Classification of Cinematographic Shots Using Lie Algebra and its Application to Complex Event Recognition

Subhabrata Bhattacharya, *Member, IEEE*, Ramin Mehran, *Member, IEEE*, Rahul Sukthankar, *Member, IEEE*, and Mubarak Shah, *Fellow, IEEE*

***Abstract*—In this paper, we propose a discriminative representation of a video shot based on its camera motion and demonstrate how the representation can be used for high level multimedia tasks like complex event recognition. In our technique, we assume that a homography exists between a pair of subsequent frames in a given shot. Using purely image-based methods, we compute homography parameters that serve as coarse indicators of the ambient camera motion. Next, using Lie algebra, we map the homography matrices to an intermediate vector space that preserves the intrinsic geometric structure of the transformation. The mappings are stacked temporally to generate vector time-series per shot. To extract meaningful features from time-series, we propose an efficient linear dynamical system based technique. The extracted temporal features are further used to train linear SVMs as classifiers for a particular shot class. In addition to demonstrating the efficacy of our method on a novel dataset, we extend its applicability to recognize complex events in large scale videos under unconstrained scenarios. Our empirical evaluations on eight cinematographic shot classes show that our technique performs close to approaches that involve extraction of 3-D trajectories using computationally prohibitive structure from motion techniques.**

***Index Terms*—Cinematographic shots, homography, lie algebra, multimedia event recognition, shot classification.**

I. Introduction

SHOT level classification of videos has been an interesting field in computer vision research, especially due to its applicability in diverse domains. These include: content based video search [12], film genre classification [8], [23] and video quality analysis [4]. With the constant need to improve online video search, interesting research [6], [8], [12], [17], [23], [29] have been pursued that address shot classification from multiple perspectives: low-level textures, intensity, high-level objects and scenes etc. While these are meaningful at content level, they are unable to capture the ambient camera motion which replicates the narrative human eye and hence are far more semantically challenging.

Camera motion in authored videos (commonly pan, tilt or zoom), are directly correlated with high-level semantic concepts described in the shot. For example, a *tracking shot* in which a camera undergoes translation on a moving platform indicates the presence of a *following* concept. Detection of such useful concepts can be used by current video search engines at a later stage to perform high-level content analysis such as detection of events from videos. This motivates us to explore the possibilities of using pure camera motion to solve the shot classification problem. Camera motion parameters, also known as telemetry, are very difficult to obtain directly as few video cameras are equipped with sophisticated sensors that can provide such accurate measurements. Furthermore, telemetry data is not generally available and is certainly not present in Internet or broadcast video. Hence, we resort to a purely image based technique to robustly estimate homographies which are coarse indicators of the camera motion incurred during capture. However, homographies are not meaningful features for discriminative classification of shots as different parameters in a homography matrix quantify different planar relationship (scale, rotation, etc.) and

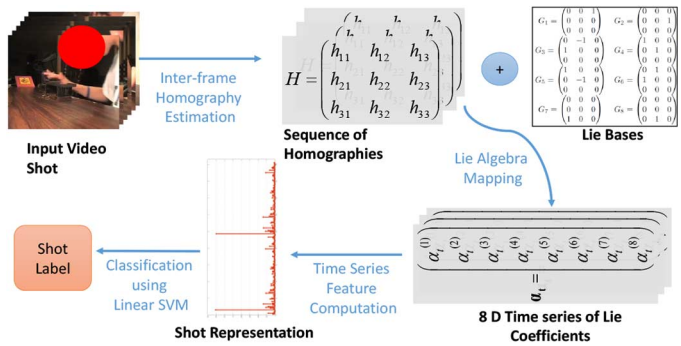


Fig. 1. A schematic diagram showing the various processes involved in our proposed approach towards classification of a typical shot. We build our complex event recognition computational pipeline (discussed in Sect. IV-E based on the above methodology. Please refer to text for a detailed explanation.

Manuscript received September 25, 2012; revised September 04, 2013 and September 04, 2013; accepted November 14, 2013. Date of publication January 16, 2014; date of current version March 13, 2014. This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20066. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Christophe De Vleeschouwer.

S. Bhattacharya and M. Shah are with the Center of Research in Computer Vision, University of Central Florida, Orlando, FL 32826 USA (e-mail: subh@cs.ucf.edu; shah@cs.ucf.edu).

R. Mehran is with Microsoft Corp., Redmond, WA, 98052 USA (e-mail: rmehran@microsoft.com).

R. Sukthankar is with Google Research, Mountain View, CA 94043 USA (e-mail: rahuls@cs.cmu.edu)

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2014.2300833

cannot be treated in separation. Also, since homographies belong to the projective group (i.e., are not closed under vector subtraction or scalar multiplication), they are not suitable for classifiers such as linear SVMs or Nearest Neighbors. Therefore representing the ambient motion in a principled manner is extremely important, to classify a shot.

While there exist methods [24], [30] to estimate camera motion using full 3D reconstruction of a scene, we argue that our method achieves a reasonable trade-off between high-accuracy and prohibitive computational cost. This enables us to contribute a global feature based on camera motion which can be used for large scale video analysis.

To this end, we propose the following methodology (Fig. 1) to represent the camera motion extracted from a video: (1) Given a shot, pairwise homographies are computed between the consecutive frames, (2) Next we map them to a linear space using Lie algebra defined under Projective Group (3) Coefficients of this linear space are used to construct multiple time series (4) Representative features are computed from these time series for discriminative classification. A schematic diagram of our computational pipeline is shown in Fig. 1.

II. Related Work & Our Contributions

A full survey of shot classification is beyond the scope of this paper. Please refer to [10], [22], [23] for a good background. In one of the earliest efforts [25], the authors qualitatively estimate camera pan, tilt, zoom, and roll from a sequence of images. [32] extends the idea to shots with camera rotation, where mutual information between motion vectors is utilized. In [21], Park *et al.* explored further using linear combination of motion vectors. While these techniques relied on optical flow to obtain motion vectors, a few teams in TRECVID 2005 [20] used motion vectors provided in MPEG stream for this purpose.

From a different perspective, Fablet *et al.* [9] make use of local spatio-temporal derivatives to classify dynamic content of shots without motion segmentation. Wang and Cheong on the other hand, explore the possibilities of using a Markov Random Field based motion foreground vs background labeling framework [27] together with cinematographic principles to classify pan, tilt, zoom, track and establishing shots. Approaches proposed in [14], [26], [28] focus on specific semantic classes of videos. For example, in [28] the authors employ structure tensor histograms to determine motion characteristics in shots from action movies. Similarly, [14], [26] leveraged on specific cinematographic techniques that only applied to sports videos to address the shot classification problem.

In this paper, we make the following contributions: (1) We obtain global camera motion by robustly estimating frame to frame homographies unlike approaches [9], [21], [25], [32] that rely on local optical flow based techniques, which are often noisy or full structure from motion based approach [31], which is computationally expensive, (2) Compared to approaches [26] that use homographies directly for classification, our lie-algebra based representation homographies is more accurate, (3) Our global features computed from a shot consider temporal continuity between frames, are superior to orderless bag of words techniques used in [20], thereby eliminating any need for explicit temporal alignment of shots of unequal lengths, (4) Our representation is capable of classifying a broader category of shots as compared to [19], [21], [25], [31], [32]. Our dataset consists of eight cinematographic shot classes [1] which we are freely distributing to the research community, (5) Our method is more versatile than approaches suggested in [14], [26], [28] which apply to specific domains such as movies or sports. It also requires fewer parameters to adjust as compared to [27], which require explicit motion segmentation, and (6) Finally, this is the first work to show how our novel camera motion representation can be used as a complementary feature for recognition of complex events in unconstrained Internet videos.

III. Approach

A. A Cinematography Primer

A complete list of cinematographic techniques can be found in [1]. In this paper we focus on the following cinematographic shot classes: aerial, bird-eye, crane, dolly, establishing, pan, tilt and zoom. The Fig. 2 shows the ambient camera motion in each shot class except for establishing shots where the camera remains stationary. Both aerial and bird-eye shots are captured from a high flying platform. The former class of shots have a strong perspective distortion, while the latter being taken from a camera ortho-normal to the ground plane, show affine transformation properties between consecutive frames. Crane or boom shots involve vertical motion of camera which may include simultaneous movement along x or y axes A dolly shot, on the other hand, is taken by placing the camera on a platform that moves smoothly on ground without any movement along z-axis. Pan and tilt shots are associated with camera rotation along z and y-axes respectively. A zoom shot, does not involve any physical camera motion. It is characterized by the change in focal length, which is an internal camera parameter. All of these motions can be efficiently captured by the projective transformation model.

B. Motion Parameter Extraction

We employ a feature based method to estimate homography between consecutive frames or every n-th frame of a given shot. In our technique, SURF features [2] are detected on each pair of frames on a dense sampling basis. Correspondence between features are established using a nearest neighbor search. We use the open source implementation available in [3] for this purpose.

Given two sets of corresponding points $\{(x_1, y_1), \ldots (x_n, y_n)\}$, and $\{(x'_1, y'_1), \ldots (x'_n, y'_n)\}$ a homography $H = \{h_{ij}\}$, is a 3×3, 8 degrees of freedom projective transformation that models the relationship between two points (x, y) and. (x', y'). in the following way:

$$x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}}, y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}}. \quad (1)$$

Using a set of N corresponding points, we can form the following linear system of equations:

$$[{a_{x_1}}^T, {a_{y_1}}^T, {a_{x_2}}^T, {a_{y_2}}^T, \ldots, {a_{x_N}}^T, {a_{y_N}}^T]^T \mathbf{H} = 0, \quad (2)$$

where $\mathbf{H}, a_x, a_y$ are the following vectors:

$$\begin{aligned} \mathbf{H} &= [h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33}]^T, \\ a_x &= [-x_i, -y_i, -1, 0, 0, 0, x'_i x_i, x'_i y_i, x'_i]^T, \\ a_y &= 0, 0, 0, -x_i, -y_i, -1, y'_i x_i, y'_i y_i, y'_i]^T. \end{aligned} \quad (3)$$

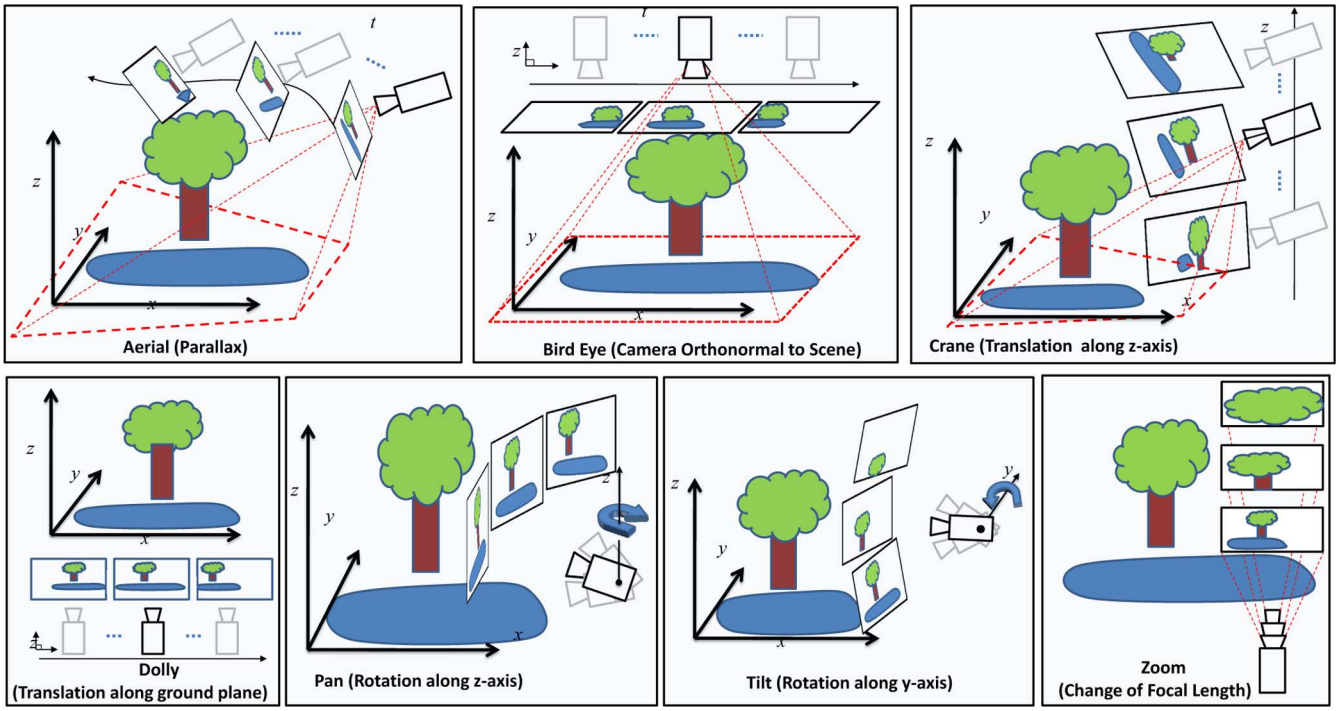


Fig. 2. Schematic diagram showing different types of shots–**Top Row**: The first two figures show aerial and bird-eye shots. In both shots the camera is attached to a high flying platform and has its characteristic motion in 3D. In case of aerial shot, there is a strong perspective which is absent in case of bird-eye shots. The third figure shows a crane shot where the crane moves along z-axis with no simultaneous motion along x or y axis. Red lines show the field of views of each camera in a particular shot setting. **Bottom Row**: The first figure shows a dolly shot where the camera is on a platform that undergoes smooth translation along the ground plane. The next three figures show pan, tilt and a zoom shot. Pan and tilt shots are associated with camera rotation along z-axis and y-axis respectively. A zoom shot as shown, does not involve any physical camera motion. The change of focal length in this case is indicated using dotted lines with different sized lenses.

Eqn. (2) is solved using random sampling consensus technique [11] that iteratively minimizes the back-projection error, defined as:

$$\sum_i (x_i' - x_i'')^2 + (y_i' - y_i'')^2 \tag{4}$$

where,

$$x_i'' = \left(\frac{h_{11}x_i + h_{12}y_i + h_{13}}{h_{31}x_i + h_{32}y_i + h_{33}} \right), \tag{5}$$

and,

$$y_i'' = \left(\frac{h_{21}x_i + h_{22}y_i + h_{23}}{h_{31}x_i + h_{32}y_i + h_{33}} \right) \tag{6}$$

In practice, since frame-to-frame homographies do not map image points to infinity, the last element of the matrix, h_{33} is set to 1, which gives 8 transformation parameters to be computed between each image pair. Except for h_{13} and h_{23}, which indicate translational motion along x and y axes respectively, these parameters are not individually meaningful (this is experimentally validated in Section IV). However, since they represent a transformation, they can be mapped efficiently to some subspace that preserves the internal structure of the transformation. We resort to Lie algebra for projective group to establish this mapping.

C. Lie Algebra Mapping of Projective Group

Recently, Lie algebra is made popular by the authors of [13] to solve a wide range of tasks in computer vision. The algebraic representation of affine and projective transforms facilitates the use of learning methods by providing an equivalent vector space that preserves the geometric transformation structure under linear operations.

Homographies belong to the projective group which has multiplicative structure. This group is neither closed under vector addition nor scalar multiplication, and therefore treating it as a linear space for classification results in undesirable effects. This is because nearest neighbor or SVM based classification do not consider geometric constraints which apply to projective groups since they belong to a nonlinear manifold. The Lie algebra mapping of the projective group is a 3×3 matrix in homogeneous space which relates to the homography matrix H through an exponential function as:

$$H = \exp(M) = I + \sum_{k=1}^{\infty} \frac{1}{k!} M^k. \tag{7}$$

Alternatively,

$$M = \log(H) = \sum_{k=1}^{\infty} \frac{-1^{k+1}}{k} (H - I)^k. \tag{8}$$

Due to linearity in the Lie algebraic representation, M can be written as the linear combination of orthogonal bases as:

$$M = \sum_{i=1}^{8} \alpha_i G_i \tag{9}$$

where, G_i are also called generators of the Lie group [7]. It is shown in [7] that for infinitesimal transformations near identity,

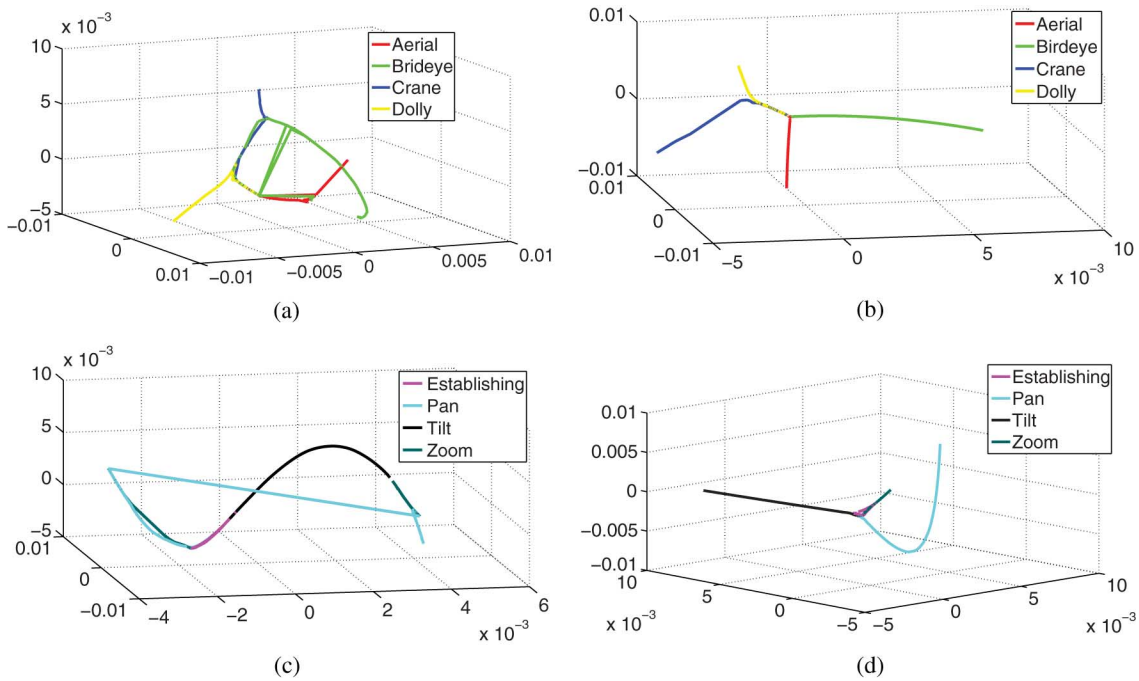


Fig. 3. Trajectory based visualization of different shots obtained from raw homography sequence and their linear space mapped counterparts. (a),(c) can be interpreted as pure homography sequences while (b),(d) are their respective Linear space mapping using Lie algebra of Projective groups. Time is shown in z-axis and scaled independently to improve visualization. The x, y axes represent dimensionally reduced H, and L coefficients from a shot sequence using PCA and do not have any physical interpretation. Note the clutter in projective case.

the higher order terms in Eqn. (8) can be ignored. Thus, α_i can be computed by projecting the first order approximation of M i.e. $H - I$ on G_i. In principle, as long as the bases are orthogonal, Eqn. (9) is valid. We select the following generators since they are already established in literature [7] and have injective mapping with the projective group of transformations:

$$G_1 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} G_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} G_3 = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$
$$G_4 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} G_5 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix} G_6 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$
$$G_7 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} G_8 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \quad .(10)$$

Using the above, the frame by frame homography matrix can be represented by $\{\alpha_i\}$ in an equivalent vector space. The effect of this transformation can be well explained using Fig. 3. The Figs. 3(a) and 3(c) are pure homography sequences from different shot categories, alphabetically arranged into 2 groups. In contrast, Figs. 3(b) and 3(d) are Linear space mapping of the original homography sequences, obtained using Eqn 9-10. Time is shown in z-axis and scaled independently to improve visualization. The x, y axes represent homographies and their corresponding linear space mappings, reduced to 2-D using PCA and do not have any physical interpretation. In both Figs. 3(a) and 3(c), we observe how cluttered these trajectories appear in the projective space, while in case of Figs. 3(b) and 3(d), they appear more distinct, arguing in favor of our original hypothesis.

Fig. 4 provides a more convincing evidence towards how our Lie algebra based representation is more efficient in terms of segregating different classes of shots in contrast to their original projective space. Both Fig. 4(a) and 4(b) show color coded similarity matrices in bag-of-X representations computed from homographies and the corresponding linear mapping, respectively. For more details on the vocabulary chosen for Bag-of-X please refer to Sect. IV. These are being referred to as Bag-of-H and Bag-of-LC throughout the rest of the paper. For ease of understanding, shot samples are arranged alphabetically according to their respective class names with aerial samples (top - aerial, bottom - zoom). We observe high intra-class similarity and inter-class dissimilarity in case of Fig. 4(b) as opposed to Fig. 4(a). In case of both similarity matrices, we observe strong degree of similarity in the establishing shot category (5th from top), which is mainly due to the identity nature of the homography matrices.

The bag-of-LC model provides reasonably discriminative representation for a given shot, and can be used as a generic shot-level descriptor. However, we intend to incorporate the temporal relationship between the Lie group coefficients which is not captured in the bag-of-LC model. With this motivation we proceed to the next step where we present an efficient manner to extract the temporal relationship in a more meaningful way, leading to a compact descriptor per shot, without the requirement of additional vector quantization.

D. Feature Extraction from Time Series

The different time series obtained after sequential arrangement of the Lie-group coefficients could be imagined as trajectories. It may be tempting to fit these trajectories into splines or

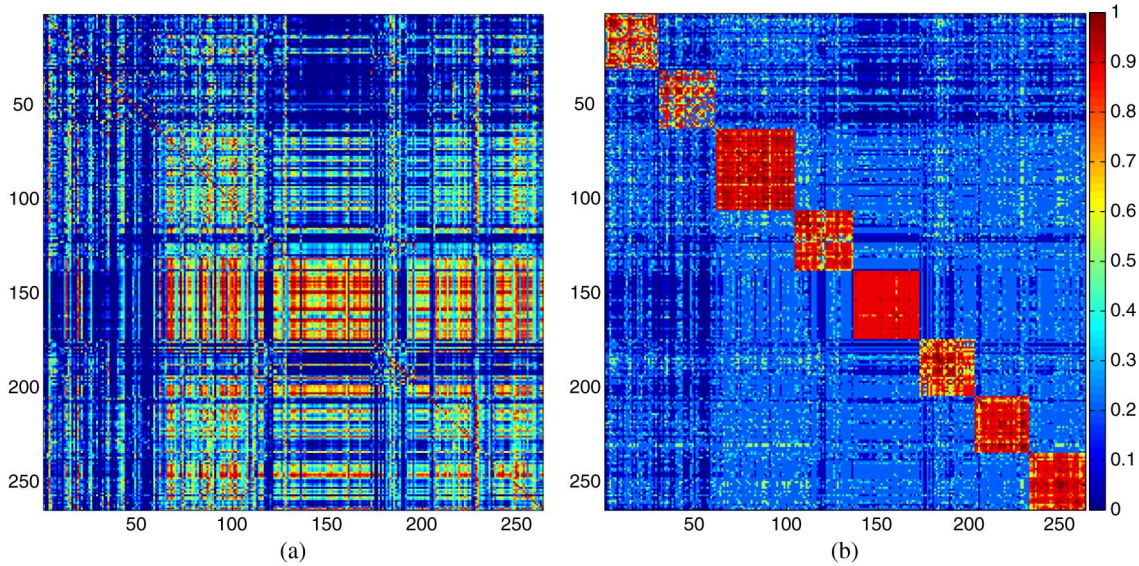


Fig. 4. Intra class similarity in original projective space and proposed Lie Space. The similarity is not meaningful in case of (a), while they are more pronounced in (b). In the latter case 8 distinct blocks are clearly visible, corresponding to the shot classes arranged in alphabetical order of their class name.

simple models by finding the parameters that best explains the data, however classification using these models is complex and can be badly distorted by outliers as they usually do not have any structural interpretation. We hypothesize that the temporal order of these coefficients is crucial for classification. This motivates us to explore computation of features from multi-dimensional time-series data from the perspective of linear dynamical systems (LDS). Modeling our time-series data using LDS is a reasonable assumption as (a) the Lie-algebra coefficients span a well defined linear space in a given shot, and (b) coefficient vector at a time step follows single chain Markov property.

Thus, using foundations from LDS theory, we can describe any coefficient vector using the following set of equations:

$$\boldsymbol{\alpha}_t = K\mathbf{x_t} + \epsilon_\mathbf{t}, \tag{11}$$

$$\mathbf{x_t} = \phi\mathbf{x_{t-1}}; \quad \mathbf{x_0} \quad \text{given}, \tag{12}$$

where, is the observation matrix $\in \mathbb{R}^{p\times\theta}$ that maps each observed time-step α_t to a relatively lower dimensional hidden state vector $\mathbf{x_t} \in \mathbb{R}^\theta$, $\epsilon_\mathbf{t} \sim \mathcal{N}(0,1)$ (noise), and ϕ is the dynamics or transition matrix $\in \mathbb{R}^{\theta\times\theta}$ which relate K s the current hidden state with the previous hidden state.

One popular way to indirectly characterize the system defined in Eqn. (12) is to analyze the Eigenspace of the Hankel matrix constructed from this system [5]. Given a sequence of coefficient vectors of length n $(\boldsymbol{\alpha}^{(0)} \ldots \boldsymbol{\alpha}^{n})$ the Hankel matrix can be can be constructed as follows, whose entries are the same along the anti-diagonals:

$$Q_i = \begin{pmatrix} \boldsymbol{\alpha_i^{(0)}} & \boldsymbol{\alpha_i^{(1)}} & \boldsymbol{\alpha_i^{(2)}} & \ldots & \boldsymbol{\alpha_i^{(n-r+1)}} \\ \boldsymbol{\alpha_i^{(1)}} & \boldsymbol{\alpha_i^{(2)}} & \boldsymbol{\alpha_i^{(3)}} & \ldots & \boldsymbol{\alpha_i^{(n-r+2)}} \\ \boldsymbol{\alpha_i^{(2)}} & \boldsymbol{\alpha_i^{(3)}} & \boldsymbol{\alpha_i^{(4)}} & \ldots & \boldsymbol{\alpha_i^{(n-r+3)}} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \boldsymbol{\alpha_i^{(r-1)}} & \boldsymbol{\alpha_i^{(r)}} & \boldsymbol{\alpha_i^{(r+1)}} & \ldots & \boldsymbol{\alpha_i^{(n)}} \end{pmatrix}, \tag{13}$$

where r is an integral estimate on the number of entries of the j-th column vector that are sufficient to express the subsequent $(j+1)$-th column in Q_i. The Eigenspace of the above matrix captures the dynamic structure of a system in a meaningful manner [5]. Since, the matrix in Eqn. 13 is not guaranteed to be square, we perform singular value decomposition on $Q_iQ_i^T$, yielding the matrix U containing the eigen vectors. The projection of $Q_iQ_i^T$ on the largest Eigenvector from U is used as the final descriptor, which is interestingly invariant to phase-shift. Thus, a quick pan motion and a slow pan are treated in the same way, as similar to pan-right and pan-left. In practice, all Hankel matrices are normalized before any processing, using the Frobenius norm for matrices given by the following equation:

$$\hat{K}_i = \frac{K_i}{\text{trace}\,(K_iK_i^T)^{\frac{1}{2}}}. \tag{14}$$

The projection results in a $r \times 8 = 64$ dimensional descriptor for a shot We maintain $r = 8$ to have sufficient overlap between column vectors of the matrix. The feature computed as above is used to train linear SVM classifiers, details of which is provided in Section IV. Throughout the paper, this method is being referred as "Proposed" for shot classification.

The descriptor computed using the above method is capable of capturing temporal dynamics across all the Lie group coefficients efficiently for a given shot. To support this argument, we compute a set of exhaustive statistics from each dimension of the 8-dimensional time-series separately. Assuming, a time series across each dimension can be represented as a vector $\mathbf{z}$, the statistical features are as follows: mean, variance (σ), first and last order statistics $(\mathbf{z}_{(1)}, \mathbf{z}_{(n)})$, range $(|\mathbf{z}_{(1)} - \mathbf{z}_{(n)}|)$, average crossing rate $(N(\frac{d(\mathbf{z}-\mathbf{z})}{dt})/N(\mathbf{z})$, dt being temporal interval, $N(.)$ is the cardinality function), average root mean square, mean and variance of skew $(\frac{(\mathbf{z}-\bar{\mathbf{z}})^3}{\sigma^3})$, signal entropy, mean and variance of kurtosis $(\frac{(\mathbf{z}-\mathbf{z})^4}{\sigma^4})$. In addition, we compute 28 pairwise correlations between each of the eight dimensions of the trajectory. Finally the sum and the squared sum of all the dimensions is computed. This results in a total of $8 \times 12 + 28 + 2 = 126$ statistical features and is used as another

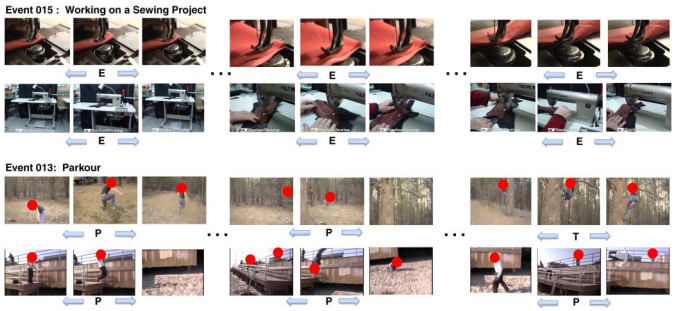


Fig. 5. Camera Motion based Representation of Events: Top two rows represents two different videos each from an event class. Each video is divided into fixed length shots of 100 frames. Outputs from 4 shot classifiers: Establishing (E), Pan (P), Tilt (T), and Zoom (Z) Shot are indicated under each shot.

baseline (SF in Fig. 7) to compare the contribution of the temporal correlation observed between Lie group coefficients.

E. Recognizing Complex Events

The novel camera representation presented above can be used in many applications. In addition to the obvious application of cinematographic shot classification, it can be extended to perform high-level analysis of unconstrained Internet videos. A majority of unproduced videos in the Internet, captured by amateur users tend to have different camera motion signatures depending on the subject or the context of an event being captured. For example, a person "attempting a board trick" or performing "parkour" is mostly captured by a camera in motion to keep the subject in focus. Consequentially, the captured video, depicts significant amount of unintentional pan, tilt or zoom. Similarly, a video of a "parade" is expected to have pan and zoom in contrast to videos that are shot indoors such as "working on a sewing project".

This motivates us to explore a principled approach towards describing such videos using our proposed camera motion descriptor. To this end, we resort to the following methodology: we divide each video into non-overlapping, fixed length shots of f frames and apply pre-trained classifiers for four of the commonly occurring shot classes - pan, tilt, zoom and establishing. Any classifier response below a certain threshold δ is neglected and the corresponding shot class is labeled as "undetermined". Thus, each video can be reduced to string of variable length containing symbols 'P', 'T', 'Z', 'E' and 'U' depending on the respective classifier outputs, as shown in Fig. 5. Given such a compact representation of a video, it is straightforward to train a Hidden Markov Model corresponding to a set of videos that depict a specific complex event category. Once these models are generated, classification for an unknown video, is performed by comparing the log-likelihood of its corresponding camera motion signature string, being generated from a particular model against all other models. The maximum likelihood identifies the target complex event class.

While it is to be noted that unconstrained videos in practice, show a combination of shots: simultaneous pan-tilt, zoom-pan etc., we believe there are two ways to address this problem. The first one is training separate classifiers for such combined shots, which requires further annotation. The second one being using the camera motion representation as is in a bag of features model (Bag-of-LC), without the need of explicit shot classification. Clearly, the first one is beyond the scope of this paper. We however report performance of the second method against the more principled approach of using a HMM based classifier over camera motion signature sequence in Section IV-D.

IV. Experiments

In this section, we first discuss our dataset of 8 distinct category of shot classes based on cinematographic guidelines. The following section provides implementation specific details on the various stages involved in our computational workflow. This is followed by results and discussion. On a separate note, we describe how this shot classification technique can be integrated into large scale complex event recognition, backing our claim with results.

A. Cinematographic Shot Dataset

Most of the earlier papers [9], [21], [25], [32] on this topic evaluate their respective approaches on their own private collections, which are not made available. We make an attempt to build the first dataset of this kind which is reusable, expandable and publicly available.[1] Our dataset consists a clean and an unconstrained part. The clean part has videos downloaded from high resolution, professional stock video[2] while the unconstrained part contains videos from amateur consumer uploaded videos found in YouTube that typically have fair amount jitters caused due to unstable mounts. These two separate sources were used for two different experiments to validate the efficiency of our shot representation. Each videos in the dataset conforms to either one of eight categories, namely: (1) Aerial, (2) Bird eye, (3) Crane, (4) Dolly, (5) Establishing, (6) Pan, (7) Tilt, and (8) Zoom. Each video is carefully screened by 3 human observers with good cinematographic knowledge to ensure there is no mixing up of camera motions in a particular video. Note that this is a difficult task since most shots do not occur in isolation as pointed out in [20]. Finally all videos are resized to an approximate resolution of 480×360 keeping the aspect ratio locked. Some sample frames from the clean part of our dataset are shown in Fig. 6. Table I contains some statistics of our dataset.

B. TRECVID MED 2011 Events Dataset

Recently, NIST has released the Multimedia event detection competition[3] dataset which consists of videos from 15 event categories namely (1) Attempting a board trick, (2) Feeding an animal, (3) Landing a fish, (4) Wedding ceremony, (5) Working on a woodworking project, (6) Birthday party, (7) Changing a vehicle tire (8) Flash mob gathering, (9) Getting a vehicle unstuck, (10) Grooming an animal, (11) Making a sandwich, (12) Parade, (13) Parkour, (14) Repairing an appliance, and (15) Working on a sewing project. We use a subset of this dataset that has 2062 videos from all these 15 event categories for our experiments. Events like "Attempting a board trick" and "Parkour" usually have a lot of jittery camera motion coupled with pan and tilt

[1]https://www.cs.ucf.edu/∼subh/csdv1.tar.gz

[2]http://www.gettyimages.com

[3]http://www.nist.gov/itl/iad/mig/med11.cfm

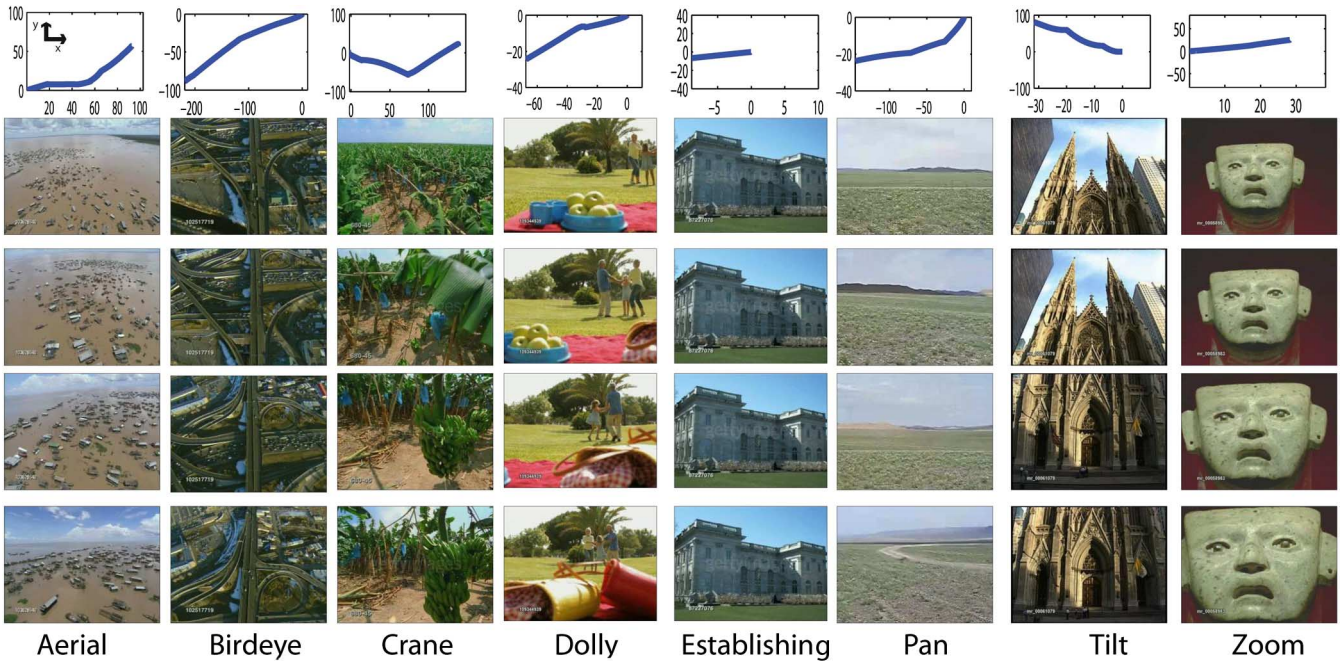


Fig. 6. Cinematographic shot dataset: Each column in the figure represents a typical shot category. The top row shows the trajectory against x and y axes of the image plane (obtained by tracking points). The second row contains the initial frame from the shot. Subsequent rows show samples 50 frames apart. Images from top to bottom provide an idea of the camera motion as the shot progresses.

TABLE I
SOME STATISTICS FROM OUR CINEMATOGRAPHIC SHOT DATASET (UNC. STANDS FOR THE UNCONSTRAINED PART OF THE DATASET)

Shot Category	Examples		Total # of Frames	
	Clean	Unc.	Clean	Unc.
Aerial	30	10	18122	3622
Bird-eye	30	7	18644	1578
Crane	43	8	20304	1226
Dolly	32	8	22241	1185
Establishing	36	9	20454	1256
Pan	30	7	22954	1806
Tilt	31	7	12718	3998
Zoom	31	8	14876	2320

motions. Similarly, videos depicting events such as "Wedding Ceremony" and "Birthday Party" are mostly captured by stationary cameras with limited pan and some amount of zoom. The goal of this experiment is to find out if we can leverage our proposed representation to capture these meaningful statistics from these amateur videos and perform crude event detection without resorting to any content extraction techniques.

Our experiments on this dataset are motivated to substantiate two important claims: Firstly, we are able to demonstrate how our proposed shot representation can be adapted to address a more challenging problem–recognition of complex events. Secondly, it provides an avenue to test our shot classification framework on a significantly large dataset (approx 30,000 shots).

C. Setup

We use an OpenCV based implementation of the SURF [2] extraction and use an approximate nearest neighbor search algorithm [16] to obtain point correspondences which is later required for homography estimation. The normalized homographies (H) and their corresponding Lie-algebra mappings (LC) are used in a bag-of-X framework typically surveyed in [12], under different codebook configurations in the range: $128, 256, \ldots, 2048$ and these help us investigate the efficacy of our shot representation incrementally. In both of these settings, SVMs with histogram intersection kernel is used for classification using a 10 fold cross validation scheme. The parameters for SVM is chosen using coarse grid search during cross validation.

We also, evaluate how our method performs against a more accurate camera trajectory estimation technique (using full structure from motion [24]). We compute similar temporal features as described in Sect. III-D from camera trajectories after connecting the 3-D camera locations (x, y, z) temporally using frame indices. This method is being referred to as TF in the remaining part of the paper. Although features extracted using this method are very discriminative, the trajectory computation in itself a prohibitive task as the 3D reconstruction algorithm needs an exhaustive set of points from all frames in a video to solve a complex optimization problem. This makes this technique a misfit for large-scale Internet videos.

Next, we investigate the discriminability of our final LDS based temporal representation by comparing against naive time-series statistics (referred as SF) as discussed in Sect. III-D. In addition to the above baselines, we compare our method with our implementations of two other relevant algorithms: Motion-Slices [19]) and HF (Threshold selection on Homography and fundamental matrices [31]). The former represents a shot using tensor histogram of spatio-temporal slices of gray-scale intensities while the latter uses a combination of homography and fundamental matrix to represent a shot. It is to be noted that, both of these methods have certain limitations as stated by their respective authors because of which they cannot be applied to all 8 classes of shots in our cinematographic shot dataset.

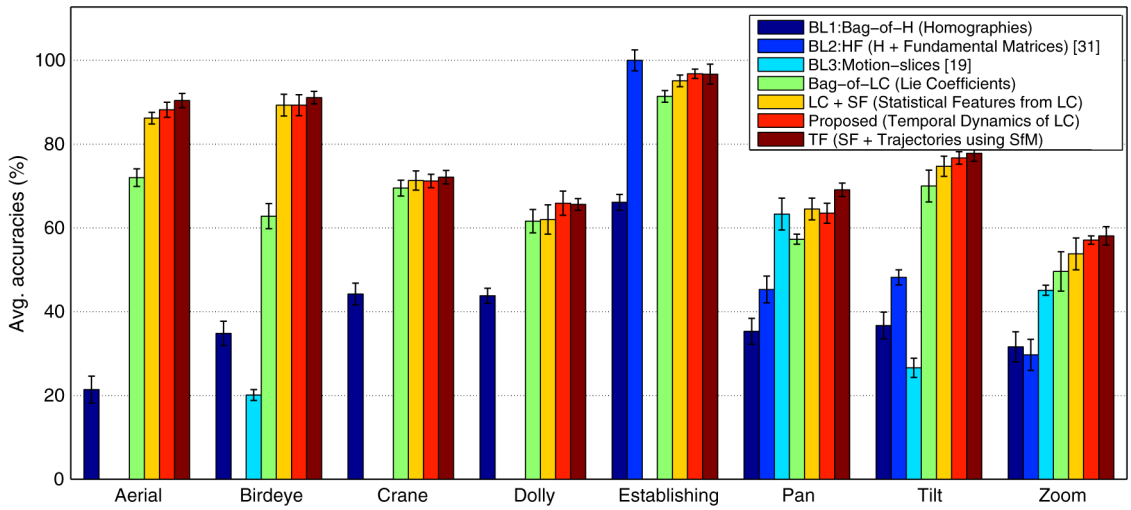


Fig. 7. Classification performance of our method against several baselines on the cinematographic shot dataset. Note that the baseline method [19] cannot be used for 4 types of shot classes: aerial, bird-eye, crane and dolly due to their algorithmic limitations. Similarly the other baseline [31] cannot be applied for crane, dolly and aerial shots (The bars corresponding to these method-shot combinations are non-existent).

Finally, to adapt our shot classification method to recognize complex events, we empirically select $f = 300$ yielding $10s$ long shots. Classifier threshold are uniformly chosen as $\delta = 0.6$ to eliminate less confident responses. We use an open-source implementation of discrete HMMs[4] with default number of states (10). In our experience, changing this parameter does not make any conclusive change in performance. To show, how we fair with the camera motion representation alone, we use a bag-of-features model with a vocabulary size of 256 on the shot-level features to describe each video. We compare both of these approaches with already published bag-of-SIFT [15] features based approaches used in [29], [31]. In both bag-of-X techniques, SVMs with histogram intersection kernels are trained in a 10-fold cross validation mode. Furthermore, to show our shot-level features contain complementary information as compared to the bag-of-SIFT representation, we perform late fusion of event-level classifier scores.

D. Results and Discussions

We begin with an important insight on the motion parameter extraction phase. Through careful temporal sampling of frames for homography estimation the overall classification performance can be improved. Temporal sampling can be perceived as the number of frames that are skipped between any given pair of frames before computing the homography between that pair. Typically, the larger the gap between two sampled frames, the more the homographies deviate from identity as the relative inter-frame motion increases. The average accuracy reaches its peak when the sampling interval is 4, i.e. homography is computed between pairs separated by four frames. This can be explained with the help of evidence from homography computation which is primarily noisy for smaller temporal intervals. At interval lengths larger than 4, the homography violates the primary assumption for Lie group mapping which states that the transformation should be approximately equal to identity.

[4]https://code.google.com/p/pmtk3/

TABLE II

COMPUTATIONAL ASPECTS: EACH ROW INDICATES A COMPUTATIONAL STEP, IMPLEMENTED IN C++/OpenCV. FROM TOP TO BOTTOM: FEATURE EXTRACTION (FE), HOMOGRAPHY ESTIMATION (HE), VECTOR SPACE MAPPING (VSM), TIME SERIES FEATURE COMPUTATION (TSFC). TF BEING 3D CAMERA TRAJECTORIES ESTIMATED DIRECTLY USING STRUCTURE FROM MOTION [24]. THE SPEED IS RECORDED FOR A 320×240 VIDEO CONTAINING 300 FRAMES ON A STANDARD DESKTOP HOSTING A 2.4 GHZ CPU

Step	Speed (in ms)	Size Dependence	Parallelizable
FE	5.3×10^2	Yes	Yes
HE	75	Yes	No
VSM	8	N/A	No
TSFC	4	N/A	No
TF	4.3×10^5	Yes	Partially

In Fig. 7, we provide an extensive analysis of the results we obtained on the cinematographic shot dataset. Each bar in the chart corresponds to one of 6 methods, grouped into 8 classes. As hypothesized in Fig. 4, a naive bag-of-LC (Lie group coefficients of homographies) representation, without any notion of temporal relationship across frames, performs significantly better (21%) than bag-of-H (pure homographies). When we add temporal information through naive statistical features (LC + SF) the performance on an average increases by another (7 - 8%). This is progressively improved by (6 - 7%) when the appropriate method is used to extract meaningful temporal pattern from the sequence of Lie group coefficients.

Furthermore, our proposed shot classification model built on top of LDS based temporal features extracted from sequences of Lie group coefficients does significantly better than all baselines and two of the previously published algorithm. Although our feature does not outperform the structure from motion based trajectory estimation technique (TF), we report comparable accuracies with the obvious advantage of speed. Please refer to Table II for more details.

It is also encouraging to see that the proposed method reports a consistent classification performance (over 75%, variance across different train-test folds are shown in error bars) for 6 out of all 8 categories. Among shot classes, establishing shots are classified with maximum confidence which indicates

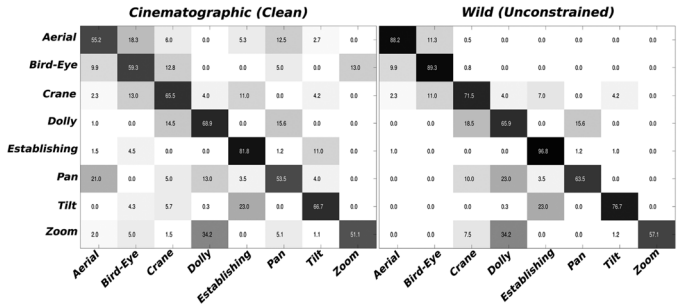


Fig. 8. Confusion matrices obtained after classification on the clean and unconstrained part of the cinematographic shot dataset. Figure on the left shows confusion among classifiers trained on clean shots and tested on the similar clean shots, while on the right, the test set is changed to unconstrained shot, indicating a drop of performance.

strong correlation of performance with proximity of the homography matrices towards the identity matrix. This is followed by the classification performance on bird-eye and aerial shots (over 88%). One of the limitations of our approach is observed in classifying zoom shots where the avg. accuracy is 20% lower than the dataset average. One reason behind such an anomaly can be linked directly to the initial step of extracting homography. In case of zoom shots, the SURF descriptors being sensitive to the degree of scaling, are often mis-matched. This results in degenerate homographies, which may result into suboptimal representations.

A deeper level of understanding can be obtained from the confusion tables listed in Fig. 8. Visually similar shots such as aerial and bird-eye depict certain degree of confusion. Likewise, pan and dolly shots are confused because of similarity when direction of rotation along z-axis coincide with slow translation along the same direction. Apart from the confusion alone, the Fig. 8 offers insight on another interesting experiment we conducted to test the robustness of our shot classifiers. In this experiment, we use our proposed final representation to describe each shot in our dataset. The confusion matrix in the left reports avg. accuracies and the respective confusions across each of the 8 classes in our cinematographic shot dataset. On the right, we report the results when the test set is switched from clean part to the unconstrained part of the dataset. We observe a 16% drop in performance which can be attributed to the nature of the unconstrained shots that contain significant jitter. However, the performance in the constrained part of the dataset is still promising (38% better than random).

Table II shows the typical computational aspect of different steps involved in the entire workflow. In our current implementation, the feature extraction process takes the maximum amount of compute cycles. However, the process is completely parallelizable both in terms of spatial and temporal perspectives as features computed in one frame can be computed independently of previous frames. Except for the feature computation and homography estimation, none of the techniques discussed in our shot classification process are dependent on the spatio-temporal resolution of video. For, brevity we keep aside asymptotic analysis of all the algorithms discussed here.

In the next section we discuss our results on a more challenging problem i.e. recognition of complex events.

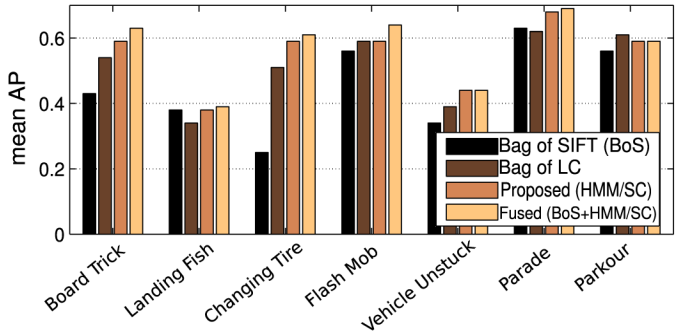


Fig. 9. Recognition of complex events that are expected to occur in outdoor setting with substantial camera motion.

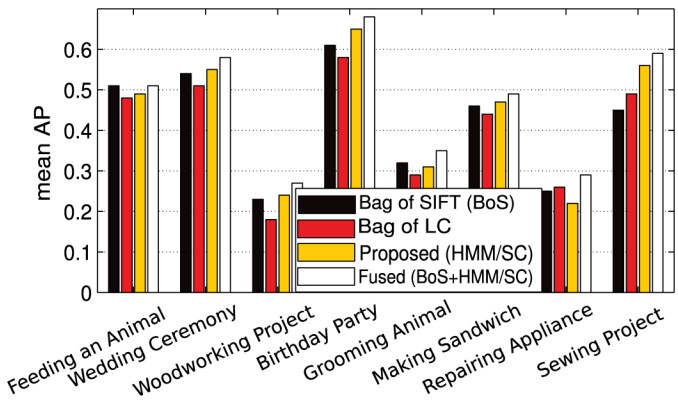


Fig. 10. Recognition of complex events that are expected to occur in indoor settings with relatively low camera motion.

E. Complex Event Recognition Based on Camera Motion

We perform exhaustive comparative analysis of event recognition performance for two separate cases. In the former case, we report the average precision of event classifiers on events that are commonly observed in outdoor settings involving significant camera motion. The results are reported in Fig. 9. The latter case involves events that are typically expected to occur in indoor settings, accompanied by limited camera motion. Fig. 10 reports results corresponding to these events.

Among outdoor events, "Attempting a Board trick", "Changing a Vehicle tire", and "Parade" are well detected using our proposed HMM based approach on top of the predefined shot classifier responses (HMM/SC). While in case of indoor events, "Birthday party" and "Working on a sewing project" are detected with high avg. precision. We also notice that in all event cases, late fusion with a content based classifier $\mathrm{Bag-of-SIFT+SVM}$, improves the result by 3 - 4%, which supplies strong evidence towards the complementary nature of our feature. Interestingly, classifiers trained on Bag-of-LC only achieve 3.5 - 5% lower than HMM/SC. Thus, for even larger datasets, we can obtain a decent trade-off between speed and accuracy by eliminating the full shot classification followed by HMM training step, opting for simpler $\mathrm{Bag-of-LC+SVM}$ based approach.

Lastly, we report detection error trade off plots in Fig. 11(a), specific to the said 5 events to show how graceful the event detectors are at different thresholds. In Fig. 11(b),

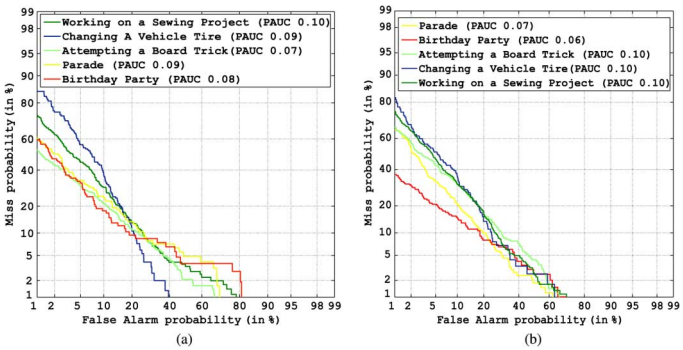


Fig. 11. Detection-Error Trade off (DET) curves for (a) 5 event classes best represented using our camera motion based features. (b) Curves for corresponding event classes obtained using a content based feature representation (Bag-of-SIFT-features).

we compare the performance with detectors based on $\mathrm{Bag-of-SIFT-features} + \mathrm{SVM}$. We fix an reasonable operating region (6% false alarm with 75% misdetection) and measure the area under each curve intersecting this operating region.

V. Conclusion

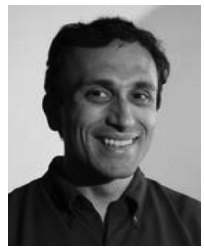
We presented a novel set of methodologies to perform robust shot classification based on camera motion adhering to cinematographic principles. In our approach, we first extracted camera motion from shots by computing frame to frame homographies. In order to represent homographies in a manageable space, we proposed the use of Lie algebra to obtain one to one linear mappings of the homographies. In order to exploit the temporal order these mappings, we compute features from time series constructed from these mappings. Our approach performs significantly better than the state of the art methods. As part of this work, we also introduced a cinematographic shot dataset that can be used by the research community to explore different avenues in this direction. Finally, we demonstrated the applicability of our proposed method to represent ambient camera motion in videos to develop insights towards solving a more challenging event detection problem. As part of future work, we intend to augment our complex event recognition framework with proper camera motion boundary detection [18], instead of these fixed length segments.

References

[1] D. Arijon, *Grammar of the film language*. Los Angeles, CA, USA: Silman-James Press, 1976.
[2] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," in *Proc. ECCV*, 2006, pp. 404–417.
[3] S. Bhattacharya, H. Idrees, I. Saleemi, S. Ali, and M. Shah, "Moving object detection and tracking in forward looking infra-red aerial imagery," *Mach. Vision Beyond Visible Spectr.*, pp. 221–252, 2011.
[4] S. Bhattacharya, B. Nojavanasghari, T. Chen, D. Liu, S.-F. Chang, and M. Shah, "Towards a comprehensive computational model for aesthetic assessment of videos," in *Proc. ACM Multimedia*, 2006, pp. 361–364.
[5] C. DiMonte and K. Arun, "Tracking the frequencies of superimposed time-varying harmonics," in *Proc. ICASSP*, 1990, pp. 2539–2542.
[6] A. D. Doulamis and N. D. Doulamis, "Optimal content-based video decomposition for interactive video navigation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 6, pp. 757–775, 2004.
[7] T. Drummond and R. Cipolla, "Application of Lie algebras to visual servoing," *Int. J. Comput. Vision*, vol. 37, 2000.
[8] H. K. Ekenel, T. Semela, and R. Stiefelhagen, "Content-based video genre classification using multiple cues," in *Proc. Int. Workshop Automated Information Extraction in Media Production*, 2010.
[9] R. Fablet, P. Bouthemy, and P. Perez, "Nonparametric motion characterization using causal probabilistic models for video indexing and retrieval," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 393–407, Apr. 2002.
[10] J. Fan, A. K. Elmagarmid, X. Zhu, W. G. Aref, and L. Wu, "ClassView: Hierarchical video shot classification, indexing, and accessing," *IEEE Trans. Multimedia*, vol. 6, no. 1, pp. 70–86, 2004.
[11] M. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.
[12] Y.-G. Jiang, S. Bhattacharya, S.-F. Chang, and M. Shah, "High-level event recognition in unconstrained videos," *Int. J. Multimedia Inf. Retriev.*, vol. 2, no. 2, pp. 73–101, 2013.
[13] R. Li and R. Chellappa, "Group motion segmentation using a spatio-temporal driving force model," in *Proc. IEEE CVPR*, 2010, pp. 2038–2045.
[14] C. Liu, Q. Huang, S. Jiang, L. Xing, Q. Ye, and W. Gao, "A framework for flexible summarization of racquet sports video using multiple modalities," *Comput. Vision Image Understand.*, vol. 113, no. 3, pp. 415–424, Mar. 2009.
[15] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, pp. 91–110, 2004.
[16] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *Proc. VISAPP*, 2009, pp. 331–340.
[17] J. Nam, M. Alghoniemy, and A. H. Tewfik, "Audio-visual content-based violent scene characterization," in *Proc. IEEE ICIP*, 1998, pp. 353–357.
[18] C.-W. Ngo, T.-C. Pong, and H.-J. Zhang, "Motion-based video representation for scene change detection," *Int. J. Comput. Vision*, vol. 50, no. 2, pp. 127–142, Nov. 2002.
[19] C.-W. Ngo, T.-C. Pong, and H.-J. Zhang, "Motion analysis and segmentation through spatio-temporal slices processing," *IEEE Trans. Image Process.*, vol. 12, pp. 341–355, 2003.
[20] P. Over, T. Ianeva, W. Kraaij, and A. F. Smeaton, "TRECVID 2005—an overview," in *Proc. TRECVID 2005*, 2005.
[21] S.-C. Park, H.-S. Lee, and S.-W. Lee, "Qualitative estimation of camera motion parameters from the linear composition of optical flow," *Pattern Recognit.*, vol. 37, no. 4, pp. 767–779, 2004.
[22] Y. Qi, A. G. Hauptmann, and T. Liu, "Supervised classification for video shot segmentation," in *Proc. ICME*, 2003, pp. 689–692.
[23] Z. Rasheed, Y. Sheikh, and M. Shah, "On the use of computable features for film classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 1, pp. 52–64, 2005.
[24] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: Exploring photo collections in 3D," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 835–846, Jul. 2006.
[25] M. V. Srinivasan, S. Venkatesh, and R. Hosie, "Qualitative estimation of camera motion parameters from video sequences," *Pattern Recognit.*, vol. 30, no. 4, pp. 593–606, 1997.
[26] S. Takagi, S. Hattori, K. Yokoyama, A. Kodate, and H. Tominaga, "Sports video categorizing method using camera motion parameters," in *Proc. ICME*, 2003, pp. II–461–II–464.
[27] H. L. Wang and L.-F. Cheong, "Taxonomy of directing semantics for film shot classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, pp. 1529–1542, 2009.
[28] S. Wang, S. Jiang, Q. Huang, and W. Gao, "Shot classification for action movies based on motion characteristics," in *Proc. IEEE ICIP*, 2008.
[29] Z. Wang, M. Zhao, Y. Song, S. Kumar, and B. Li, "Youtubecat: Learning to categorize wild web videos," in *Proc. IEEE CVPR*, 2010, pp. 879–886.
[30] B. Williams, M. Cummins, J. Neira, P. Newman, I. Reid, and J. Tardos, "An image-to-map loop closing method for monocular SLAM," in *Proc. IEEE/RSJ IROS*, 2008.
[31] Y. Zhai, J. Liu, X. Cao, A. Basharat, A. Hakeem, S. Ali, M. Shah, C. Grana, and R. Cucchiara, "Video understanding and content-based retrieval," in *Proc. TREC Video Retrieval Evaluation Workshop Online (TRECVID)*, 2005.
[32] X. Zhu, X. Xue, J. Fan, and L. Wu, "Qualitative camera motion classification for content-based video indexing," in *Proc. Advances in Multimedia Information Processing*, 2002, pp. 1128–1136.

Subhabrata Bhattacharya is a Postdoctoral Research Scientist at the Digital Video and Multimedia Laboratory at Columbia University in the city of New York. His research in computer vision is in the areas of complex event recognition in web videos, airborne surveillance, and computational video aesthetics. He received the BE degree in Computer Science and Engineering from Burdwan University, India in 2003. He has worked under research roles in IBM and Infosys on high performance computing until 2008. He received his PhD degree in Computer Engineering, from the University of Central Florida in 2013. He was nominated for the best paper award in ACM Multimedia Conference in 2010 and he received the ACM Multimedia grand challenge 2nd Place award in 2013.

Ramin Mehran is a software developer engineer at Microsoft. He received his Ph.D. on computer vision from University of Central Florida. He was previously a graduate research assistant at Center for Research in Computer Vision (formerly Computer Vision Lab) at University of Central Florida and a research intern at Ecole Polytechnique Federale de Lausanne. He received his Master of Science and B.S.E in Electrical Engineering from K.N. Toosi University of Technology in Iran with majors in Control Systems and Telecommunications. His current work is focused on computer vision and data mining.



Rahul Sukthankar is a scientist at Google Research, an adjunct research professor in the Robotics Institute at Carnegie Mellon and courtesy faculty in EECS at the University of Central Florida. He was previously a senior principal researcher at Intel Labs, a senior researcher at HP/Compaq Labs and research scientist at Just Research. He received his Ph.D. in Robotics from Carnegie Mellon and his B.S.E. in Computer Science from Princeton. His current research focuses on computer vision and machine learning, particularly in the areas of object recognition, video understanding and information retrieval.

Mubarak Shah is the Trustee Chair Professor of Computer Science, and is also the founding director of the Center for Research in Computer Vision at UCF. His research interests include: video surveillance, visual tracking, human activity recognition, visual analysis of crowded scenes, video registration, UAV video analysis, etc. Dr. Shah is a fellow of IEEE, AAAS, IAPR and SPIE. In 2006, he was awarded a Pegasus Professor award, the highest award at UCF. He is ACM distinguished speaker. He was an IEEE Distinguished Visitor speaker for 1997-2000 and received IEEE Outstanding Engineering Educator Award in 1997. He received the Harris Corporation's Engineering Achievement Award in 1999, the TOKTEN awards from UNDP in 1995, 1997, and 2000; Teaching Incentive Program award in 1995 and 2003, Research Incentive Award in 2003 and 2009, Millionaires' Club awards in 2005 and 2006, University Distinguished Researcher award in 2007, honorable mention for the ICCV 2005 Where Am I? Challenge Problem, and was nominated for the best paper award in ACM Multimedia Conference in 2005. He is an editor of international book series on Video Computing; editor in chief of Machine Vision and Applications journal, and an associate editor of ACM Computing Surveys journal. He was an associate editor of the IEEE Transactions on PAMI, and a guest editor of the special issue of International Journal of Computer Vision on Video Computing.