction, MVIs are identified. Then, in the third step, MVIs are used to produce a sequence of compact representations of the temporal motion. In the last step, the compact representations are used to obtain a normalized histogram as the local motion feature of video shots. As shown in Figure 1, the output of the above-mentioned four steps for feature extraction is a CAMHID, that integrates the motion features of the local regions. The following subsections describe the detail of CAMHID construction.

A. Block Matching Based Motion Estimation

BM based ME is a popular technique to estimate local motion. This technique has been widely used for video compression, particularly for motion compensation in the current state-of-the-art video coding standards [39]. The ME techniques find the optimal optical displacement in an MB of a frame. The optimal displacement is represented by an MV which corresponds to the coordinate displacements of the best matching block in the reference frame. For an MB in the i-th frame, MV is searched in the $(i+1)$-th frame. Let $\{f^i, f^{(i+1)}\}$ be two consecutive frames taken from a video shot. We construct a motion vector field (MVF) by extracting all MVs. In order to do that, frame f^i is subdivided into non-overlapping MBs of size $N \times N$ (see Figure 2). For each MB, the most similar block in frame $f^{(i+1)}$ is identified by searching in the area of size $(M+N)\times(M+N)$ in $f^{(i+1)}$ as shown in Figure 2, where $M = 2\times N$. For an MV denoted by $mv^i_{(x,y)}$, we need to compute the horizontal displacement $u^i_{(x,y)}$ and the vertical displacement $v^i_{(x,y)}$. Formally, we write:

$$(u^i_{(x,y)}, v^i_{(x,y)}) = \arg \min_{\substack{u\in\{-\frac{M}{2}+1,\ldots,\frac{M}{2}\} \\ v\in\{-\frac{M}{2}+1,\ldots,\frac{M}{2}\}}} e(x,y,u,v,i) \quad (1)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology

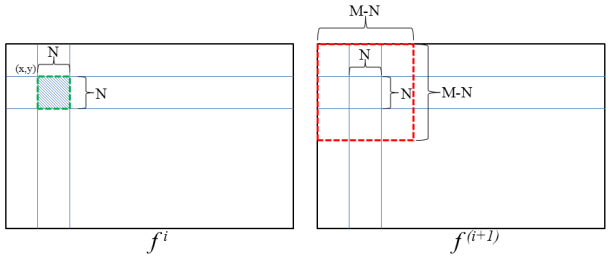


Fig. 2. BM based MV searching from two consecutive frames f^i and $f^{(i+1)}$. For an $N \times N$ MB at (x,y) in f^i, the searching area is marked in $f^{(i+1)}$. The size of the searching area is $(M+N) \times (M+N)$ centring at the searching block region.

where,

$$e(x,y,u,v,i) = \sum_{p=0}^{N-1}\sum_{q=0}^{N-1} |f^i(x+p,y+q) - f^{(i+1)}(x+u+p,y+v+q)|$$

The optimal optical displacement identified by Eq. (1) is used to compute $u^i_{(x,y)} = (x-u)$ and $v^i_{(x,y)} = (y-v)$ which represent the horizontal and vertical displacements of the $mv^i_{(x,y)}$ respectively. Likewise, we construct $MVF^i = \{mv^i_{(x,y)}\}$, for $\forall x \in \alpha, \forall y \in \beta$. Here, α and β correspond to the width and height of the video frame respectively.

B. MV of Interest Detection

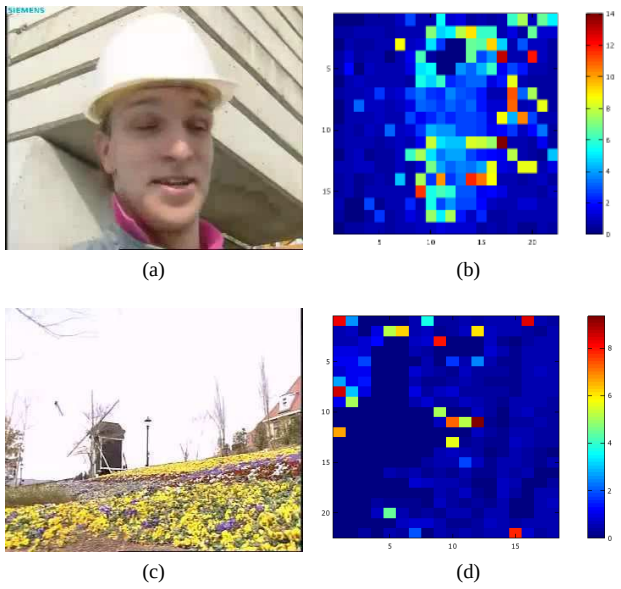


Fig. 3. Motion analysis using benchmark video sequences. (a) Foreman video sequence. (b) The variance of the gradient of the motion vectors of the first 10 MVFs of the Foreman sequence are computed and depicted. (c) Flower garden video sequence. (d) The variance of the gradient of the motion vectors of the first 10 MVFs of the Flower garden sequence are computed and depicted.

In this subsection, we propose a technique to identify the MVI by analyzing the local motions in the temporal direction. Our goal is to identify the spatial region where the motion information has a direct relationship with the camera movements. The basic camera movements are categorized as static, pan, tilt, zoom and combination of them. The camera can be operated manually by holding in a hand or by mounting on a tripod or any form of transportation. In professional video shooting (e.g., the shooting for sports, news, and film), video shots are captured with smooth, jerking free and consistent camera motions. The objects in video frames can be static or dynamic. Due to camera movement, the MVs belonging to the static object region have a direct relationship with the camera movements. However, according to our observation, motion pertaining to non-rigid bodies and focused objects (e.g., objects are being shot) are independent of camera motions. Non-rigid bodies (e.g., rippling water and waving leaves) and players/actors often produce random and jerky motions with respect to the camera frames. Figure 3 shows camera motion analysis by using benchmark video sequences. Although the first 10 frames of the Foreman sequence are identified as static, the person in the frame actually produces a random motion with respect to the camera. Figure 3(b) and 3(d) show the motion variance over the first 10 video frames in the Foreman and Flower garden video sequences. As it can be seen, MBs belonging to the non-rigid bodies produce high variances. The flower garden sequence mostly preserves camera motion information in most of the MBs. The washed out areas of the helmet (in the Foreman sequence) and a big part of the sky area (in the flower garden sequence) do not produce any motion information. Based on this observation, in this subsection, we search for the MVIs where camera motion consistency is preserved in the computed MVs. In order to do that, we go through the following steps. First of all, for each MB, we compute the gradient of MVs in the temporal direction. The gradients of horizontal displacement $u^i_{(x,y)}$ and of vertical displacement $v^i_{(x,y)}$ are computed separately for each MB of the entire shot. Formally, we write:

$$\begin{aligned} \nabla u_{(x,y)} &= \{\nabla u^1_{(x,y)}, \nabla u^2_{(x,y)}, \ldots, \nabla u^{(t-1)}_{(x,y)}\} \\ \nabla v_{(x,y)} &= \{\nabla v^1_{(x,y)}, \nabla v^2_{(x,y)}, \ldots, \nabla v^{(t-1)}_{(x,y)}\} \end{aligned} \quad (2)$$

where,

$$\begin{aligned} \nabla u^i_{(x,y)} &= u^{(i+1)}_{(x,y)} - u^i_{(x,y)} \\ \nabla v^i_{(x,y)} &= v^{(i+1)}_{(x,y)} - v^i_{(x,y)} \end{aligned}$$

Next, MVs are determined as MVI or not by employing a simple and effective statistics based traditional measure of distance on the computed $\nabla u_{(x,y)}$ and $\nabla v_{(x,y)}$. In order to identify MVIs, we check the consistency of the motion activity of MBs by using the gradient of the MVs (described in Eq. (2)). If the MB at (x,y) of frame f^i shows that the motion is significantly consistent for k $k < (t-1)$ consecutive MBs in the temporal direction, then the MV is declared as an MVI. The MVI determination process is formally written in Eq. (3).

$$mvi^i_{(x,y)} = \begin{cases} true & P(\nabla u^i_{(x,y)}, \nabla v^i_{(x,y)} | \mu, \Sigma) < \tau \\ false & otherwise \end{cases} \quad (3)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology

where $P()$ computes the multivariate normal probability, assuming that $\nabla u_{(x,y)}$ and $\nabla v_{(x,y)} \sim N(\mu, \Sigma)$,

$$\mu = \begin{bmatrix} \mu_{\nabla u^i_{(x,y)}} \\ \mu_{\nabla v^i_{(x,y)}} \end{bmatrix} \quad \Sigma = \begin{bmatrix} \sigma_{\nabla u^i_{(x,y)}} & \sigma_{\nabla u^i_{(x,y)}} \sigma_{\nabla v^i_{(x,y)}} \\ \sigma_{\nabla u^i_{(x,y)}} \sigma_{\nabla v^i_{(x,y)}} & \sigma_{\nabla v^i_{(x,y)}} \end{bmatrix},$$

$$\sigma_{\nabla u^i_{(x,y)}} = \sqrt{\frac{1}{k-1} \sum_{j=0}^{k-1} \left(\nabla u^{i+j}_{(x,y)} - \mu_{\nabla u^i_{(x,y)}} \right)^2},$$

$$\sigma_{\nabla v^i_{(x,y)}} = \sqrt{\frac{1}{k-1} \sum_{j=0}^{k-1} \left(\nabla v^{i+j}_{(x,y)} - \mu_{\nabla v^i_{(x,y)}} \right)^2},$$

$$\mu_{\nabla u^i_{(x,y)}} = \frac{1}{k} \sum_{j=0}^{k-1} \nabla u^{i+j}_{(x,y)},$$

$$\mu_{\nabla v^i_{(x,y)}} = \frac{1}{k} \sum_{j=0}^{k-1} \nabla v^{i+j}_{(x,y)},$$

and τ is a threshold for motion inconsistency tolerance which is set experimentally.

C. Motion Characterization

The MVI based motion characterization technique is described in this subsection. Our intention is to characterize a video shot in such a way that preserves the motion content in a compact manner. Figure 4 shows roughly region-wise shot motion summary of common video shot types (according to the definition of each shot type). It shows that the motion patterns of static, tilt and pan shot are similar in every frame region. However, zoom shot's motion pattern varies and is dependent on local regions. Therefore, in order to identify the characteristics of a video shot, we need to consider this fact. Accordingly, we divide the computed MVFs into nine (i.e., 3×3) local non-overlapping regions of equal size. The MVF at the local region (p,q) of frame i is denoted by $MVF^i_{(p,q)}$, where $p \in \{1,2,3\}$ and $q \in \{1,2,3\}$. For each of the local regions, $MVF_{(p,q)}$'s motion contents in the temporal direction are separately and compactly represented. Figure 5 shows the basic strategy of the motion characterization procedure. As shown in the figure, $mvi^i_{(x,y)} \in MVF^i_{(p,q)}$ of n consecutive temporal regions are accumulated for a compact representation. During the accumulation, we compute the mean distance magnitude of the motion vectors. Let us assume that we have l MVIs. Formally, we represent the mean as follows.

$$\mu_{mag} = \frac{1}{l} \sum |mv^i_{(x,y)}|, \quad \forall mv^i_{(x,y)} \in mvi^i_{(x,y)} \quad (4)$$

If the value of μ_{mag} in Eq. (4) is zero, then it is obvious that there is no camera motion present in the corresponding frames. Practically, some small unnoticeable camera motions may remain present in the static video shots. In order to handle this kind of small camera motions, the mean magnitude is checked. If the mean magnitude is smaller than a predefined threshold, then the corresponding region is characterized as a static region. Otherwise, the motion related to the local region is characterized and compactly represented. In this work, the region motion is identified as static if the value of μ_{mag} is less than 1.0.

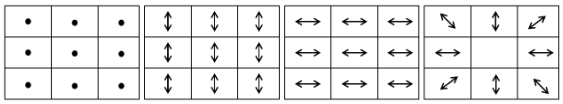


Fig. 4. Region-wise shot motion summary of static, tilt, pan and zoom shots (left to right). Each of the local regions represents the rough camera motion direction.

The compact representation is accomplished using the singular value decomposition (SVD) technique [40]. At the beginning, a matrix A is created with the accumulated MVI from n blocks of $MVF^i_{(p,q)}$. Matrix A contains l MVs and the MVs are d dimensional vectors. In our case, $d = 2$ as we have only two components in the motion vectors. Therefore, A is an $l \times 2$ matrix. Then, we apply SVD on A to decompose it as follows.

$$A = U \Lambda V^T, \quad (5)$$

where U is an $(l \times d)$ orthonormal matrix. The columns of U are the eigenvectors of AA^T. $\Lambda = diag(\lambda_1, \ldots, \lambda_d)$ is a $d \times d$ diagonal matrix containing the singular values in descending order. The singular values are the square roots of the eigenvalues of both AA^T and A^TA. The magnitude of each singular value corresponds to the importance of the corresponding principal component. V is a $(d \times d)$ orthonormal matrix and the columns of V are the eigenvectors of A^TA. We are interested in V as it encodes the coefficients used to expand A in terms of U. As the top s $(s < d)$ principle components approximate a significant amount of information of the original data [41], we represent the camera motion using the most dominant principle component of V. Therefore, the accumulated MVI from n blocks of $MVF^i_{(p,q)}$ is compactly represented by the most dominant component. The vector is identified as $pc^{(i,n)}_{(p,q)}$, where n is a constant.

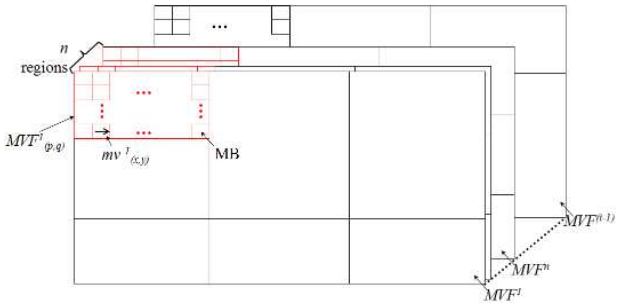


Fig. 5. Local motion characterization using motion vectors. The MVIs from n consecutive local regions are accumulated for a compact representation.

D. Shot Motion Representation

In the previous subsection, each time a set of n temporal regions are either characterized as static or further characterized according to the direction of the most dominant principal component. For each local region, we use the $pc^{(i,n)}_{(p,q)}$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology

obtained in Subsection III-C to form a histogram. First of all, the oriented angles of the principal components $pc^{(in)}_{(pq)}$ are computed. Eq. (6) is used to compute the oriented angles.

$$\theta^{(in)}_{(pq)} = \begin{cases} \arccos\left(v \cdot pc^{(in)}_{(pq)}\right) & \text{if } y_{pc} \geq 0 \\ 360^\circ - \arccos\left(v \cdot pc^{(in)}_{(pq)}\right) & \text{otherwise} \end{cases} \quad (6)$$

where y_{pc} is the y component of $pc^{(in)}_{(pq)}$ and v is an unit vector along x axis. The angle is quantized with Q levels in the range $[0^\circ, 360^\circ]$. Figure 6 shows the angle quantization strategy. The first angle level range is 345° to 15° in counter clockwise direction, and rest of the levels are equally spaced along the angle circle. Using the computed angle and the static motion information, The histogram for a region is formulated as follows.

$$H_{(pq)}(c) = \left[h^0_{(pq)}(c), h^1_{(pq)}(c), \ldots, h^Q_{(pq)}(c)\right] \quad (7)$$

where $h^i_{(pq)}(c)$ represents the count (or height) of the i-th bin. The first bin represents the static region count. The rest are corresponding to the quantized angles with indices $i \in \{1, \ldots, Q\}$. Finally, the histogram is normalized for a uniform representation using the following equation.

$$\hat{H}_{(pq)}(c) = \left[\frac{h^0_{(pq)}(c)}{C}, \frac{h^1_{(pq)}(c)}{C}, \ldots, \frac{h^Q_{(pq)}(c)}{C}\right] \quad (8)$$

where $C = \sum_{i=0}^{Q} h^i_{(pq)}(c)$. Eq. (8) is used as the feature of a region to represent the camera motion related to the region.

After all mentioned in Subsections III-A-III-D, we integrate all histogram features corresponding to the regions to form a single feature vector. This vector represents the camera motion of input video shot as a camera motion descriptor.

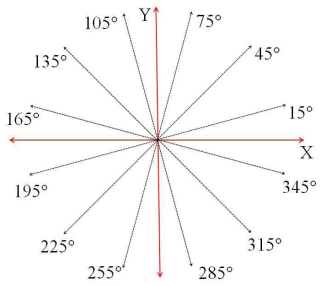


Fig. 6. Compactly represented motion quantization rule. The angle of each $pc^{(in)}_{(pq)}$ resides in one of the 12 ranges.

IV. Features extraction for Cinematographic Shot Classification

In this section, we extend CAMHID to perform the cinematographic shot classification task, which involves classifying cinematographic shots into the film directing semantic classes. In order to do that, we extend the feature space by extracting more features which considers the depth of a shot. In the following, we first describe the cinematographic shot framework and describe the directing semantic classes. We also discuss the need of additional features which enhance the discriminating capability in cinematographic directing semantic classes.

A. Cinematographic Framework

Film making is completely based on the film making grammars. The directors heavily apply these film making grammars on every single cinematographic shot. The directors' main intention is to visualize the screenplay by capturing a cinematographic shot through a set of camera motions and a set of viewpoints. Capturing grammatically correct cinematographic shots ensures the viewer attentions on the predetermined actor(s), object(s) or place(s) based on the screenplay (henceforth, we will use 'object' and 'actor' interchangeably). According to [33], two of the major issues which determine the viewer attentions are as follows.

- Camera operation: a set of well defined camera operations are routinely performed to ensure the presence of different actions from the object of interest on the view plane. The camera operation is a strong indication of the categories of actions happening from the directing point of view. For example, static shots are often used to display the emotion of the actors, and panning shots are often used to make sure the presence of object of interests on a view plane.
- Camera distance: The size of the objects of interest on a view plane carries different semantics from the direction sense. In cinematography, wide shots are often used to relate the object of interest with the surrounding environments while close-up shots are often used to display the emotional aspects on the actor's faces.

Based on these two issues, the construction of a taxonomy of the cinematographic directing semantics is discussed in the following. In directing semantic classes, the quality of camera operation and camera distance are more important than their quantity. For example, the differentiation between a slow zooming and a fast zooming is a subjective matter and quantitative measurements can be another research topic. Hence, in this work, we only consider the qualitative camera motion and distance. The relationship of the camera motion and object distance is important in directing semantic classes. The presence of a focused object makes the viewers feel like they are tracking the object. For example, a panning shot with a focused object gives a feeling to the viewers that the viewers personally track the object. However, without any focused object, a panning shot simply introduces a place to the viewers. In cinematography, this type of shots is only used to establish a new setting influencing viewers' mind. Scene composition is another aspect of cinematography which handles different issues such as distance of camera, camera angle and light. Among them, distance of camera is crucial as it determines the degree of emotional involvement of a viewer. In cinemas, we often see that highly emotional scenes are presented by using close-up/medium shots. Long distance shots are used to establish the context of a focused object. For the task of cinematographic shot classification, we group close-up and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology

medium shots into one class as it is not easy to distinguish the purposes of using these two types of shots. In a wide range of cinemas, the use of close-up and medium shots are very similar for similar emotional shots. However, long shots are mainly used for contextual tracking and contextual establishments.

B. Cinematographic Directing Semantic Classes

The cinematographic directing semantic classes are created using the basic directing element discussed in the previous subsection. In [33], the cinematographic shots are analyzed based on the directing elements and finally end up creating seven semantic classes: 1) stationary, 2) contextual-tracking, 3) focus-tracking, 4) focus-in, 5) focus-out, 6) establishment, and 7) chaotic shots. In reality, the directing semantic classes of cinematographic shots are not clearly categorized into other video domains (e.g., attack shots in soccer video). However, meaningful indexing is still possible using the introduced directing semantic classes. In the following, seven semantic classes are briefly described.

Stationary (S) shots: A significant portion of cinematographic shots are dialogue shots and the dialogue shots are mainly captured by using stationary shots. Stationary shots contain minimum amount of camera movements to concentrate the viewer attentions on the actor's activities. Figure 7(a) shows an example of static shots. In this particular case, as it can be seen, the shot is captured by focusing on the actress using close-up shot while the camera movement remains almost static.

Tracking shot: Tracking shots are the one which are captured by focusing on object(s) and follow along the direction of the movement. This type of shots is used to closely relate the viewers to the objects [42]. It makes the viewers feel like they are following the objects. Because of its own characteristics, tracking shots are considered as an important shot class. There are two types of tracking shots used in cinematography. 1) Contextual tracking (CT) shots which establish a relationship of an object with the context by capturing a bigger picture of the scene. The focused object is captured using a long shot so that the object looks smaller but provides scenic detail of the shooting set by using panning camera movements. Figure 7(b) shows an example of contextual tracking shot where the actress is being shot with a clear indication of the context (cityscape view). 2) Focus tracking (FT) is another variant of tracking shots which provides a closer view of the objects. The intention behind taking this type of shots is to focus on the closer detail of the object while tracking. Figure 7(c) shows an example of focus tracking shots.

Focus-in (FI) shots: In cinematography, focus-in shots are captured in two ways: 1) zooming in by shortening the focal length of the camera lens and 2) moving the camera to the object to shorten the camera distance for a closer view of the object. Both of these are mainly used to provide a greater detail of a focused object to highlight some important detail. Figure 7(d) shows an example of focus-in shot, where the object is getting bigger by changing the focal length.

Focus-out (FO) shots: Focus-out shots are used to detach emotional involvement of the viewers from an object or relax the viewers by changing the viewing space. This effect is usually achieved through zooming out or dolly out shots, as the camera gradually moves away from the subject and creates emotional distance. Figure 7(e) shows an example of focus-out shots by changing the position of the camera distance.

Establishment (E) shots: Establishment shots form another important directing semantic class which is used in cinematography. This type of shots is used to introduce a location to establish a relationship with the following sequence of shots. This type of shots is often taken by panning the camera without focusing on any particular object. Figure 7(f) shows an example of establishment shots.

Chaotic (C) shots: This type of shots are characterized by the chaotic movement of the camera to follow an object or an object action. Chaotic shots are the one which cannot be characterized to be anyone of the above mentioned classes. Generally, in this type of shots, a random camera motion happens due to focusing on an object's random motion. In order to represent fast action (or motions), directors apply this technique. In this shot type, it is not usually for the fast moving object to dominate viewer attention. Such shots are usually used to represent thrills and used more often in action films. Figure 7(g) shows an example of chaotic shots.

C. Feature Extraction from Cinematographic Shot Classes

The far right column of Figure 7 shows the CAMHID features of each camera motion type. Figure 7(a) shows an example static shot and the corresponding CAMHID which combines the features from the four prime corner regions. As it can be seen, histogram bins regarding no motion have more counts than the rest. Similarly, in other corresponding CAMHIDs, only camera motion information is incorporated. Although CAMHID is capable of describing the camera motion efficiently, it has a limitation to represent the camera distance of the cinematographic directing semantic classes. As mentioned, camera distance is another important characteristics to be considered for classifying cinematographic shots. In this section, we extend the features representing the depth to overcome that limitation. The additional set of features is extracted from the readily available MVI. As the corresponding MBs of MVIs roughly represent the regions which preserve the camera movement, computing the ratio of the local MVI regions estimates the rough local depth. In order to do that, for each $MVF^{i}_{(p,q)}$, $\forall p \in \{1,2,3\}$, $\forall q \in \{1,2,3\}$ and $\forall i \in \{1,2,\ldots,m\}$, we count the number of MVIs belonging to each local region ($mvi^{i}_{(x,y)} \in MVF^{i}_{(p,q)}$). Formally, we write:

$$C_{(p,q)} = \sum_{mvi^{i}_{(x,y)} \in MVF^{i}_{(p,q)}} 1 \quad (9)$$

where $C_{(p,q)}$ is the count of the number of MVI present in the local region (p,q) for the entire shot. Then, local counts are normalized. Formally, we write as follows.

$$\hat{C}_{(p,q)} = C_{(p,q)} / (v * t) \quad (10)$$

where v is number of possible motion vectors in a video frame and t is number of frames in an input video. The normalized features along with CAMHID is the feature vector used

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology

(a) static shot

(b) contextual tracking shot

(c) focus tracking shot

(d) focus in shot

(e) focus out shot

(f) establishment shot

(g) chaotic shot

Fig. 7. Cinematographic directing semantic shot classes and the corresponding CAMHID are shown in (a) - (g). The left column represents the first frames of the shots. Second column shows an intermediate frame of each shot. The third column represents the last frame of each shot. The right column shows the camera motion histogram descriptor by combining the local camera motion features on the four corners. The four corner local regions are identified on the last frame in (a) and the corresponding local histograms are identified in the corresponding CAMHID.

for classifying directing semantic classes of cinematographic shots. In the next section, we show the effectiveness of our proposed features.

V. Experimental Results

To show the performance of the proposed camera motion histogram descriptor, CAMHID is experimented and evaluated in this section. To do that, firstly, we evaluate the classification performance using a dataset based on basic camera motions. Secondly, we evaluate the classification performance on directing semantic classes of cinematographic shots using the second dataset. Each dataset consists of training and testing sets, and the performance is evaluated based on the precision, recall and f_1-scores on the testing datasets. The following subsections describe the detail of evaluation procedure.

A. Dataset Preparation and Feature Extraction

To evaluate the classification performance of the proposed CAMHID and its extension, we conduct experiments on two of our own created datasets. The first dataset (Dataset 1) is created based on the basic camera motion types. According to the basic camera motion types, video shots are classified into four basic classes: 1) static, 2) pan, 3) tilt and 4) zoom. The static shots are captured by placing or holding the camera

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology

firmly without any significant camera movement. For close-up view of objects, static video shots are often captured. The panning shots are captured by rotating the camera about the vertical axis. For the purpose of following objects horizontally, pan shots are often used. Tilt shots are captured by rotating the camera about the horizontal axis. Tilt shots are used for following objects in the vertical direction. Zoom shots are captured by changing the focal length of camera lens. Depending on the situation, we observe two types of zoom shots: zoom in and zoom out. A video shot is captured continuously, which may contain different types of camera motions (for video content structuring detail, please see [43]). In that case, the label is given based on most dominant camera motion in the shot. We segment video shots from Hollywood films and label the shots manually. For training the SVM classifier, we create training data from three Hollywood films. For evaluating the performance, a testing dataset is created. The shots are also taken from Hollywood films. The detailed breakdown of the testing data in Dataset 1 is given in Table I.

The second dataset (Dataset 2) is created based on the directing semantic classes of cinematographic shot. Similar to Dataset 1 preparation, we label a training set and a testing set manually. For training the SVM classifiers, we create training data from five Hollywood films. Then, for testing the classification performance, a testing dataset is created. The shots are also taken from Hollywood films. The detailed breakdown of each shot type in the testing data is given in Table II. The created datasets are available for public use [44].

TABLE I
DETAILED BREAKDOWN OF THE TESTING DATA IN DATASET 1.

	static	pan	tilt	zoom
no. of shot	1630	229	97	88

While extracting features from a video shot, we set the MB size to be of 16×16 pixels. In order to evaluate the performance of the proposed CAMHID, we first conduct an experiment to decide the optimum grid size. In order to do that, we extract the CAMHID features by segmenting the input video shots into different grid sizes and train an SVM using 3-fold and 5-fold cross validation approaches respectively. Then, we compute precision and recall rates to compare the performances of different grid sizes. Table III shows the performances of different grid sizes. As it can be seen, when grid size is 3×3, we achieve the best classification performance. For the rest of experiments, we select the grid size to be 3×3. After selecting the grid size, we conduct an experiment on optimum size of the training data size selection. Figure 8 shows the learning curve of Dataset 1. To conduct this experiment, we use a 3-fold cross validation approach. As it can be seen, with the changing size of the training data for Dataset 1, the training error remains steady. However, the cross validation error decreases gradually with the changing size of the training data size. Based on this experiment, we decide the training set size of Dataset 1 to be 75 shots from each class. Therefore, the total training set size is set to be 300 ($= 75 \times 4$) shots. A similar experiment is conducted on Dataset 2 to decide the optimum training shot size. It is found that the optimum training data size for Dataset 2 is 90 shots for each class. Therefore, our training data size of Dataset 2 is 630 ($= 90 \times 7$) shots.

TABLE II
DETAILED BREAKDOWN OF THE TESTING DATA IN DATASET 2.

	S	CT	FT	FI	FO	E	C
no. of shot	2110	452	912	190	39	180	1675
'%'	37.96	8.13	16.41	3.42	0.70	3.24	30.14

TABLE III
CLASSIFICATION PERFORMANCE ON DATASET 1 USING DIFFERENT GRID SIZES FOR CAMHID FEATURE EXTRACTION. FOR THIS EXPERIMENT WE USE SVM WITH RBF KERNEL WHEN $\gamma = 2^{-8}$ AND $C = 2^{9}$.

Grid	3-fold cross validation		5-fold cross validation	
	precision(%)	recall(%)	precision(%)	recall(%)
2×2	72.53 ± 3.31	69.16 ± 3.26	75.31 ± 2.16	78.33 ± 3.15
3×3	82.51 ± 2.12	77.33 ± 2.43	81.32 ± 2.29	82.52 ± 2.77
4×4	61.87 ± 4.72	64.62 ± 3.41	67.52 ± 4.21	65.72 ± 6.73
5×5	80.13 ± 3.16	77.03 ± 4.17	80.99 ± 3.22	79.41 ± 2.92

In CAMHID, local region features mainly describe the camera motion of the corresponding local region. To determine the effectiveness of different local regions' motion descriptors, a feature selection experiment is conducted. Figure 9 shows the f_1-scores on different feature sets extracted from Dataset 1. As it can be seen, with the changing size of the training data, the f_1-score measure using four corner regions outperforms the rest. Accordingly, we collect the features from the prime corner regions (top left, top right, bottom left and bottom right) in the rest of the experiments. From all four corner regions, we compute their features and put them sequentially to obtain the value of CAMHID.

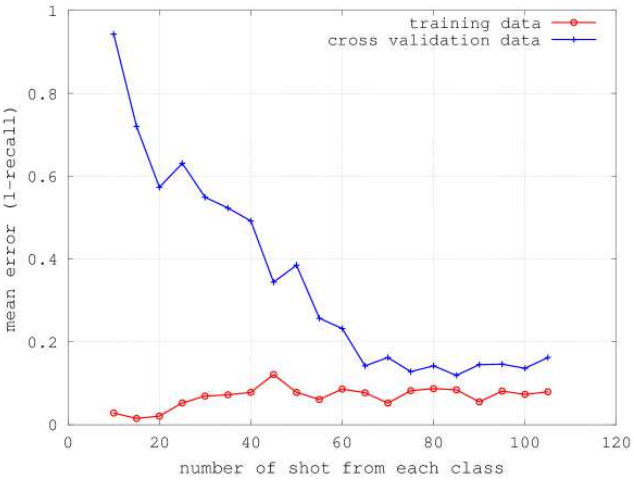


Fig. 8. Learning curve using different training data sizes on Dataset 1.

The best performance of the proposed method is achieved by setting the optimum values for k and n experimentally. In order to conduct the experiment to find the best k, we compute the average accuracies for each given k and changing n values. Figure 10(a) shows the experimental results for given

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology

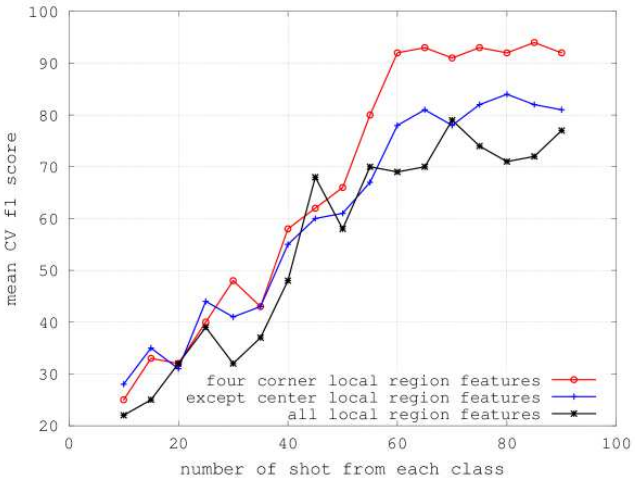


Fig. 9. Classification performance analysis using different feature sets.

k values. The biggest average value is recorded as the optimum k value. In order to conduct the experiment to find the best n, we compute the average accuracies for each given n and changing k values. Figure 10(b) shows the experimental results for given n values. The biggest average value is recorded as the optimum n value. As it can be seen in Figures 10(a) and 10(b), for $k = 10$ and $n = 5$ respectively, we achieve the optimum results. At the end of these experiments, the optimum k and n are set to be 10 and 5 respectively for conducting the rest of experiments.

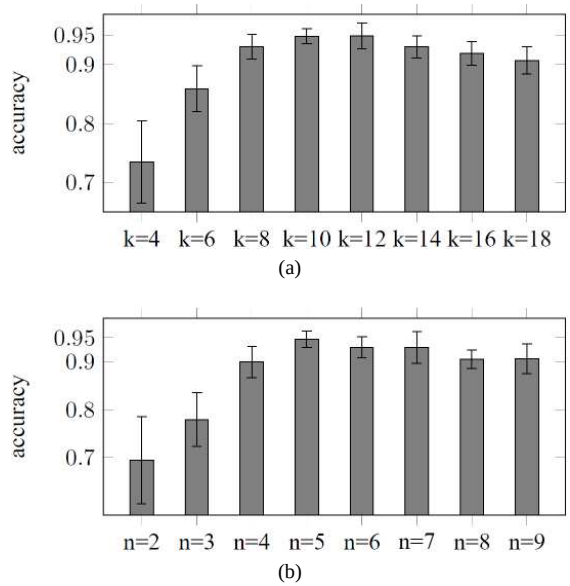


Fig. 10. The selection of optimum k and n selection based on classification accuracy measurement on Dataset 1. (a) Average classification accuracies for fixed k values and $n \in \{2,4,6,8\}$. (b) Average classification accuracies for fixed n value and $k \in \{6,8,10,12\}$. For both (a) and (b) cases, τ is set to be 0.25.

Threshold τ also has great influence to achieve the best performance. In order to find an optimum threshold value for τ, we compute classification accuracies for changing threshold values. Figure 11 represents the experimental results. As it can be seen, when τ is set to be 0.25, we achieve the optimum results. For the rest of experiments, we set τ to be 0.25.

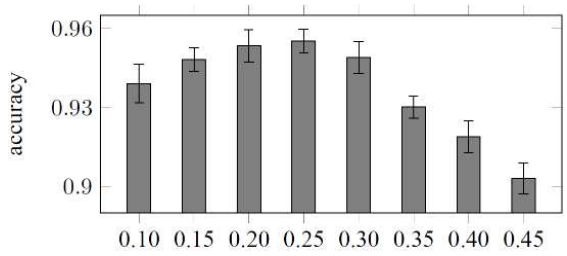


Fig. 11. The selection of τ representing the optimum motion inconsistency tolerance threshold by measuring classification accuracies for changing threshold values. The values along x-axis represent threshold τ values. For this experiment, we use $k = 10$ and $n = 5$.

B. SVM Classification

In this research, we use One-Against-One SVM technique for video shot classification. The effectiveness of One-Against-One approach has been presented in [45]. To summarize the One-Against-One multi-class SVM, let us assume that we have n training data in a d dimensional space belonging to c classes $\{\mathbf{x}^i, y^i\}$, $\mathbf{x}^i \in \mathbf{R}^d$, $i = 1,\ldots,n$, $y^i \in \{1,\ldots,c\}$. This approach constructs $c(c-1)/2$ classifiers using the training data. Each of the classifiers is obtained by using the training data of the corresponding two classes. For class i and class j, the binary classification problem is formally written as:

$$\begin{aligned} \min_{w_{ij}, b_{ij}, \xi_{ij}} \quad & \frac{1}{2}(w_{ij})^T w_{ij} + C \sum_t \xi_{ij}^t \\ & (w_{ij})^T \varphi(x^t) + b_{ij} \geq 1 - \xi_{ij}^t \quad \text{if } y^t = i \\ & (w_{ij})^T \varphi(x^t) + b_{ij} \leq -1 + \xi_{ij}^t \quad \text{if } y^t = j \end{aligned} \tag{11}$$

where ξ_{ij}^t is a non-negative slack variable, $\varphi(x^i)$ is a function to map x^i into a higher dimensional space and C is the penalty parameter. By minimizing $\frac{1}{2}(w_{ij})^T w_{ij}$, we want to maximise the margin, $\frac{2}{||w_{ij}||}$, between class i and class j. The penalty term $C \sum_t \xi_{ij}^t$ is used to reduce the number of training errors for linearly non-separable cases. The goal is to find an optimal separating hyperplane by obtaining a balance between the regularization term $\frac{2}{||w_{ij}||}$ and the training errors. To improve the separability, the data are mapped into a higher dimensional dot product space using the function φ. If the dot product space is expressed by $K(x^i, x^j) = \varphi(x^i) \cdot \varphi(x^j)$, then $K(x^i, x^j)$ is called a kernel function. The kernel used must meet Mercer's condition which is described in [46]. In this work, kernel selection is made experimentally. In the experiments, we consider three kernels, namely polynomial, sigmoid and RBF kernels. For each of the kernels, precision and recall rates are measured in 3-fold and 5-fold cross validation settings. Table IV shows the experimental results. As it can be seen, RBF kernel turns out to be the best performer in this experiment. Therefore, we select RBF kernel for conducting the rest of experiments. The accuracy of SVM classification depends on the values of two parameters C and γ. Careful selection of these two parameters is important. Otherwise the classifier may perform poorly in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology

TABLE IV
CLASSIFICATION ACCURACY MEASUREMENTS ON DATASET 1 USING DIFFERENT KERNELS FOR SVM CLASSIFICATION. FOR THIS EXPERIMENT WE SET $\gamma = 2^{-8}$ AND $C = 2^{9}$.

Kernels	**3-fold cross validation**		**5-fold cross validation**	
	precision (%)	recall (%)	precision (%)	recall (%)
Polynomial kernel (2nd order), $(\gamma x_i^T x_j + r)^d$, $r = 1$	91.33 ± 2.39	90.14 ± 1.82	92.83 ± 1.66	88.11 ± 2.95
Polynomial kernel (4th order)	92.51 ± 1.77	90.33 ± 1.82	91.21 ± 2.01	91.35 ± 2.10
Sigmoid kernel, $\tanh(\gamma x_i^T x_j + r)$, $r = 1$	88.87 ± 1.62	82.48 ± 2.72	89.05 ± 1.21	86.72 ± 2.99
RBF kernel, $e^{-\gamma \|x^i - x^j\|^2}$	92.99 ± 1.26	94.07 ± 1.09	93.02 ± 1.39	94.34 ± 0.97

the testing phase. A cross-validation approach is commonly used to determine the best parameters. We find the best penalty parameter C from the range $\{2^{-5}, 2^{-4}, \ldots, 2^{10}\}$ and width control parameter γ from the range $\{2^{-10}, 2^{-1}, \ldots, 2^{5}\}$.

Once the training is accomplished, the testing is done using the voting strategy called "Max Wins", proposed in [47]. In summary, for each comparison given data x, the sign of $((w_{ij})^T \varphi x + b_{ij})$ indicates the class of belonging. If the sign indicates that x belongs to class i, then the vote of class i is increased. Otherwise, the vote of class j is increased. At the end, the class with maximum vote is declared as the class of x. In case of draw, the lowest index class is considered as the winner.

C. Evaluation

The performance of CAMHID descriptor and cinematographic features are evaluated in this subsection. For both datasets, the effectiveness is shown by using confusion matrix and by comparing recall rates, precision rates and f_1 scores.

In Dataset 1, shots are classified into 4 classes: static, pan, tilt and zoom. The SVM classifier is trained using the training data and the performance is evaluated using the testing data. Table V shows the confusion matrix of the shot classification performance on Dataset 1. The recall rates, precision rates and f_1-scores are reported in Table VI. As shown in Table VI, the classification accuracy is reasonably high. We compare the performance of shot classification with three state-of-the-art approaches. In [15], video shots were classified into static, pan, tilt and zoom classes. Although the achieved classification accuracy was very high, the dataset size was very small (consisting of 45 shots only). Another approach was reported in [25] where video shots were classified based on camera movements. In that case, the authors used a dataset of only 32 MPEG-1 video sequences, which is also considered as a very small dataset. In [18], dense trajectory based technique was reported to describe video motions. The application shown in [18] was for action recognition. We test the performance of the descriptor based on dense trajectories only as mentioned in [18]. Other features such as HOG, HOF and MBH mentioned in [18] are irrelivant to our work, so we do not make any comparison on these features in this paper. Our shot classification results are compared with the methods described in [15], [25] and [18]. Figure 12 shows the recall, precision and f_1-score comparison. The average recall and precision rates of [15] are 89.29% and 88.0% respectively, rates of [25] are 97.66% and 90.03% respectively and rates of [18] are 77.94% and 74.49% respectively. As the datasets used in [15] and [25] are not made available for public access, we cannot directly compare our results with the results using those two methods. However, the method proposed in [18] has made their source code for public use [48]. Therefore, we use the available code to measure the performance in classifying video shots in basic camera motion classes using Dataset 1. As the static trajectories are eliminated during pre-processing, we have not used the static shots of Dataset 1 for training and testing the method in [18]. Therefore, for evaluation, we have used only pan, tilt and zoom shots to determine the performance. The average recall and precision rates of the proposed method are 94.23% and 95.27% respectively. The average f_1-score of [15], [25], [18] and the proposed method are 88.08%, 93.62%, 76.14% and 94.48% respectively. Although the approaches in [15] and [25] perform well in small datasets, the f_1-score performance of the proposed method outperforms the rest and it is considered to be more acceptable as the proposed method performs robustly on a much larger dataset. To summarize the above comparison, we claim that our method outperforms the method shown in [15], [25] and [18] on average in terms of recall rate, precision rate and f_1-score. Furthermore, our method is more promising and consistent.

TABLE V
CONFUSION MATRIX OF SHOT CLASSIFICATION USING DATASET 1.

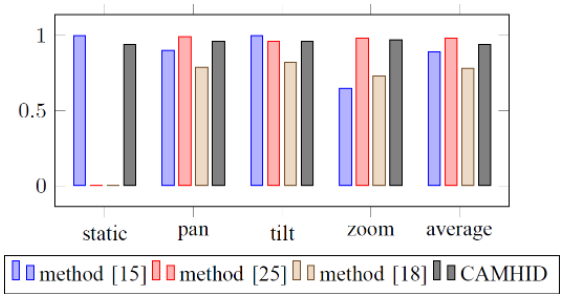
	static	pan	tilt	zoom
static	0.94	0.03	0.02	0.01
pan	0.01	0.96	0.01	0.02
tilt	0.02	0.01	0.96	0.01
zoom	0.01	0.02	0.00	0.97

TABLE VI
RECALL (R), PRECISION (P) AND f_1 SCORE (f_1) MEASURES OF SHOT CLASSIFICATION PERFORMANCE USING DATASET 1.

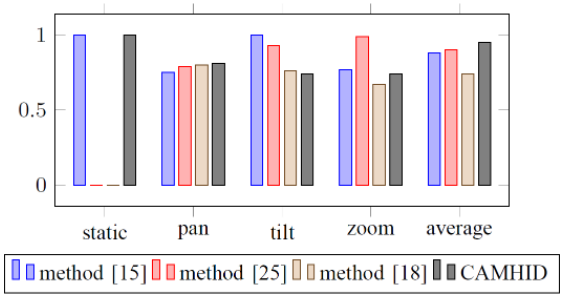
	static	pan	tilt	zoom	average
R	0.94	0.96	0.96	0.97	0.94
P	1.00	0.81	0.74	0.74	0.95
f_1	0.97	0.88	0.84	0.84	0.94

The performance of shot classification is further evaluated using Dataset 2 to show the ability of CAMHID and extended features in classifying cinematographic shots into the directing semantic classes. After training the SVM classifier using the training data of Dataset 2, the performance is evaluated using the testing data of Dataset 2. The confusion matrix is reported in Table VII. It is found that stationary shots are mostly confused with chaotic shots. This kind of misclassification mainly happens due to the threshold applied. This happens

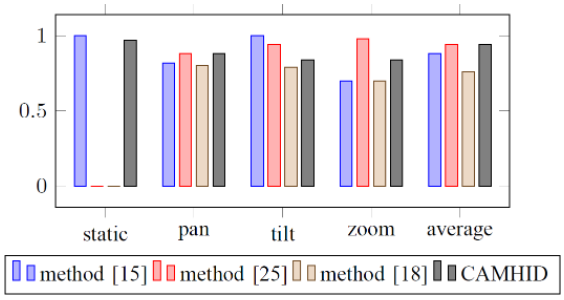
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology



(a) recall



(b) precision



(c) f_1-score

Fig. 12. Camera motion based video shot classification results comparison with [15], [25] and [18] using Dataset 1. (a) Recall rate comparison. (b) Precision rate comparison. (c) f_1-score comparison.

due to a small magnitude differences which fall near the borderline of motion magnitude. However, the amount of wrong classifications is at a minimum level. Focus tracking shots and Contextual tracking shots introduce another level of confusion. Since the establishment shots also have similar motion patterns, this shot type also introduces additional confusion in classification. Although there is a level of confusion, the classification results using the proposed method are still promising. Table VIII shows the detailed classification performance using recall rates, precision rates and f_1-scores. To evaluate the performance of our proposed method, we compare the result with a state-of-the-art methods described in [33]. Figures 13 shows the performance comparison of our method with the methods described in [33]. In [33], the authors proposed two methods to classify cinematographic shots into directing semantic classes. The first method classifies the shots with occlusion handling (OH) mechanism and the second method does so without occlusion handling (WOH) mechanism. As it can be seen in Figure 13(a), the recall rate of our method for all the classes is higher than the results using the state-of-the-art approach. Although for most of the classes, precision rates using our method are higher than those using the method shown in [33], for other classes our precision rates show a bit lower than the state-of-the-art. To show a fairer comparison, Figure 13(c) demonstrates the comparison results of f_1-scores. It is shown that, the proposed method has higher f_1-scores for the classes except contextual tracking and focus-out classes. The average f_1-scores of [33] with occlusion handling and without occlusion handling are 83.03% and 81.55% respectively. However, it turns out that the average f_1-score of the proposed method is 85.02% which is higher than the other two methods.

TABLE VII
CONFUSION MATRIX OF SHOT CLASSIFICATION USING DATASET 2.

	S	CT	FT	FI	FO	E	C
S	94.22	0.57	0.71	0.57	0.28	0.57	3.08
CT	0.44	87.61	5.53	1.55	0.88	3.10	0.88
FT	0.88	8.00	88.16	0.66	0.55	0.66	1.10
FI	0.53	2.11	1.05	91.58	0.53	3.16	1.05
FO	0.00	2.56	2.56	0.00	92.31	0.00	2.56
E	0.00	2.22	0.56	1.67	0.00	94.44	1.11
C	1.55	1.07	3.04	2.03	0.66	1.31	90.33

TABLE VIII
RECALL (R), PRECISION (P) AND f_1 SCORE (f_1) MEASURES OF SHOT CLASSIFICATION PERFORMANCE USING DATASET 2.

	S	CT	FT	FI	FO	E	C
R	94.22	87.17	88.16	88.95	89.74	94.44	90.33
P	98.08	77.87	89.23	72.53	55.56	73.91	94.68
f_1	96.11	82.26	88.69	79.91	68.63	82.92	92.45

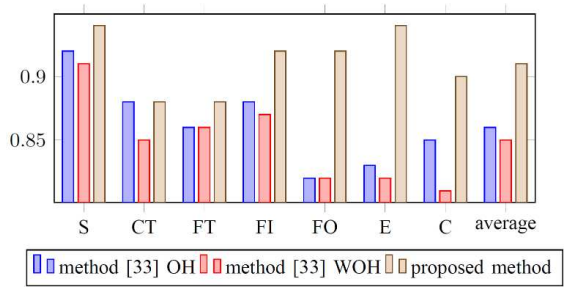
The CAMHID feature extraction processing time for a 30 frame video shot of size 688× 272 pixels using a PC (Windows XP, Microsoft Visual Studio 8.0 with OpenCV 2.3, Intel Core i5 2.5 GHz, 4 GB Memory) takes 6.79 seconds.

VI. CONCLUSION

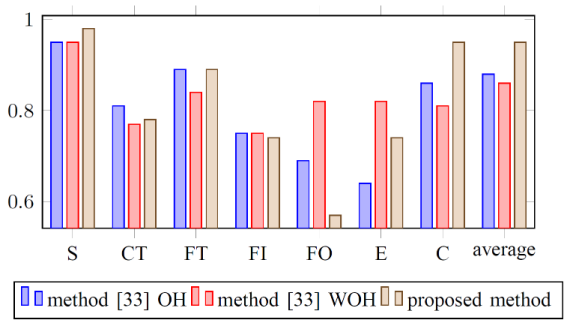
In this paper, a novel camera motion characterization technique has been proposed to compute the camera motion descriptor for an input video. The camera motion has been characterized by analyzing the extracted raw motion vectors in the temporal domain. The temporal characterization of the camera motions is then described by using a histogram, which combines local camera motion characterization features. We have applied the proposed technique to classify a video database into basic camera motion classes. We have further applied the proposed technique to classify cinematographic shots by extending the feature space. In the feature extension part, we consider the depth of the scene which is considered as one of the most important characteristics in cinematographic shot directing semantic classes. We have applied the motion descriptor with the extended feature on a separate dataset where shots are to be classified into directing semantic classes. We have evaluated and compared the performance of the proposed descriptor with state-of-the-art approaches. It has

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2345933, IEEE Transactions on Circuits and Systems for Video Technology

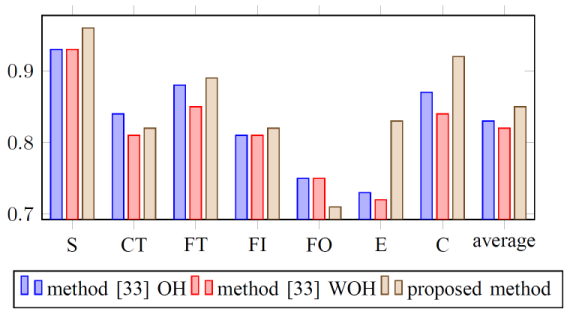
been demonstrated that the proposed descriptor has a strong capability to effectively discriminate different types of camera movements and shot types.



(a) recall



(b) precision



(c) f_1-score

Fig. 13. Cinematographic shot classification results in comparison with [33]. (a) Recall rate comparison. (b) Precision rate comparison. (c) f_1-score comparison.

REFERENCES

[1] 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," *Multimedia, IEEE Transactions on*, vol. 14, no. 4, pp. 975–985, 2012.

[2] Z.-J. Zha, M. Wang, Y.-T. Zheng, Y. Yang, R. Hong, and T.-S. Chua, "Interactive video indexing with statistical active learning," *Multimedia, IEEE Transactions on*, vol. 14, no. 1, pp. 17–27, 2012.

[3] M. Wang, X.-S. Hua, R. Hong, J. Tang, G.-J. Qi, and Y. Song, "Unified video annotation via multigraph learning," *Circuits and Syst. for Video Techn., IEEE Transactions on*, vol. 19, no. 5, pp. 733–746, 2009.

[4] J. Wu and M. Worring, "Efficient genre-specific semantic video indexing," *Multimedia, IEEE Transactions on*, vol. 14, no. 2, pp. 291–302, 2012.

[5] B. Geng, Y. Li, D. Tao, M. Wang, Z.-J. Zha, and C. Xu, "Parallel lasso for large-scale video concept detection," *Multimedia, IEEE Transactions on*, vol. 14, no. 1, pp. 55–65, 2012.

[6] M. Sjoberg, M. Koskela, S. Ishikawa, and J. Laaksonen, "Real-time large-scale visual concept detection with linear classifiers," in *Pattern Recognition (ICPR), 2012 21st International Conference on*. IEEE, 2012, pp. 421–424.

[7] W. Hu, N. Xie, L. Li, X. Zeng, and S. Maybank, "A survey on visual content-based video indexing and retrieval," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 41, no. 6, pp. 797–819, 2011.

[8] S.-F. Chang, W. Chen, H. J. Meng, H. Sundaram, and D. Zhong, "A fully automated content-based video search engine supporting spatiotemporal queries," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 8, no. 5, pp. 602–615, 1998.

[9] L.-Y. Duan, J. S. Jin, Q. Tian, and C.-S. Xu, "Nonparametric motion characterization for robust classification of camera motion patterns," *Multimedia, IEEE Transactions on*, vol. 8, no. 2, pp. 323–340, 2006.

[10] Y.-F. Ma, X.-S. Hua, L. Lu, and H.-J. Zhang, "A generic framework of user attention model and its application in video summarization," *Multimedia, IEEE Transactions on*, vol. 7, no. 5, pp. 907 – 919, oct. 2005.

[11] P. Bouthemy, M. Gelgon, and F. Ganansia, "A unified approach to shot change detection and camera motion characterization," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 9, no. 7, pp. 1030–1044, 1999.

[12] T. T. de Souza and R. Goularte, "Video shot representation based on histograms," in *Proceedings of the 28th Annual ACM Symposium on Applied Computing*. ACM, 2013, pp. 961–966.

[13] Y.-P. Tan, D. D. Saur, S. R. Kulkami, and P. J. Ramadge, "Rapid estimation of camera motion from compressed video with application to video annotation," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 10, no. 1, pp. 133–146, 2000.

[14] 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," *Multimedia, IEEE Transactions on*, vol. 7, no. 4, pp. 648–666, 2005.

[15] C.-W. Ngo, T.-C. Pong, and H.-J. Zhang, "Motion analysis and segmentation through spatio-temporal slices processing," *Image Processing, IEEE Transactions on*, vol. 12, no. 3, pp. 341–355, 2003.

[16] M. V. Srinivasan, S. Venkatesh, and R. Hosie, "Qualitative estimation of camera motion parameters from video sequences," *Pattern Recognition*, vol. 30, no. 4, pp. 593–606, 1997.

[17] T. Brox and J. Malik, "Large displacement optical flow: descriptor matching in variational motion estimation," *PAMI, IEEE Transactions on*, vol. 33, no. 3, pp. 500–513, 2011.

[18] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 3169–3176.

[19] H. Wang, C. Schmid *et al.*, "Action recognition with improved trajectories," in *International Conference on Computer Vision*, 2013.

[20] G. Abdollahian, C. M. Taskiran, Z. Pizlo, and E. J. Delp, "Camera motion-based analysis of user generated video," *Multimedia, IEEE Transactions on*, vol. 12, no. 1, pp. 28–41, 2010.

[21] J. Oh and P. Sankuratri, "Automatic distinction of camera and object motions in video sequences," in *Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference on*, vol. 1, pp. 81–84 vol.1.

[22] S. Lee and I. Hayes, M.H., "Real-time camera motion classification for content-based indexing and retrieval using templates," in *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, vol. 4, May, pp. IV–3664–IV–3667.

[23] D.-J. Lan, Y.-F. Ma, and H.-J. Zhang, "A systemic framework of camera motion analysis for home video," in *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, vol. 1, Sept., pp. I–289–92 vol.1.

[24] J.-G. Kim, H. S. Chang, J. Kim, and H.-M. Kim, "Efficient camera motion characterization for MPEG video indexing," in *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*, vol. 2. IEEE, 2000, pp. 1171–1174.

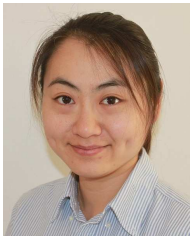
[25] R. Ewerth, M. Schwalb, P. Tessmann, and B. Freisleben, "Estimation of arbitrary camera motion in MPEG videos," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 1. IEEE, 2004, pp. 512–515.

[26] S. Jeannin and A. Divakaran, "MPEG-7 visual motion descriptors," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 11, no. 6, pp. 720–724, 2001.

[27] R. Fablet, P. Bouthemy, and P. Perez, "Nonparametric motion characterization using causal probabilistic models for video indexing and

retrieval," *Image Processing, IEEE Transactions on*, vol. 11, no. 4, pp. 393–407, 2002.
[28] C.-W. Ngo, T.-C. Pong, and H.-J. Zhang, "Motion-based video representation for scene change detection," *International Journal of Computer Vision*, vol. 50, no. 2, pp. 127–142, 2002.
[29] M. Irani and P. Anandan, "Video indexing based on mosaic representations," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 905–921, 1998.
[30] Y.-P. Tan, D. D. Saur, S. R. Kulkami, and P. J. Ramadge, "Rapid estimation of camera motion from compressed video with application to video annotation," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 10, no. 1, pp. 133–146, 2000.
[31] Z. Duric and A. Rosenfeld, "Image sequence stabilization in real time," *Real-Time Imaging*, vol. 2, no. 5, pp. 271–284, 1996.
[32] Y.-S. Yao and R. Chellappa, "Electronic stabilization and feature tracking in long image sequences," DTIC Document, Tech. Rep., 1995.
[33] H. L. Wang and L. F. Cheong, "Taxonomy of directing semantics for film shot classification." *IEEE Trans. Circuits Syst. Video Techn.*, pp. 1529–1542, 2009.
[34] N. V. Patel and I. K. Sethi, "Video shot detection and characterization for video databases," *Pattern Recognition*, vol. 30, no. 4, pp. 583–592, 1997.
[35] W. Xiong and J. C.-M. Lee, "Efficient scene change detection and camera motion annotation for video classification," *Computer Vision and Image Understanding*, vol. 71, no. 2, pp. 166–181, 1998.
[36] Y.-F. Ma and H. Zhang, "Contrast-based image attention analysis by using fuzzy growing." in *ACM Multimedia'03*, 2003, pp. 374–381.
[37] H. Lim and H. W. Park, "A symmetric motion estimation method for motion-compensated frame interpolation," *Image Processing, IEEE Transactions on*, vol. 20, no. 12, pp. 3653–3658, 2011.
[38] A.-M. Huang and T. Q. Nguyen, "A multistage motion vector processing method for motion-compensated frame interpolation," *Image Processing, IEEE Transactions on*, vol. 17, no. 5, pp. 694–708, 2008.
[39] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the h.264/avc video coding standard," *Circuits and Syst. for Video Techn., IEEE Transactions on*, vol. 13, no. 7, pp. 560 –576, july 2003.
[40] V. Klema and A. Laub, "The singular value decomposition: Its computation and some applications," *Automatic Control, IEEE Transactions on*, vol. 25, no. 2, pp. 164–176, 1980.
[41] M. Black, Y. Yacoob, A. Jepson, and D. Fleet, "Learning parameterized models of image motion," in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, jun 1997, pp. 561 –567.
[42] C. Dorai and S. Venkatesh, *Media Computing: Computational Media Aesthetics*, ser. The Kluwer International Series in Video Computing / Editor Mubarak Shah. Kluwer, 2002.
[43] M. Wang and H. Zhang, "Video Content Structuring," vol. 4, no. 8, p. 9431, 2009, revision 91922.
[44] M. A. Hasan, M. Xu, X. He, and C. Xu. CAMHID: Camera motion histogram descriptor and its application to cinematographic shot classification. [Online]. Available: http://data.research.uts.edu.au/public/TCSVT-Hasan/
[45] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *Neural Networks, IEEE Transactions on*, vol. 13, no. 2, pp. 415–425, 2002.
[46] V. N. Vapnik, *The nature of statistical learning theory*. New York, NY, USA: Springer-Verlag New York, Inc., 1995.
[47] J. Friedman, "Another approach to polychotomous classifcation," Technical report, Stanford University, Department of Statistics, Tech. Rep., 1996.
[48] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu. Dense trejectories video description. [Online]. Available: https://lear.inrialpes.fr/people/wang/dense_trajectories

Muhammad Abul Hasan received B.Sc. degree in Computer Science from East West University, Dhaka, Bangladesh in 2003 and M.S. degree from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Republic of Korea in 2009. Currently he is a Ph.D. student in the School of Computing and Communications, Faculty of Engineering and Information Technology, University of Technology, Sydney (UTS), Australia. His current research interests include machine learning and computer vision, medical image processing, and multimedia content analysis.



Min Xu received the B.E. degree from University of Science and Technology of China, in 2000, M.S. degree from National University of Singapore in 2004 and Ph.D. degree from University of Newcastle, Australia in 2010. Currently, she is a lecturer in School of Computing and Communications, Faculty of Engineering and IT, University of Technology, Sydney. Her research interests include multimedia content analysis, video adaptation, interactive multimedia, pattern recognition and computer vision.

Xiangjian He received the Bachelor of Science degree in Mathematics from Xiamen University in 1982, the Master of Science degree in Applied Mathematics from Fuzhou University in 1986, the Master of Science degree in Information Technology from the Flinders University of South Australia in 1995, and the PhD degree in Computing Sciences from the University of Technology, Sydney, Australia in 1999. From 1982 to 1985, he was with Fuzhou University. From 1991 to 1996, he was with the University of New England. Since 1999, he has been with the University of Technology, Sydney, Australia. He is a full professor and the Director of Computer Vision and Recognition Laboratory. He is also working as a Deputy Director of the Research Centre for Innovation in IT Services and Applications at the University of Technology, Sydney.

Changsheng Xu is a Professor of Institute of Automation, Chinese Academy of Sciences, Beijing, and Executive Director of China-Singapore Institute of Digital Media. He was with Institute for Infocomm Research, Singapore, from 1998 to 2008. He was with the National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences from 1996 to 1998. His research interests include multimedia content analysis, indexing and retrieval, digital watermarking, computer vision, and pattern recognition. He published over 170 papers in those areas. Dr. Xu is an Associate Editor of *ACM/Springer Multimedia Systems Journal*. He served as Program Co-Chair of 2009 ACM Multimedia Conference, Short Paper Co-Chair of ACM Multimedia 2008, General Co-Chair of 2008 Pacific-Rim Conference on Multimedia and 2007 Asia-Pacific Workshop on Visual Information Processing (VIP2007), Program Co-Chair of VIP2006, Industry Track Chair and Area Chair of 2007 Intern Industry Track Chair, and Area Chair of 2007 International Conference on Multimedia Modeling. He also served as Technical Program Committee Member of major international multimedia conferences, including ACM Multimedia Conference, International Conference on Multimedia & Expo, Pacific-Rim Conference on Multimedia, and International Conference on Multimedia Modeling. He received the 2008 Best Editorial Member Award of *ACM/Springer Multimedia Systems Journal*. He is a member of ACM.