International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 14 (2022) pp. 270-284
© MIR Labs, www.mirlabs.net/ijcisim/index.html

Submitted: 08 September, 2021; Accepted:12 February, 2022; Published: 11 April, 2022

SALiEnSeA: Spatial Action Localization and Temporal Attention for Video Event Recognition

Prithwish Jana[1], Swarnabja Bhaumik[2] and Partha Pratim Mohanta[3]

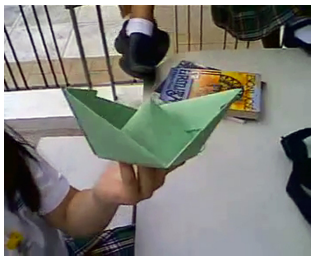
[1]Department of Computer Science and Engineering,
Indian Institute of Technology Kharagpur, India
pjana@ieee.org

[2]Deloitte USI
swarnabjazq22@gmail.com

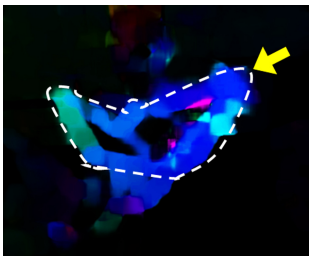
[3]Electronics and Communication Sciences Unit,
Indian Statistical Institute, Kolkata, India
partha.p.mohanta@gmail.com

Abstract: **Automated event and activity recognition in unconstrained videos has become a societal necessity. In this paper, we address video event classification and analyze the influence of preprocessing through action localization on the classification task. We propose an approach for event classification in videos, that is aided by unsupervised preprocessing through temporal attention and subsequent spatial action-localization at those specific attentive instants of time. The unsupervised temporal attention is achieved through a graph-based algorithm for selection of representative (key) frames. Our spatial action localization technique *SALiEnSeA* identifies the most-'dynamic' motion patch in each key-frame. It is based on an oil-painting approach of refining and stacking motion components. These focused actions along with spatial and temporal information are fed into three separate deep neural-network pipelines consisting of ResNet50 and LSTM. A multi-tier hierarchical fusion thereby, consolidates frame-level and video-level predictions. The experiment is performed on four benchmark datasets: CCV, KCV, UCF-101 and HMDB-51. The holistically developed solution framework for action localization-aided event classification provides encouraging results. By introducing a separate modality for action-localized *SALiEnSeA* patches, we get improved video classification performance on top of the traditional modality of RGB frames. This outperforms standard neural-network based approaches as well as state-of-the-art multimodal models in use, for video classification.**

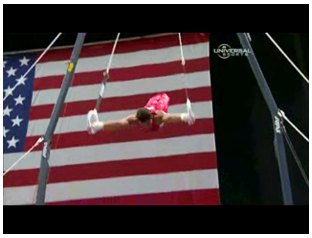
Keywords: Video Classification, Event and Activity Recognition, Unsupervised Action Localization, Motion and Video Analysis, Deep Neural Network



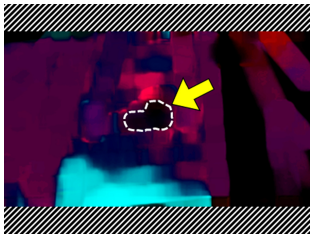
(a) Event "Boat" in KCV [1]



(b) Optical-Flow after HSV transform (*Ref.* Section III-B)



(c) Event "Still Rings" in UCF-101 [2]



(d) Optical-Flow after HSV transform (*Ref.* Section III-B)

Figure. 1: Understanding (a)-(b) the pros and (c)-(d) some of the common challenges (cons) involved in relying upon the motion-heavy regions in optical-flow matrix

I. Introduction

The past two decades saw the rise of social media, and the number of active users currently accounts for almost 50% of the world population. Moreover, the number of social media users is increasing by about 9% each year. Consequently, many videos are being uploaded to these content sharing social-media platforms. Be it for regulating inappropriate content, or annotating content for a personalized search experience, an automatic visual event recognition system is always called for in today's era of Internet dominance and automated surveillance. Since the advent of *computer vision*, active research in visual scene understanding has helped the subject reach its present form. In recent times, its different applications are being used in almost all the sectors of everyday life – from the basic necessities to the technological advancements.

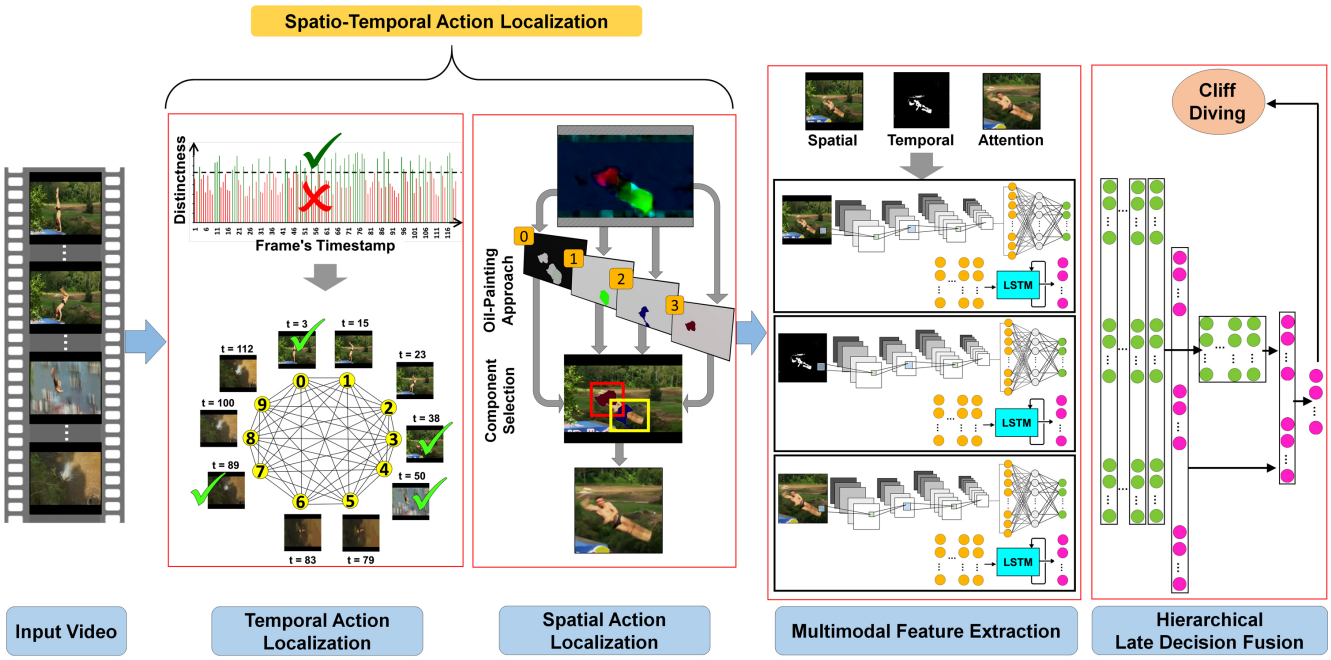


Figure. 2: Illustration of the major steps involved in the proposed method.

Events/Activities. Typically, an event includes person(s) and/or object(s) and refers to their mutual behavior. Events cover small-scale as well as large-scale activities. *Small-scale activities* [3][4] include movement of minor body parts, as for example movement of fingers (playing piano, typing, knitting) and facial gestures (smiling, crying). *Large-scale activities* [5][6], on the other hand, may involve full-body movement (gymnastics, jumping), locomotion (walking), etc. This paper aims to identify large-scale and non-intricate small-scale activities, and thereby understand the high-level ongoing event in videos.

Spatio-Temporal Attention. In the recent past, researchers have come up with promising activity-recognition results by feeding deep networks with bulks of data, as for instance, 3D-ConvNet [7] using untrimmed video and combination of spatio-temporal modalities using whole key-frames [8]. But this gives rise to a concern, whether surplus information actually assist the network or, end up putting it in dilemma with contradicting data. Real-life unconstrained videos typically exhibit high intra-class variance. This is exemplified in Figure 1(a) where acknowledging the fact that "boat"s are commonly surrounded by water in most videos, this video of paper-boat is an exception in itself. To exploit a deep architecture to the best of efforts, such kind of variation must be minimized so that videos of the same class appear least disparate [9]. Only then, a deep neural net would be unaffected by varied backgrounds. Thus, a close-up attention image would be beneficial, as denoted by yellow arrow in Figure 1(b). Hence, event recognition cannot be regarded as a mere classification task. This issue is addressed in the proposed work by performing both *temporal* (time-wise) and *spatial* (space-wise) *action localization*. The former pin-points most action-stuffed key-frames and the latter identifies and feeds those frames' dominant subject to the deep neural architecture.

Challenges. High-end cameras like an event camera [10] or intelligent thermal camera [11], possess superior vision sensors that can asynchronously assess intensity changes. For such a fixed camera, attention-detection is a relatively straightforward task because all the background pixels are static and the moving pixels corresponds to the subject of the video. However, in unconstrained videos captured by amateurs on regular cameras, majority of the pixels are in motion and it leads to number of false-positives. Thus, simple extraction of the most motion-heavy patches would not only lead to less accurate results, but the frame's subject may also be excluded. This is exemplified in Figure 1(c), where the frame exhibits slanting-line patterns throughout the background along with presence of top-bottom margins. The area where lines meet the margin (high intensity sky-blue patch in Figure 1(d)) gives a false portrayal of motion (*Ref. Scissor-effect* in Section III-B). Here, the subject is relatively static (denoted by yellow arrow in Figure 1(d)) amidst the jittery background. Evidently, motion-heavy patches do not always correspond to the frame's subject. Motivated by the fact, in this paper we have proposed an action localization scheme that identifies the attention patches in frames of unconstrained video. The efficacy of the proposed scheme is demonstrated through examples in Figure 6.

Contributions. In this paper, we put forward an efficient video event recognition approach that is based on action localization. The proposed action localization preprocessing focuses on the most semantically salient space-time portions of videos to classify an event. We introduce an unsupervised, light-weight spatial action localization technique, that pin-points the subject in a video key-frame, full of non-contextual background information. The frame-subject is represented by the salient and most-'dynamic' motion patch identified in each key-frame. Thus, the focus is on feeding ConvNets with more salient, pin-pointed action information.

Later in Section IV, we show that this extension enables any existing set of modalities on an existing deep neural architecture, to provide better predictions to a wide range of unconstrained videos. Effectively, we embark on the holistic classification process without essentially occurring for all the constituent video frames and all the spatial locations in the selected key-frames. The action localization scheme makes the result robust to both wide-shot and close-shot videos because in both cases the frame's subject is cropped out. Also since this is unsupervised, there is no necessity of large labeled datasets and long training phases. This has increased the practical applicability of our method on real-life video clips. Further, the proposed late decision-fusion technique leads us to a judicious consensus of the frame- and video-level predictions with due degree of importance on the individual classification pipelines, corresponding to each modality. The steps involved in the proposed method is shown in Figure 2.

The paper is organized into five sections. Following the introduction, some of the relevant past works are presented in Section II. Next, in Section III we elaborate our proposed methodology. The experimental results are discussed in Section IV. Finally, Section V draws the epilogue alongside providing some scopes of further improvements.

II. Related Work

The traditional event recognition methods are mainly based on extracting the visual cues from a sparse or dense labeling of frames [12]. With progress of time, researchers started fusing multimodal information, viz. spatial, temporal and acoustic data, for video classification. Such approaches include the conventional way of deploying separate convolutional networks to extract features from each modality and thereafter, using end-to-end fusion networks or aggregation of probability distributions, like that of Li et al. [13]. Moreover, some researchers like Jiang et al. [14] preferred to bring in more information about feature and inter-class relationships to find out similarities among the semantic categories. Although these induced a huge leap in classification performance, but still there was a major room for improvement.

It was observed that deep networks performed much better when they were given an information about the precise spatio-temporal location of an ongoing action. By virtue of *temporal action-localization*, the 'essential' temporal segments are fed to a classifier network which correspondingly categorizes each such clip [15]. Pei et al. [16] introduced a recurrent attention-gated network to interpret the physical saliency of each time-step from video sequences. Aote et al. [17] used a key-frame extraction approach based on fuzzy c-means clustering, and utilized saliency map and color histogram of these key-frames to annotate videos. Some researchers prefer preserving all the temporal data and rather than temporal attention, perform *spatial action-localization*. Karpathy et al. [18] was instrumental in introducing multi-resolution CNN that fed on foveated center-crop attention images, and this was unique in its own way for video classification task. But unsurprisingly, this fails in unconstrained videos where the object of interest remains missed out, as they rarely occupy the central patch of frames (e.g. Figure 6(a), 6(k)). Here, object detection can assist the classification process [19]. For fine-grained discrimination between spatio-temporally similar events like birthday party and anniversary ceremony, researchers sometimes prefer to perform frame-wise object detection in videos [20]. As for example, Burić et al. [21] perform object detection per frame by Mask Region-based CNN (R-CNN) for improved recognition of activities in sports videos. Gkioxari et al. [22] had put forward a modification of the R-CNN that considers multiple region proposals to identify a primary person-focused region and a secondary context region. However, being too much human-centric, it does not offer best results in videos involving inanimate objects, animals or minor human-body parts (e.g. Figure 1(a), 6(d), 6(f)). In fact, as pointed out by Pacheco et al. [23], these pre-trained object detector-based approaches can even fail in specific tasks e.g. action classification in videos of infants, because they were trained primarily on datasets comprising adult and commonly occurring objects.

Integrating spatial and temporal action-localization gives an even more accurate space-time blob, and this rules out all redundant/irrelevant data. Li et al. [24] uses a RNN-based spatial attention framework that takes three information into account, viz. the visual content, the accumulated attention from temporally-correlated past frames and the current state of an ongoing action. Peng et al. [25] use two independent spatial (static) and temporal (motion) attention networks to respectively extract discerning static and motion features. These are learned collaboratively by a static-motion model that exploits the interdependence between them and thereby, classifies a video. Li et al. [26] came up with Spatio-Temporal Attention Networks that exploit attention, both at the spatial and temporal level. Effectively, the spatial attention on frame/optical-flow (modality) are learnt by an *AttCell* and fed through a CNN, before being concatenated and utilized in LSTM that structures the inter-modality temporal attention. Liu et al. [6] propose a two-stream ConvNet, a temporal attention module (TAM) to identify key-frames, and a spatial attention module (SAM) to extract the action-relevant areas within a frame. Although their results mostly tally with common human perception, the SAM results are prone to noises and in slow-changing videos, TAM is occasionally susceptible to choosing near-similar key-frames.

Fusion of multimodal information is adopted nowadays to improve image and video classification performance. Adhikari et al. [27] have performed a study of bird classification from images and vocal notes using audio-visual features and multimodal deep CNNs. To integrate features extracted from multiple streams in videos, *early fusion* [28], *kernel-level fusion* [13] and *late decision fusion* [18, 29] are the three major types of fusion performed. Researchers like Ng et al. [30] had preferred to use a combination of fusion techniques to exploit their individual merits simultaneously. For the fusion part, we have improved the conflation strategy put forward in one of our earlier works [8]. Spatio-temporal action localization already reduces extra unnecessary information from our CNN inputs. The effectiveness of this late decision fusion strategy is studied in Section IV, where fusion is performed on predictions from CNN and LSTM on each individual modality, and different combinations of these.

III. Proposed Approach

The goal of our proposed approach is efficiently to recognize the 'event' pertaining to a video, using action-focused input information to a deep neural architecture.

A. Key-Frame Representation of Video

This stage serves the purpose of *temporal localization* of actions. In this step, a video of N frames is symbolized by a pre-fixed number n, of representative key-frames $\{f_k\}$, such that $n << N$. Taking cues from the method of Jana et al. [31], two major steps are followed in this regard.

The *first step* involves elimination of those frames which exhibit temporally redundant information. For this, inter-frame motion between consecutive frames is estimated by dense optical-flow algorithms like Lucas-Kanade tracker [32] or Farneback's dense optical flow estimator [33]. Modern optical-flow techniques [34][35] following a supervised approach by incorporating fast semantic segmentation [36] can also be used, but it may bring in extra computational needs. Next, 'motion' is quantified by generating a *motion histogram* of magnitude and slope of the corresponding flow vectors. Higher ℓ_1-norm amongst such histograms of consecutive frames indicates a higher distinctness and lower chance of information redundancy on the time axis. Subsequently, only a handful of frames are preserved that satisfy a pre-calculated minimum criteria. Typically, this minimum criteria can be considered as the 75^{th} percentile mark in a box-plot representation. This was taken in accordance with the definition of inter-quartile range [37], a popular measure of spread, whose upper-boundary is defined by this mark in box-plot. As an effect of this step, the search space for key-frames is reduced drastically and temporal redundancy is mostly eliminated.

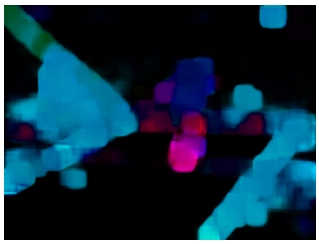
The *second step* involves choosing a set of most temporally distinct frames, alongside maintaining a lower limit of difference in timestamps. For this, frames are denoted by nodes of a complete graph whose edge-weights signify ℓ_1-norm amongst motion histograms corresponding to its terminal node. Also, a minimum acceptable timestamp difference (t_{min}) is maintained amongst the *selected* key-frames. The value of t_{min} can be represented as $\left\lfloor \frac{N}{r \times n - 1} \right\rfloor$. It is the strictest when relaxation factor r is unity and sampled frames are separated no closer than an n-sampling of equally-spaced key-frames from the video of N frames. For our case, we fix r at 2. Thereafter, an iterative edge-selection algorithm is followed. In each iteration of this algorithm, 'viable' edges are defined as those which satisfy the minimum timestamp difference among both the terminal nodes and among each of the terminal and all the selected nodes. The highest-weighted edge is selected from these viable edges, resulting in a pair of terminal nodes (key-frames) in each iteration. This iterative process continues until there remains no 'viable' edges or the predefined number (n) of key-frames are selected, whichever occurs first.

Since this key-frame extraction algorithm exploits the inter-frame temporal variation and maximizes their distinctness, we are sure of starting off with the most action-packed temporal moments in a video.

B. SALiEnSeA: *Spatial Action Localization in Engendering Semantic Attention*

Many social media videos contain black margins around them. It may be observed that, such margins are mostly line-symmetric across the horizontal and vertical central axes. We start off our spatial action localization algorithm by discarding these symmetric near-black margins from each key-frame f_k (as in Figure 3(a)) that do not manifest any signs of motion. The challenging part is that, due to successive uploads and downloads or occasional jittery noises, these black borders are not pitch-black and thus, their RGB values deviate from $(0, 0, 0)$. This was tackled by converting f_k to grayscale and binarizing using fuzzy c-means based clustering [38]. After this pixel-wise thresholding, same-height horizontal chunks and same-width vertical chunks containing black pixels only, were removed from the top-bottom margins and the left-right margins, respectively. This step, not only reduces our subsequent search space, but also helps to identify pseudo-motion patches, like the high-intensity blue patch in Figure 1(d). This will be elaborated in the subsequent paragraphs while discussing *Scissor-effect*.

(a) Frame obtained after cropping



(b) OF

(c) OF_q with 4 clusters

Figure. 3: (Best viewed in color) (a) Cropping near-black borders from frame, f_k depicted within red corners (b) Corresponding optical flow represented as HSV image (c) Color-quantized optical flow image

Next, for each such frame f_k and its immediate next frame, the dense optical-flow [33] is calculated. As a result, a sense of motion is associated with each pixel (x, y) by a flow-vector $\overrightarrow{v_{(x,y)}}$ representation. In order to put this information more pictorially, we transform $\overrightarrow{v_{(x,y)}}$ to polar coordinates and represent the dense-optical flow as an HSV image (OF). The Hue (H) channel stores the direction of $\overrightarrow{v_{(x,y)}}$, and the Saturation (S) channel stores its magnitude. This is illustrated in Figure 3(b), where the high-intensity patches denote high motion and the similar colors signify motion in the same direction. This HSV image is quantized by applying k-means clustering, where each data point is represented by three attributes, corresponding to each of the three color channels. OF_q is the resulting quantized image thus

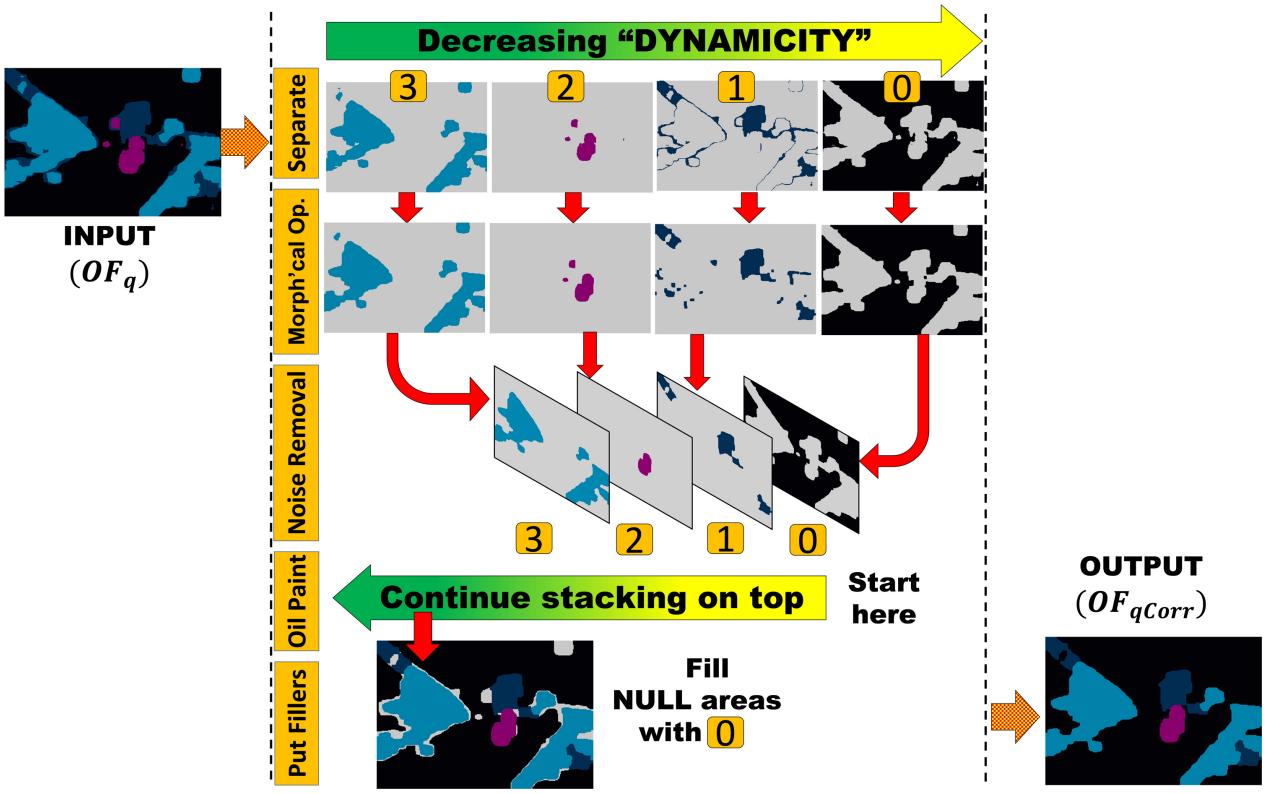


Figure. 4: Phases in the oil-painting approach of converting OF_q to OF_{qCorr}

obtained, after color space clustering. This is illustrated in Figure 3(c), where pixels belonging to the same cluster imply that they move in the same direction with similar pace.

Oil-Painting Approach towards Refining $\mathbf{OF_q}$. For each cluster-label lbl in OF_q, a separate binary image (I_{lbl}) is formed such that,

$$I_{lbl}(x,y) = \begin{cases} lbl, & OF_q(x,y) = lbl \\ 0, & elsewise \end{cases} \tag{1}$$

A sequence of morphological operations is performed on I_{lbl}. For these operations, we define pixel neighborhood by a $W \times W$ square structuring element (S). Morphological erosion [39] of a binary image B, removes minor details, that are smaller than S, and shrinks its larger components from their outer boundaries. It is defined as $B_{eroded} = B \ominus S$ such that,

$$B_{eroded}(x,y) = (B \ominus S)(x,y)$$
$$= \begin{cases} lbl, & \forall_{i,j \in [-\lfloor \frac{W}{2} \rfloor, \lfloor \frac{W}{2} \rfloor]} B(x+i, y+j) \times S(i,j) = lbl \\ 0, & elsewise \end{cases} \tag{2}$$

On the contrary, morphological dilation [39] of a binary image B, fills up gaps and holes, that are smaller than S. As such, this operation in turn expands each of the connected components from their outer boundaries. The equation concerning morphological dilation of a binary image B, can be formulated as $B_{dilated} = B \oplus S$ such that,

$$B_{dilated}(x,y) = (B \oplus S)(x,y)$$
$$= \begin{cases} lbl, & \exists_{i,j \in [-\lfloor \frac{W}{2} \rfloor, \lfloor \frac{W}{2} \rfloor]} B(x+i, y+j) \times S(i,j) = lbl \\ 0, & elsewise \end{cases} \tag{3}$$

Firstly, we perform morphological closing (i.e. dilation followed by erosion) of I_{lbl} to fill up small holes in the image. Subsequently we perform morphological opening (i.e. erosion followed by dilation) to eliminate small components and narrow connections from the opened image. Since frame-subject covers substantial spatial area, a noise-removal step is essential here, whereby we eliminate all small-sized outlier components.

Finally, we assign a *dynamicity* value to each of the cluster-labels, that we had obtained previously. It is evident that the cluster-label for which corresponding pixels has the highest average saturation value (in OF), accounts for the *most-dynamic* set of pixels in a frame. By the term '*most-dynamic*', we imply that they have the highest amount of motion associated with them. We go on stacking all the modified I_{lbl}-s, starting off with the *least-dynamic* label at the bottom and ending with the *most-dynamic* one on the top. This resonates with the philosophy of oil-painting, where the canvas is started off with a coat of underpainting, and finished with a coat of overpainting. Here also, the *least-dynamic* label serves the purpose of underpainting, or the stationary background. And the final 'coat' i.e. the *most-dynamic* label, functions as the overpainting corresponding to the detailing and frame's subject.

As a post-processing step, pixels with still no label-assignments are assigned value of the *least-dynamic* label.

The corrected and color-quantized optical flow image, thus obtained, is OF_{qCorr}. The entire process is pictorially explained in Figure 4. After this step, all individual connected-components $\{CC_{id}\}$ are large enough to independently represent a major area in the image.

Identification of Meaningful Component(s) from $\mathbf{OF_{qCorr}}$. With all connected components large enough, we label each pixel in OF_{qCorr} by a pair, (lbl, CC_{id}). Thereby, each pixel belongs to one of these Labeled-Connected-Components (LCCs), thus obtained. For every LCC, the corresponding average saturation value (from OF) is calculated, that measures the amount of motion (*dynamicity*) associated with them.
It is understandable that a frame's subject would have some motion associated with it, that is visually disparate from the motional behavior of its surroundings. But a simple identification of the most *dynamic* LCCs would most certainly be erroneous towards subject localization. Misleading LCCs almost always touch the frame's exterior perimeters. The succeeding points substantiate the reason behind this special location at which this discrepancy occurs:

- Optical Flow algorithms assumes *Spatial Coherence*, i.e. pixels in a neighborhood should exhibit similar motion. But this is contradicted at the frame's near-black margin strips, where two set of pixels, with contrasting motional behaviors, meet.
- When a scissor is being closed and the angle between its blades is minimal, then the notch of intersection acquires a high velocity [40]. A similar *Scissor-effect* is noticed at the intersection of edges and the frame's margin strips, especially when the edges are almost parallel to the latter. This creates an illusion that pixels near the frame boundaries are moving at a fast pace, as in Figure 1(c)-1(d).

We propose to exclude all the *border LCC*s, i.e. LCCs that touch any of the frame boundaries. This is *beneficial* because *border LCC*s cannot be relied upon as they exhibit pseudo-motions, and is *benign* because in a key-frame, the subject has the least possibility to not occupy a central location.
Subsequently, we start with the *most-dynamic* non-border component (LCC_1), and consider the bounding-box ($BBox_1$) around it. If $BBox_1$ satisfies our preset criteria of covering a minimum substantial area ($area_{min}$) of the image, then LCC_1 becomes our sole chosen component `[CASE: 1-LCC]`. But if LCC_1 is not big enough, we make a decision as to whether to include the second-most dynamic non-border component (LCC_2), or not. For this, we expand $BBox_1$ uniformly from all the sides, so that it occupies the minimum area. Considering $BBox_1$ to be of dimension $h \times w$, the length (d) that is to be incremented on all sides of $BBox_1$, is obtained by solving $(h + 2d) \times (w + 2d) \geq area_{min}$. The minimum value of d, thus obtained, is

$$d = \frac{-(h+w) + \sqrt{(h-w)^2 + 4 \times area_{min}}}{4} \tag{4}$$

Next, we compare the bounding-box around LCC_2 (i.e. $BBox_2$) and the expanded $BBox_1$. When both these boxes overlap, there is a high possibility that LCC_1 and LCC_2 together may represent something more semantically meaningful. They are close-by, and may have been separated due to the jittery video quality and faults in optical-flow algorithms. Thus, we choose both LCC_1 and LCC_2 `[CASE: 2-LCC]`.

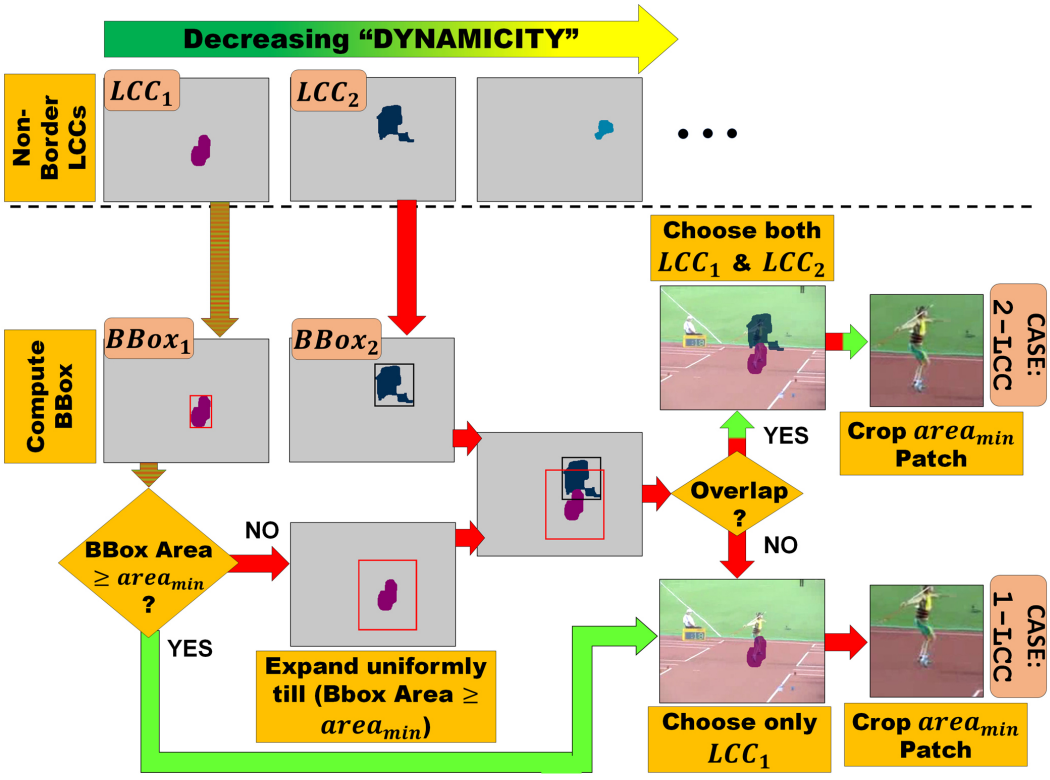
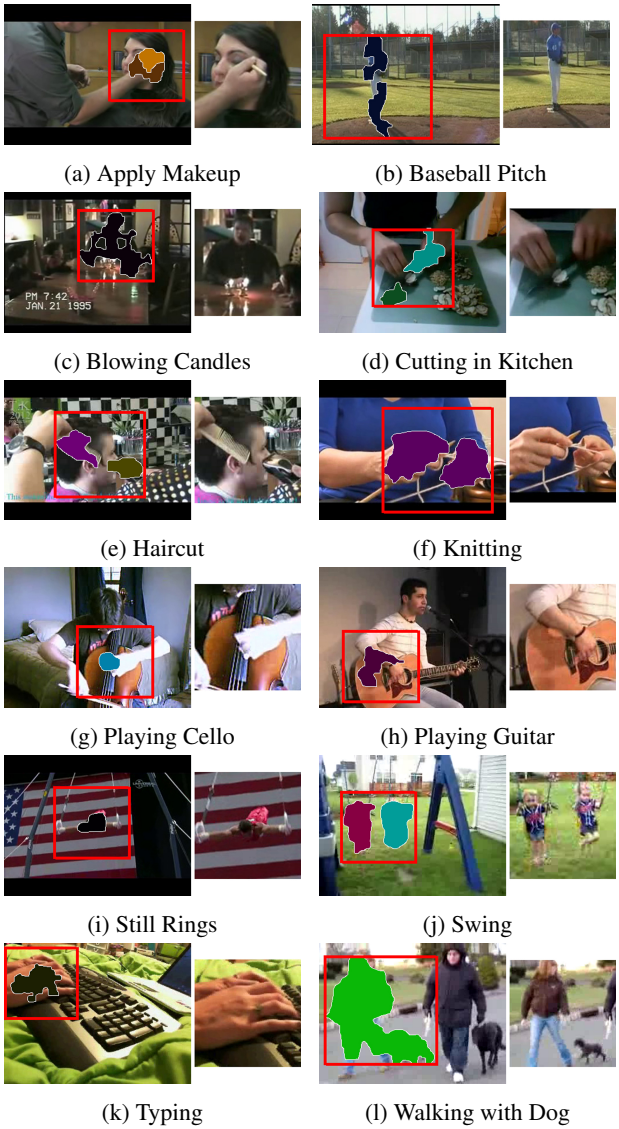


Figure. 5: Process of pin-pointing area-of-interest in f_k, using OF_{qCorr}.

But if they do not overlap, it signifies both these components are uncorrelated. In that case, we choose only LCC_1 `[CASE: 1-LCC]`. The sequence of steps are elucidated in Figure 5. As a final step, the bounding-box is made to assume shape of a square by incrementing the height or width, whichever is lower. If the area covered is short of $area_{min}$, the box is incremented from all sides till the minimum area is achieved. We crop the square patch corresponding to this box from the original RGB image, to obtain our attention patch.



(a) Apply Makeup (b) Baseball Pitch

(c) Blowing Candles (d) Cutting in Kitchen

(e) Haircut (f) Knitting

(g) Playing Cello (h) Playing Guitar

(i) Still Rings (j) Swing

(k) Typing (l) Walking with Dog

Figure. 6: (a-l) Spatial localization of square-shaped attention patches (red boxes).

It is observable that even if the subject is not at the center of a frame (e.g. Figure 6(a), 6(g), 6(h), 6(k)), the proposed method is able to efficiently localize actions in the frame. Even spatially separated objects that are close-by, but are semantically related w.r.t motion are identified jointly in the same attention patch (e.g. Figure 6(d), 6(e), 6(j), 6(l)).

C. *Overall Deep Architecture and Fusion of Classification Results*

In this section, we propose a deep architecture and a late decision fusion strategy to efficiently process the multimodal information obtained from the video.

Deep Neural Architecture. Our deep learning framework for classification of unconstrained videos into various social events, human activities, etc. consists of ResNet50 [41] (a CNN) followed by LSTM [42]. Three different kinds of information, viz. *Spatial*, *Temporal* and *Attention*, are separately exploited through this deep learning framework. This overall hybrid framework is depicted in Figure 7. The spatial wing inputs RGB images corresponding to the key-frames, generated as a result of temporal action localization. The temporal wing, in turn, inputs matrices formed from magnitude of dense optical flows, corresponding to the key-frames and its immediate next frame. The attention wing inputs square-shaped patches corresponding to action-packed regions in key-frames i.e. the outcomes of spatial action localization by SALiEnSeA on key-frames.

Multimodality Fusion. The countable set of all possible events (e.g. diving, kick-ball, draw-sword, etc.) constitutes the collection of all *outcomes* (S). The CNN decisions from each key-frame can be regarded as *frame-wise predictions* (FP). LSTM accumulates features for each frame and gives a *video-wise prediction* (VP). This paper focuses on a late decision fusion approach [8] that takes into account all such FPs pertaining to each key-frame and VP corresponding to the whole video, from each of the modalities involved.
Let's suppose that M is the set of modalities and F is the set of key-frames. We want to consolidate the collection of different probability mass functions (PMF),

$$\{P_{m,f}\}_{\forall m\in M,\forall f\in F} \cup \{P_{m,F}\}_{\forall m\in M} \tag{5}$$

all defined on the same set of possible outcomes, S. Here, $P_{m,f}$ represents a FP proffered by m-stream of CNN for frame f and $P_{m,F}$ represents a VP proffered by m-stream of LSTM. Two PMFs are said to belong to the same homogeneous sub-group when all of the following conditions are satisfied.

- Either both are FP or both are VP
- If both are FP, they should either correspond to the same frame (different modalities) or to the same modality (different frames of same video)
- If both are VP, both should correspond to the same video but different modalities

Considering $\mathscr{P}$ to be such a homogeneous sub-group of probability distributions, the outcome depends on the joint probability of all $P \in \mathscr{P}$. This behavior is accurately captured when we consider the conflation [43] of all $P \in \mathscr{P}$, which is proportional to the product of the corresponding probability values and is obtained by,

$$_{a\in\mathrm{S}}P(X=a) = \frac{\prod\limits_{P\in\mathscr{P}} P(X=a)}{\sum\limits_{c\in\mathrm{S}}\left(\prod\limits_{P\in\mathscr{P}} P(X=c)\right)} \tag{6}$$

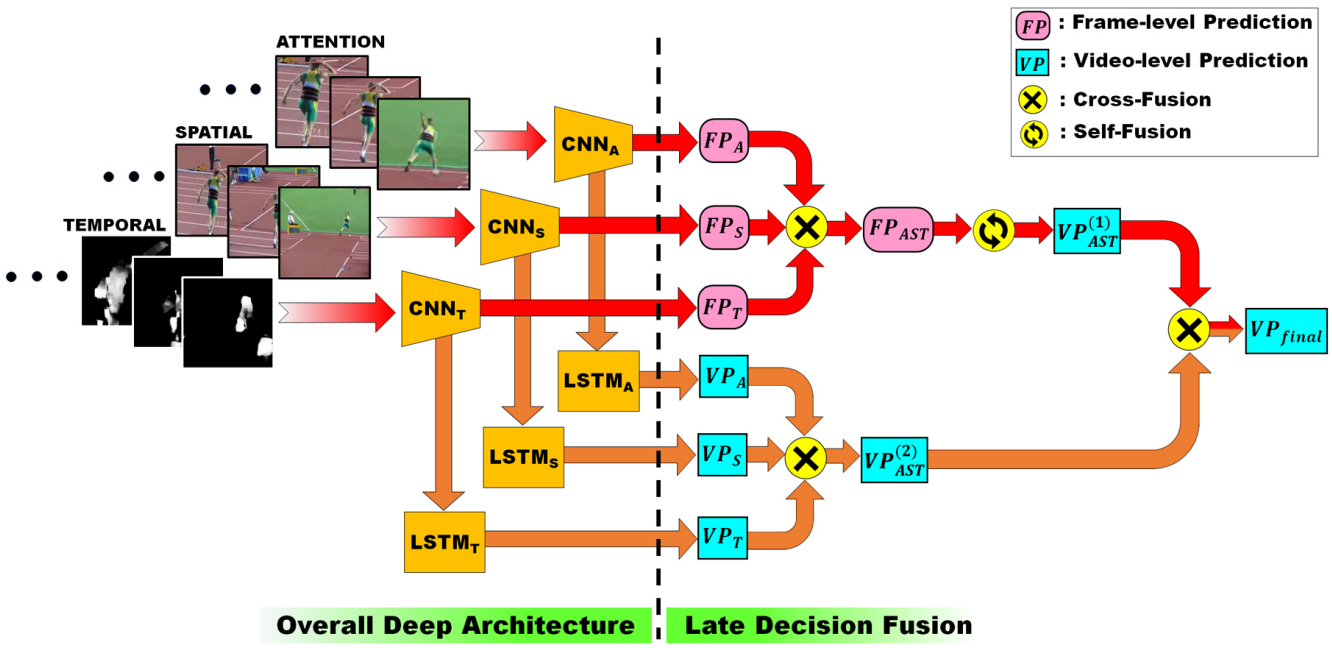


Figure. 7: The deep architecture and decision-fusion strategy used to integrate frame- and video-level predictions. A, S and T respectively denote Attention, Spatial and Temporal

We further define two different kinds of fusion operations: *cross-fusion* and *self-fusion*. The first operation, Cross-Fusion is denoted by the function

$$\prod_{f \text{ or } F}^{'} : \{P_{m,\, f \text{ or } F}\}_{\forall m \in M} \rightarrow P_{M,\, f \text{ or } F}$$

and is defined as the biased-conflation [8] of FPs (or VPs), corresponding to the same frame f (or same video F) but belonging to different modalities. The output is a PMF corresponding to the same key-frame (or video) representing a consolidation of all the modalities. Self-Fusion is denoted by the function

$$\sum_{m \text{ or } M}^{'} : \{P_{m \text{ or } M, f}\}_{\forall f \in F} \rightarrow P_{m \text{ or } M, F}$$

and defined as the biased-conflation [8] of the FPs, corresponding to all the key-frames in a video, each representing a consolidation of same modality m or set of modalities M. The outcome is a VP representing a consolidated PMF of all the key-frames in a video.

Finally, there should be a hierarchical order in which these fusions should be performed so as to bring about a meaningful consolidation at each step. The operations involved in the CNN and LSTM pipelines and their consolidation, can be expressed as:

CNN fusion pipeline :

$$FP_{f,\text{CNN}} = \prod_{f}^{'} \left(\{P_{m,f}\}_{\forall m \in M} \right)$$

$$VP_{\text{CNN}} = \sum_{M}^{'} \left(\bigcup_{\forall f \in F} FP_{f,\text{CNN}} \right) \tag{7}$$

LSTM fusion pipeline :

$$VP_{\text{LSTM}} = \prod_{F}^{'} \left(\{P_{m,F}\}_{\forall m \in M} \right)$$

CNN and LSTM consolidation :

$$VP_{\text{CNN+LSTM}} = \prod_{F}^{'} \left(VP_{\text{CNN}} \bigcup VP_{\text{LSTM}} \right) \tag{8}$$

Cross-fusions are applied to bring about consolidation at the same hierarchical level (i.e. consolidation of FPs to get a FP or consolidation of VPs to get a VP). On the other hand, self-fusion is applied to move up the hierarchy ladder (i.e. consolidation of FPs to get a VP). As can be observed from Figure 7 there are two parallel fusion pipelines corresponding to CNN and LSTM respectively, that ultimately converge at the last level. For the CNN pipeline, firstly cross-fusion is applied for each of the key-frames followed by a self-fusion operation. Since predictions are already of video-level, the LSTM pipeline do not necessitate self-fusion as such, and thus, only cross-fusion is applied. Finally, the individual VPs from CNN and LSTM streams are consolidated through a cross-fusion operation.

IV. Experimental Results and Discussion

A. Implementation Aspects

All our codes were executed on an Intel Core i7-7700K processor, with a clock-speed of 4.20×8 GHz and Ubuntu 16.04 operating system. Key-frame extraction from videos was implemented in *MATLAB R2018b*. All of the subsequent parts in the pipeline were carried out using *Python 3.6.9* and its respective libraries for scientific computation and computer vision tasks. For the purpose of training, fine-tuning and testing of our proposed multi-pipeline deep architecture, we have relied upon the Keras [44] deep learning library, with Tensorflow [45] as a backend engine.

Input images in the spatial (S) and temporal (T) pipelines were resized to the dimension of 224×224. On the contrary,

images of the attention (A) pipeline contained much concentrated (focused) information and was thus resized to half the previous size, 112×112. Regarding our deep neural architecture, instead of training it from scratch, *transfer learning* was relied upon. We started off with an existing CNN model (for each pipeline) that was pre-trained on the $\sim$15M images of the ImageNet [46] dataset, spanning 1000 categories. With such an initiation of the learnable parameters, the learning process received a huge boost in terms of a reduction in training time and quantity of training data. The topmost layer (of 1000 nodes) in each of the pipelines is replaced by a layer whose size is specific to each dataset. This layer generates a class-wise probability distribution, that is utilized later in our decision-fusion strategy. The LSTM model was trained from the last/penultimate fully-connected layer of CNN.

B. *Datasets and Metric used*

The proposed method was tried and tested on four benchmarking datasets.

Columbia Consumer Videos (CCV) [47]. This dataset is consisted of 9,317 unedited consumer videos from YouTube (due to broken URLs, only 5,046 were downloadable), that can be sub-classified into 20 event categories. Here, some of the events like "wedding-ceremony" and "wedding-dance" share very similar semantic properties, even indistinguishable to the human annotator. This makes the event-recognition task from this collection highly challenging.

Kodak Consumer Videos (KCV) [1]. Spreading 29 event-concepts, this dataset includes 3,321 consumer videos. Although less intra-class variance, these videos suffer from serious quality issues in comparison to the other three datasets. This makes this dataset "unconstrained" to the truest sense.

UCF-101 [2]. In total there are 13,320 short clips from YouTube distributed across 101 action classes. Our choice behind choosing this dataset was backed by the richest [2] action-diversity this dataset possess, ranging from small-scale facial movements to large-scale locomotory activities. The training set of the more recent THUMOS-14 and THUMOS-15 [48] datasets uses the same videos as in UCF-101. But since the untrimmed videos of their test and validation set include multiple activities and we do not try to evaluate multi-class classification performance here, we instead use the train-test split provided by UCF-101.

Human Motion Database (HMDB-51) [49]. This has 7,000 freely-available movie clips spread over 51 human-action classes, with huge-variance with respect to duration.

Although we carried out an unsupervised spatio-temporal action localization on the videos, we do not intend to evaluate its localization performance explicitly. Instead, we analyze whether inclusion of this preprocessing step actually helps our main motive of video classification or not. For the video classification part, we evaluate accuracy by the percentage of correctly classified video instances in the test-set.

C. *Assessment of SALiEnSeA and Proposed Multimodality Fusion*

Evaluation of the Proposed Fusion Strategy. We compare our decision-fusion strategy with other conventional late fusion strategies in use, e.g. Average of prediction [43], Borda count [50] and Highest-rank fusion [51].

In Table 1 and Table 2, we present a detailed comparison of these aforementioned conventional fusion strategies to the proposed biased-conflation based fusion method. To compare different fusion strategies on the same yardstick, they

Symbols used: ✗ = Wrong, ⊗ = Wrong and equal to another, ✓ = Correct
In this table, a pipeline is said to predict correctly when its highest-probable predicted class is same as the video event.
Highest value in each row per dataset is/are highlighted in bold font.

Possible combination for 3 modalities		*Attn*	*Sp*	*Tm*	Total % present in dataset, and % of dataset corrected									
					CCV [47]					KCV [1]				
					% Present in dataset	% of dataset corrected by				% Present in dataset	% of dataset corrected by			
						Avg	Borda	1st-Rank	Ours		Avg	Borda	1st-Rank	Ours
All 3 same	All wrong, but same	⊗	⊗	⊗	6.69	**0.00**	**0.00**	**0.00**	**0.00**	2.24	**0.00**	**0.00**	**0.00**	**0.00**
	All correct	✓	✓	✓	30.96	**30.96**	**30.96**	**30.96**	**30.96**	17.21	**17.21**	**17.21**	**17.21**	**17.21**
2 same, 1 different	All wrong	⊗	⊗	✗	5.00	0.08	0.32	0.04	**0.40**	7.35	0.07	0.19	0.00	**1.01**
	(2 same,	⊗	✗	⊗	2.62	0.06	0.09	0.01	**0.16**	3.14	0.00	**0.15**	0.00	0.00
	1 different)	✗	⊗	⊗	3.73	0.03	0.00	0.03	**0.12**	3.21	**0.00**	**0.00**	**0.00**	**0.00**
	2 same	✓	⊗	⊗	1.71	0.13	0.00	0.08	**0.45**	0.67	0.00	0.00	0.00	**0.06**
	wrong,	⊗	✓	⊗	2.93	1.55	1.38	1.44	**2.02**	1.98	0.63	0.75	**0.90**	0.84
	1 correct	⊗	⊗	✓	2.33	0.31	0.66	**1.07**	0.27	0.86	0.30	0.37	**0.52**	0.49
	2 correct, 1 wrong	✓	✓	✗	13.89	13.50	10.71	6.91	**13.89**	13.92	12.24	7.58	7.50	**13.92**
		✓	✗	✓	2.38	1.69	1.63	1.13	**2.02**	1.01	0.90	0.86	0.45	**0.98**
		✗	✓	✓	8.73	8.55	**8.73**	8.67	**8.73**	6.31	**6.31**	**6.31**	**6.31**	**6.31**
All 3 different	All different wrong	✗	✗	✗	6.90	0.30	0.47	0.13	**1.38**	22.58	0.41	0.90	0.19	**2.39**
	2 different	✗	✗	✓	2.52	1.04	**1.18**	1.10	**1.18**	2.76	1.64	1.61	1.27	**1.91**
	wrong,	✗	✓	✗	7.07	5.24	4.06	3.47	**5.96**	12.73	6.68	5.41	5.71	**7.22**
	1 correct	✓	✗	✗	2.55	0.70	0.54	0.24	**1.38**	4.03	0.82	0.71	0.37	**1.27**
TOTAL (frame-level fusion for all 3 modalities)					100	64.14	60.73	55.28	**68.92**	100	47.70	41.34	41.10	**53.61**

Table 1: Comparison of fusion results in terms of accuracy, for consolidation of frame-level predictions on the CCV [47] and KCV [1] datasets

Symbols used: ✗ = Wrong, ⊗ = Wrong and equal to another, ✓ = Correct
In this table, a pipeline is said to predict correctly when its highest-probable predicted class is same as the video event.
Highest value in each row per dataset is/are highlighted in bold font.

Possible combination for 3 modalities		*Attn*	*Sp*	*Tm*	Total % present in dataset, and % of dataset corrected									
					UCF-101 [2]					HMDB-51 [49]				
					% Present in dataset	% of dataset corrected by				% Present in dataset	% of dataset corrected by			
						Avg	Borda	1st-Rank	Ours		Avg	Borda	1st-Rank	Ours
All 3 same	All wrong, but same	⊗	⊗	⊗	1.00	**0.00**	**0.00**	**0.00**	**0.00**	6.25	**0.00**	**0.00**	**0.00**	**0.00**
	All correct	✓	✓	✓	36.91	**36.91**	**36.91**	**36.91**	**36.91**	13.64	**13.64**	**13.64**	**13.64**	**13.64**
2 same, 1 different	All wrong	⊗	⊗	✗	2.98	0.07	0.26	0.01	**0.73**	5.42	0.03	**0.12**	0.01	0.05
	(2 same,	⊗	✗	⊗	0.81	0.00	**0.03**	0.00	0.00	3.11	0.01	0.08	0.00	**0.22**
	1 different)	✗	⊗	⊗	1.58	0.01	0.00	0.00	**0.05**	9.51	0.05	0.00	0.03	**0.13**
	2 same	✓	⊗	⊗	0.68	0.04	0.00	0.02	**0.22**	2.18	0.16	0.00	0.08	**0.80**
	wrong,	⊗	✓	⊗	1.20	0.48	0.47	**0.61**	0.54	1.24	0.42	0.44	**0.57**	0.44
	1 correct	⊗	⊗	✓	1.04	0.43	0.62	0.56	**0.83**	1.48	0.31	0.56	0.71	**0.85**
	2 correct, 1 wrong	✓	✓	✗	16.42	15.90	10.96	8.40	**16.15**	**5.77**	5.37	3.95	2.87	5.77
		✓	✗	✓	1.73	1.67	1.56	0.83	**1.73**	2.88	2.30	2.00	1.46	**2.88**
		✗	✓	✓	10.18	10.14	**10.18**	10.16	**10.18**	10.54	10.37	**10.54**	10.47	**10.54**
All 3 different	All different wrong	✗	✗	✗	9.05	0.59	0.94	0.18	**1.24**	21.42	0.68	0.85	0.25	**1.29**
	2 different	✗	✗	✓	2.51	**1.79**	1.69	1.24	1.63	5.35	2.49	2.65	2.56	**3.11**
	wrong,	✗	✓	✗	11.10	8.07	5.88	5.42	**9.10**	7.85	5.41	4.12	3.72	**5.92**
	1 correct	✓	✗	✗	2.83	1.36	0.83	0.46	**1.69**	3.37	0.90	0.56	0.31	**1.29**
TOTAL (frame-level fusion for all 3 modalities)					100	77.46	70.33	64.79	**81.00**	100	42.14	39.51	36.68	**46.93**

Table 2: Comparison of fusion results in terms of accuracy, for consolidation of frame-level predictions on the UCF-101 [2] and HMDB-51 [49] datasets

are subdivided into fifteen sub-categories based on the correctness of individual pipelines. We evaluate the performances in order of increasing difficulty levels (DL−1 to DL−7) of getting correct result after fusion. The easiest and trivial case (DL−1) to fuse is ✓✓✓ i.e. when all the modalities vote for the correct event category. The next set of more difficult case (DL−2) is ✓✓✗, ✓✗✓, ✗✓✓ i.e. when any two modalities are correct and the third is wrong. Next comes the scenario (DL−3) of ✗✗✓, ✗✓✗, ✓✗✗ when only one modality gives correct result and other two modalities are wrong, but each voting for different event categories. Subsequent to these, we have an even more difficult case (DL−4) viz. ✓⊗⊗, ⊗✓⊗, ⊗⊗✓ i.e. when only one modality is correct and other two are wrong, but the wrong modalities vote for the same event category. Another case (DL−5) is ✗✗✗ when all the modalities are wrong, but they individually vote for different event categories. Next comes the case (DL−6) of ⊗⊗✗, ⊗✗⊗, ✗⊗⊗ i.e. when all the modalities are wrong, but two vote for the same event category. The most difficult case (DL−7) arises in ⊗⊗⊗ when all three modalities are wrong but, they vote for the same event.
All the fusion techniques perform equally well in DL−1 and after fusion, predicts 100% of all such cases correctly. The efficacy of the proposed fusion strategy can be observed in the middle levels of difficulty i.e. DL−2 to DL−5. In this band, our method outperforms almost all the other fusion strategies in each of the four datasets. Level DL−6 is equally bad for all the fusion strategies because the wrongly predicted event pulls the decision towards itself, by virtue of its majority. The most difficult level i.e. DL−7 is unrecoverable and none of the fusion strategies could render a correct consolidated prediction.

Evaluation of SALiEnSeA. As we mentioned before, the proposed spatial action localization technique SALiEnSeA, can efficiently localize actions in the frame irrespective of the relative position of the subject within a frame. From Figure 6, it is evident that our approach correlates between semantically related but spatially separated objects, and thus can generate a single attention patch around such objects.

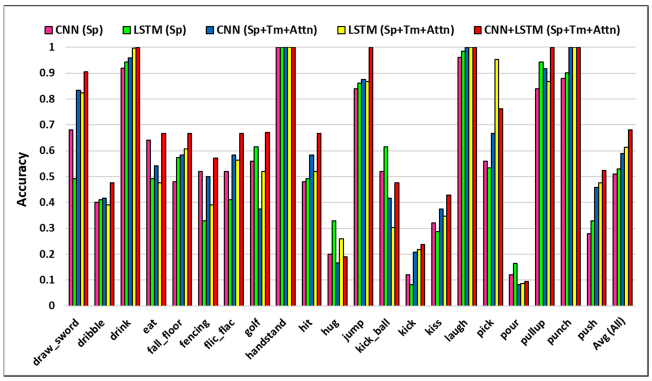


Figure. 8: Comparison of class-wise VP accuracies at different hierarchies of the proposed hierarchical multi-tier fusion strategy, for some selected event-classes of HMDB-51 dataset.

As evident from Figure 8, our attention scheme helps the most in increasing performance for long-shot video classes like "draw-sword", "push", etc. Also for most classes, CNN+LSTM fusion over all the three modalities proffer the highest accuracy, in comparison to their lower-hierarchy counterparts. Regarding the hyperparameters, it was experimentally observed that for morphological operations, window-size (W) of 5×5 was the most efficient in sepa-

fc1 : LSTM trained with features obtained from 1st fully-connected layer of ResNet50
fc2 : LSTM trained with features obtained from 2nd fully-connected layer of ResNet50
fcLast : LSTM trained with features obtained from the last fully-connected layer of ResNet50
Abbreviations : *Sp* = Spatial, *Tm* = Temporal, *Attn* = Attention

Deep Architecture	Type	CCV [47]	KCV [1]	UCF-101 [2]	HMDB-51 [49]
ResNet50 (*Sp*)	FP	65.57	52.16	78.70	44.04
ResNet50 (*Tm*)	FP	48.91	28.13	54.25	38.88
ResNet50 (*Attn*)	FP	54.61	36.84	62.56	32.83
ResNet50 (*Sp+Tm*)	FP	68.58	44.55	80.56	50.47
ResNet50 (*Tm+Attn*)	FP	59.62	35.57	71.69	42.97
ResNet50 (*Sp+Attn*)	FP	68.10	51.25	79.09	44.74
ResNet50 (*Sp+Tm+Attn*)	FP	68.92	53.61	81.00	51.93
ResNet50 (*Sp+Tm*)	VP	**78.84**	57.09	87.33	55.14
ResNet50 (*Tm+Attn*)	VP	72.28	50.09	85.28	56.06
ResNet50 (*Sp+Attn*)	VP	78.25	58.43	86.37	56.45
ResNet50 (*Sp+Tm+Attn*)	VP	78.16	**60.07**	**88.37**	**58.83**
$LSTM_{fc1}$ (*Sp*)	VP	75.93	58.96	85.47	53.33
$LSTM_{fc1}$ (*Tm*)	VP	65.65	39.93	73.35	47.80
$LSTM_{fc1}$ (*Attn*)	VP	70.23	51.31	80.39	47.30
$LSTM_{fc1}$ (*Sp+Tm*)	VP	78.49	57.22	90.30	57.72
$LSTM_{fc1}$ (*Tm+Attn*)	VP	77.14	55.56	84.60	58.01
$LSTM_{fc1}$ (*Sp+Attn*)	VP	78.91	**59.68**	90.50	57.51
$LSTM_{fc1}$ (*Sp+Tm+Attn*)	VP	**79.98**	59.30	**91.92**	**61.29**
$LSTM_{fc2}$ (*Sp*)	VP	75.85	**60.45**	85.02	53.14
$LSTM_{fc2}$ (*Tm*)	VP	65.14	37.69	71.48	47.01
$LSTM_{fc2}$ (*Attn*)	VP	69.97	49.06	76.89	46.12
$LSTM_{fc2}$ (*Sp+Tm*)	VP	77.98	57.97	87.44	60.31
$LSTM_{fc2}$ (*Tm+Attn*)	VP	76.13	53.69	84.22	54.03
$LSTM_{fc2}$ (*Sp+Attn*)	VP	77.22	58.55	87.92	55.14
$LSTM_{fc2}$ (*Sp+Tm+Attn*)	VP	**80.59**	58.55	**90.56**	**61.29**
$LSTM_{fcLast}$ (*Sp*)	VP	75.68	60.07	85.34	53.27
$LSTM_{fcLast}$ (*Tm*)	VP	64.80	36.94	69.42	48.39
$LSTM_{fcLast}$ (*Attn*)	VP	69.72	48.31	75.99	45.60
$LSTM_{fcLast}$ (*Sp+Tm*)	VP	79.75	**61.70**	86.65	58.01
$LSTM_{fcLast}$ (*Tm+Attn*)	VP	76.80	51.44	83.57	54.51
$LSTM_{fcLast}$ (*Sp+Attn*)	VP	78.66	58.55	88.29	56.16
$LSTM_{fcLast}$ (*Sp+Tm+Attn*)	VP	**80.84**	60.80	**90.08**	**60.37**
ResNet50+$LSTM_{fc1}$ (*Sp+Tm*)	VP	82.18	59.09	92.30	64.66
ResNet50+$LSTM_{fc1}$ (*Tm+Attn*)	VP	77.88	52.56	90.06	63.23
ResNet50+$LSTM_{fc1}$ (*Sp+Attn*)	VP	81.33	**62.14**	91.48	64.69
ResNet50+$LSTM_{fc1}$ (*Sp+Tm+Attn*)	VP	**82.84**	60.80	**93.87**	**68.29**
ResNet50+$LSTM_{fc2}$ (*Sp+Tm*)	VP	82.26	59.46	92.14	61.66
ResNet50+$LSTM_{fc2}$ (*Tm+Attn*)	VP	78.21	52.19	90.04	62.05
ResNet50+$LSTM_{fc2}$ (*Sp+Attn*)	VP	81.33	**62.14**	91.48	64.66
ResNet50+$LSTM_{fc2}$ (*Sp+Tm+Attn*)	VP	**82.84**	60.45	**93.76**	**67.12**
ResNet50+$LSTM_{fcLast}$ (*Sp+Tm*)	VP	82.01	59.84	92.35	63.05
ResNet50+$LSTM_{fcLast}$ (*Tm+Attn*)	VP	77.45	52.94	90.22	61.24
ResNet50+$LSTM_{fcLast}$ (*Sp+Attn*)	VP	81.16	**62.14**	91.24	64.66
ResNet50+$LSTM_{fcLast}$ (*Sp+Tm+Attn*)	VP	**83.48**	61.10	**93.79**	**67.12**

Table 3: Performance comparison of CNN and LSTM on each modality, separately and fused

rating small noisy components from their large blob counterparts. Also, an $area_{min}$ covering $20 - 30\%$ of the whole image can almost always identify the subject of a frame, be it long-shot or close-shot.

In Table 3, we tabulate frame-level (FP) and video-level (VP) performances on each of the three modalities. For CCV [47], UCF-101 [2] and HMDB-51 [49] datasets, the highest accuracy is achieved by fusing ResNet50 and LSTM predictions for all the three modalities. For CCV, this was achieved when the LSTM was trained with features from the last fully-connected layer of ResNet50. But for UCF-101 and HMDB-51, features from the first fully-connected layer of ResNet50 gave better results. For KCV [1], highest accuracy is achieved by excluding temporal pipeline. Overall, it can be observed that the attention pipeline almost always plays a supportive role in enhancing accuracy, at each of the hierarchies.

D. Comparing Results with State-of-the-Arts'

In Table 4, we aim to compare our event recognition scheme with state-of-the-arts, and adjudge how the proposed spatio-temporal action localization comes to the aid of it. For CCV, the fact that the result of the proposed method based on three modalities has a better accuracy than others (Jana et al. [8], Li

Abbreviations : *Sp* = Spatial, *Tm* = Temporal, *Attn* = Attention

Dataset	Method	Year	Action Localization?	Multi-stream Info?	Acc (%)
CCV [47]	Pei et al. [16]	2017	Temporal Attention-Gated Model	Only *Tm*	63.00
	Jiang et al. [14]	2017	Feature-Class Relationships	rDNN (*Sp*+*Tm*+Acoustic)	73.50
	Umer et al. [29]	2017	N/A	Only multi-scale *Sp*	80.46
	Soltanian et al. [53]	2018	Conceptwise Power-Law Norm	Frm-level CNN descriptor	75.10
	Li et al. [26]	2018	*Sp*+*Tm* Attention Network	CNN+LSTM (Frm+OptFlow)	80.70
	Jana et al. [8]	2019	Key-Frm (*Tm*)	CNN+LSTM (*Sp*+*Tm*)	81.89
	Li et al. [54]	2020	Hierarchical Attention	ResNet+LSTM (*Sp*+*Tm*)	74.21
	Zhang et al. [52]	2020	Anchor Selection	Transferred CNN+LSTM (*Sp*+*Tm*)	75.10
	PROPOSED	–	Key-Frm (*Tm*)+SALiEnSeA (*Sp*)	CNN+LSTM (Attn+*Sp*+*Tm*)	**83.48**
KCV [1]	Chen et al. [55]	2013	N/A	Space-Time (*Sp*+*Tm*) features	49.61
	Yan et al. [56]	2014	N/A	GLocal Feature Selection	55.60
	Jana et al. [31]	2019	N/A	Only *Tm*	52.41
	Jana et al. [8]	2019	Key-Frm (*Tm*)	CNN+LSTM (*Sp*+*Tm*)	57.52
	PROPOSED	–	Key-Frm (*Tm*)+SALiEnSeA (*Sp*)	CNN+LSTM (Attn+*Sp*+*Tm*)	**62.14**
UCF-101 [2]	Karpathy et al. [18]	2014	Fovea Stream (Center-Crop)	Slow-Fusion (*Tm*), MultiRes-CNN (*Sp*)	68.00
	Simonyan et al. [57]	2014	N/A	*Sp*+*Tm* ConvNet	88.00
	Soltanian et al. [53]	2018	Conceptwise Power-Law Norm	Frm-level CNN descriptor	80.10
	Li et al. [13]	2018	Background Subtraction Feature	3-stream ConvNet+FusionNet	82.80
	Li et al. [26]	2018	*Sp*+*Tm* Attention Network	CNN+LSTM (Frm+OptFlow)	91.60
	Peng et al. [25]	2018	*Sp*+*Tm* Attention	*Sp*+*Tm* Collaborative Learning	94.00
	Jana et al. [8]	2019	Key-Frm (*Tm*)	CNN+LSTM (*Sp*+*Tm*)	89.03
	Zhang et al. [52]	2020	Anchor Selection	Transferred CNN+LSTM (*Sp*+*Tm*)	88.00
	Liu et al. [6]	2020	SAM+TAM	*Sp*+*Tm* Attention ConvNet	**94.33**
	PROPOSED	–	Key-Frm (*Tm*)+SALiEnSeA (*Sp*)	CNN+LSTM (Attn+*Sp*+*Tm*)	93.87
HMDB-51 [49]	Simonyan et al. [57]	2014	N/A	*Sp*+*Tm* ConvNet	59.40
	Girdhar et al. [58]	2017	Pose-regularized Attn Pool	Only *Sp*	54.40
	Li et al. [13]	2018	Background Subtraction Feature	3-stream ConvNet+FusionNet	55.58
	Peng et al. [25]	2018	*Sp*+*Tm* Attention	*Sp*+*Tm* Collaborative Learning	68.70
	Jana et al. [8]	2019	Key-Frm (*Tm*)	CNN+LSTM (*Sp*+*Tm*)	61.91
	Zhang et al. [52]	2020	Anchor Selection	Transferred CNN+LSTM (*Sp*+*Tm*)	59.10
	Liu et al. [6]	2020	SAM+TAM	*Sp*+*Tm* Attention ConvNet	**69.14**
	PROPOSED	–	Key-Frm (*Tm*)+SALiEnSeA (*Sp*)	CNN+LSTM (Attn+*Sp*+*Tm*)	68.29

Table 4: Performance comparison of different approaches on each of the four datasets. The top two accuracy scores are highlighted in gray.

et al. [26] and Zhang et al. [52]) reliant on only two modalities is an indicator of the benefit of having a separate attention pipeline. Regarding KCV dataset also, the proposed method outperforms other methods by achieving an overall accuracy of 62.14%. Our method achieves the third position in the UCF-101 and HMDB-51 datasets, by managing to correctly predict 93.87% and 68.29% of their respective test-set.

V. Conclusion and Future Scope

In this paper, an automated event and activity recognition system is developed, that specializes in handling unconstrained untrimmed low-quality videos. For this, we focus on a preprocessing step involving spatio-temporal action-localization that discards the bulks of information in a video and brings out the most salient 'attention' pieces that are beneficial to classify a video. Temporal attention is achieved by action-localization on the time axis whereby, a set of representative key-frames are extracted from the video. The iterative graph-based approach is such that the distinctness amongst the chosen key-frames and their difference in time-of-appearance are maximized simultaneously. The inputs to attention pipeline is proffered through the proposed SALiEnSeA technique of identifying the high-motion action patch in a video key-frame by adjudging dynamicity of homogeneous motion components. For classification, a three-pipeline hybrid ResNet+LSTM deep architecture is proposed along with a hierarchical late decision-fusion scheme, to combine frame-level and video-level predictions.

In comparison to common late decision-fusion strategies in use, the biased-conflation based fusion was observed to proffer much higher accuracy when three modalities were fused simultaneously. In fact, the *corrective performance* i.e. the ability to correct a fused prediction when majority of its constituent predictions are wrong, is the highest. The attention pipeline proved to be beneficial for improving accuracy, in all the datasets. It was finally observed that the proposed approach outperformed recent state-of-the-arts in datasets of CCV and KCV that are purely constituted of unconstrained and untrimmed videos. Also, our results are at-par with the state-of-the-arts in identifying fine-grained human actions like those in datasets of UCF-101 and HMDB. In the future we aim to incorporate ideas of ethical machine learning to do away with intra- and inter-modality biasness in the multi-stream ResNet+LSTM hybrid architecture.

References

[1] A. Yanagawa, A. C. Loui, J. Luo, S.-F. Chang, D. Ellis, W. Jiang, L. Kennedy, and K. Lee, "Kodak Consumer Video Benchmark Data Set: Concept Definition and Annotation," *Columbia Uni-*

versity ADVENT Technical Report, pp. 246–2008, 2008. Available: http://www.ee.columbia.edu/ln/dvmm/consumervideo/ (last accessed: April, 2021).

[2] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A Dataset of 101 Human Actions Classes from Videos in the Wild," *arXiv preprint arXiv:1212.0402*, 2012. Available: https://www.crcv.ucf.edu/data/UCF101.php (last accessed: April, 2021).

[3] S. Agarwal, B. Santra, and D. P. Mukherjee, "Anubhav: Recognizing Emotions through Facial Expression," *The Visual Computer*, vol. 34, no. 2, pp. 177–191, 2018.

[4] G. Li, H. Tang, Y. Sun, J. Kong, G. Jiang, D. Jiang, B. Tao, S. Xu, and H. Liu, "Hand Gesture Recognition based on Convolution Neural Network," *Cluster Computing*, vol. 22, no. 2, pp. 2719–2729, 2019.

[5] S. Hongeng, R. Nevatia, and F. Bremond, "Video-based Event Recognition: Activity Representation and Probabilistic Recognition Methods," *Computer Vision and Image Understanding*, vol. 96, no. 2, pp. 129–162, 2004.

[6] S. Liu, X. Ma, H. Wu, and Y. Li, "An End to End Framework with Adaptive Spatio-Temporal Attention Module for Human Action Recognition," *IEEE Access*, 2020.

[7] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning Spatiotemporal Features with 3D Convolutional Networks," in *Proceedings of the International Conference on Computer Vision (ICCV)*, pp. 4489–4497, IEEE, 2015.

[8] P. Jana, S. Bhaumik, and P. P. Mohanta, "A Multitier Fusion Strategy for Event Classification in Unconstrained Videos," in *Proceedings of the International Conference on Pattern Recognition and Machine Intelligence (PReMI)*, pp. 515–524, Springer, 2019.

[9] P. Jana, S. Bhaumik, and P. P. Mohanta, "Unsupervised Action Localization Crop in Video Retargeting for 3D ConvNets," in *2021 IEEE Region 10 Conference (TENCON)*, IEEE, 2021.

[10] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza, "High Speed and High Dynamic Range Video with an Event Camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[11] S. Mejia, J. E. Lugo, R. Doti, and J. Faubert, "Pedestrian Modeling using the Least Action Principle with Sequences obtained from Thermal Cameras in a Real Life Scenario," *International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM)*, vol. 9 (2017), pp. 145–152, 2017.

[12] S. Bhaumik, P. Jana, and P. P. Mohanta, "Event and Activity Recognition in Video Surveillance for Cyber-Physical Systems," in *Emergence of Cyber Physical System and IoT in Smart Automation and Robotics: Computer Engineering in Automation*, pp. 51–68, Cham: Springer, 2021.

[13] C. Li and Y. Ming, "Three-stream Convolution Networks after Background Subtraction for Action Recognition," in *Video Analytics. Face and Facial Expression Recognition*, pp. 12–24, Springer, 2018.

[14] Y.-G. Jiang, Z. Wu, J. Wang, X. Xue, and S.-F. Chang, "Exploiting Feature and Class Relationships in Video Categorization with Regularized Deep Neural Networks," *IEEE Transactions on Pattern Analysis and Machine intelligence*, vol. 40, no. 2, pp. 352–364, 2017.

[15] Y.-G. Jiang, S. Bhattacharya, S.-F. Chang, and M. Shah, "High-level Event Recognition in Unconstrained Videos," *International Journal of Multimedia Information Retrieval*, vol. 2, no. 2, pp. 73–101, 2013.

[16] W. Pei, T. Baltrusaitis, D. M. Tax, and L.-P. Morency, "Temporal Attention-Gated Model for Robust Sequence Classification," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6730–6739, IEEE, 2017.

[17] S. S. Aote and A. Potnurwar, "An Automatic Video Annotation Framework based on Two Level Keyframe Extraction Mechanism," *Multimedia Tools and Applications*, vol. 78, no. 11, pp. 14465–14484, 2019.

[18] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale Video Classification with Convolutional Neural Networks," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1725–1732, IEEE, 2014.

[19] S. Bharati, P. Podder, and M. R. H. Mondal, "Artificial Neural Network Based Breast Cancer Screening: A Comprehensive Review," *International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM)*, vol. 12 (2020), pp. 125–137, 2020.

[20] P. Jana and P. P. Mohanta, "Recent Trends in 2D Object Detection and Its Use in Video Event Recognition," in *Advancement of Deep Learning and its Applications in Object Detection and Recognition*, River Publishers, 2022.

[21] M. Burić, M. Pobar, and M. Ivašić-Kos, "Object Detection in Sports Videos," in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 1034–1039, IEEE, 2018.

[22] G. Gkioxari, R. B. Girshick, and J. Malik, "Contextual Action Recognition with R*CNN," *arXiv preprint arXiv:1505.01197v3*, 2015.

[23] C. Pacheco, E. Mavroudi, E. Kokkoni, H. G. Tanner, and R. Vidal, "A Detection-based Approach to Multiview Action Classification in Infants," in *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 6112–6119, IEEE, 2021.

[24] Z. Li, Y. Huang, M. Cai, and Y. Sato, "Manipulation-Skill Assessment from Videos with Spatial Attention Network," in *Proceedings of the International Conference on Computer Vision (ICCV) Workshops*, IEEE, 2019.

[25] Y. Peng, Y. Zhao, and J. Zhang, "Two-stream Collaborative Learning with Spatial-Temporal Attention for Video Classification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 3, pp. 773–786, 2018.

[26] D. Li, T. Yao, L.-Y. Duan, T. Mei, and Y. Rui, "Unified Spatio-Temporal Attention Networks for Action Recognition in Videos," *IEEE Transactions on Multimedia*, vol. 21, no. 2, pp. 416–428, 2018.

[27] N. Adhikari, S. Bhattacharya, and M. Sultana, "A Comprehensive Survey on Bird Species Identification Models," *International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM)*, vol. 13 (2021), pp. 319–335, 2021.

[28] R. M. Kishi, T. H. Trojahn, and R. Goularte, "Correlation based Feature Fusion for the Temporal Video Scene Segmentation Task," *Multimedia Tools and Applications*, vol. 78, no. 11, pp. 15623–15646, 2019.

[29] S. Umer, M. Ghorai, and P. P. Mohanta, "Event Recognition in Unconstrained Video using Multi-scale Deep Spatial Features," in *Proceeding of the 9th International Conference on Advances in Pattern Recognition (ICAPR)*, IEEE, 2017.

[30] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond Short Snippets: Deep Networks for Video Classification," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 4694–4702, IEEE, 2015.

[31] P. Jana, S. Bhaumik, and P. P. Mohanta, "Key-frame based Event Recognition in Unconstrained Videos using Temporal Features," in *Proceedings of the IEEE Region 10 Symposium (TENSYMP)*, pp. 349–354, IEEE, 2019.

[32] Y. Ahmine, G. Caron, E. M. Mouaddib, and F. Chouireb, "Adaptive Lucas-Kanade Tracking," *Image and Vision Computing*, vol. 88, 2019.

[33] G. Farnebäck, "Two-frame Motion Estimation based on Polynomial Expansion," in *Proceedings of the Scandinavian Conference on Image Analysis*, pp. 363–370, Springer, 2003.

[34] L. Sevilla-Lara, D. Sun, V. Jampani, and M. J. Black, "Optical Flow with Semantic Segmentation and Localized Layers," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3889–3898, 2016.

[35] M. Bai, W. Luo, K. Kundu, and R. Urtasun, "Exploiting Semantic Information and Deep Matching for Optical Flow," in *European Conference on Computer Vision*, pp. 154–170, Springer, 2016.

[36] A. Mukherjee, P. Jana, S. Chakraborty, and S. K. Saha, "Two Stage Semantic Segmentation by SEEDS and Fork Net," in *2020 IEEE Calcutta Conference (CALCON)*, pp. 283–287, IEEE, 2020.

[37] K. Potter, H. Hagen, A. Kerren, and P. Dannenmann, "Methods for Presenting Statistical Information: The Box Plot," *Visualization of Large and Unstructured Data Sets*, vol. 4, pp. 97–106, 2006.

[38] P. Jana, S. Ghosh, R. Sarkar, and M. Nasipuri, "A Fuzzy C-means based Approach Towards Efficient Document Image Binarization," in *2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR)*, pp. 332–337, IEEE, 2017.

[39] P. Jana, S. Ghosh, S. K. Bera, and R. Sarkar, "Handwritten Document Image Binarization: An Adaptive K-means based Approach," in *2017 IEEE Calcutta Conference (CALCON)*, pp. 226–230, IEEE, 2017.

[40] C. A. Pickover, *Time: A Traveler's Guide*. Oxford University Press, USA, 1999.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, IEEE, 2016.

[42] S. Hochreiter and J. Schmidhuber, "Long Short-term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[43] T. Hill, "Conflations of Probability Distributions," *Transactions of the American Mathematical Society*, vol. 363, no. 6, pp. 3351–3372, 2011.

[44] F. Chollet *et al.*, "Keras," 2015. Available: `https://keras.io/` (last accessed: April, 2021).

[45] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, *et al.*, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," 2015. Available: `https://www.tensorflow.org/` (last accessed: April, 2021).

[46] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, pp. 1097–1105, 2012.

[47] Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui, "Consumer Video Understanding: A Benchmark Database and an Evaluation of Human and Machine Performance," in *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, 2011. Available: `http://www.ee.columbia.edu/ln/dvmm/CCV/` (last accessed: April, 2021).

[48] H. Idrees, A. R. Zamir, Y.-G. Jiang, A. Gorban, I. Laptev, R. Sukthankar, and M. Shah, "The THUMOS Challenge on Action Recognition for Videos 'in the Wild'," *Computer Vision and Image Understanding*, vol. 155, 2017.

[49] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A Large Video Database for Human Motion Recognition," in *Proceedings of the 2011 International Conference on Computer Vision*, pp. 2556–2563, IEEE, 2011. Available: https://serre-lab.clps.brown.edu/resource/hmdb-a-large-human-motion-database/ (last accessed: April, 2021).

[50] P. Emerson, "The Original Borda Count and Partial Voting," *Social Choice and Welfare*, vol. 40, no. 2, pp. 353–358, 2013.

[51] M. Singh, R. Singh, and A. Ross, "A Comprehensive Overview of Biometric Fusion," *Information Fusion*, vol. 52, pp. 187–205, 2019.

[52] L. Zhang and X. Xiang, "Video Event Classification based on Two-stage Neural Network," *Multimedia Tools and Applications*, 2020.

[53] M. Soltanian and S. Ghaemmaghami, "Hierarchical Concept Score Postprocessing and Concept-Wise Normalization in CNN-Based Video Event Recognition," *IEEE Transactions on Multimedia*, vol. 21, no. 1, pp. 157–172, 2018.

[54] Y. Li, C. Liu, Y. Ji, S. Gong, and H. Xu, "Spatio-Temporal Deep Residual Network with Hierarchical Attentions for Video Event Recognition," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 16, no. 2s, 2020.

[55] L. Chen, L. Duan, and D. Xu, "Event Recognition in Videos by Learning from Heterogeneous Web Sources," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2666–2673, IEEE, 2013.

[56] Y. Yan, H. Shen, G. Liu, Z. Ma, C. Gao, and N. Sebe, "GLocal Tells You More: Coupling GLocal Structural for Feature Selection with Sparsity for Image and Video Classification," *Computer Vision and Image Understanding*, vol. 124, pp. 99–109, 2014.

[57] K. Simonyan and A. Zisserman, "Two-stream Convolutional Networks for Action Recognition in Videos," in *Proceedings of the Conference on Neural Information Processing Systems*, pp. 568–576, 2014.

[58] R. Girdhar and D. Ramanan, "Attentional Pooling for Action Recognition," in *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, pp. 34–45, 2017.

Author Biographies

Prithwish Jana is pursuing Master of Technology (M.Tech) in Computer Science and Engineering (CSE) at Indian Institute of Technology (IIT) Kharagpur, India. Earlier, he completed his Bachelor of Engineering (B.E) in CSE from Jadavpur University, Kolkata, India in 2020. He won Jadavpur University Gold Medal 2020 for being the best graduating student in the CSE department. Prithwish's research interests include artificial intelligence, deep machine learning, computer vision, medical imaging, data science and fairness in decision making. He is a recipient of several esteemed accolades and fellowships including Indian National Talent Search (NTSE 2012), Kishore Vaigyanik Protsahan Yojana (KVPY 2015), Jagadis Bose National Science Talent Search (JBNSTS 2016), Reliance Foundation Scholarship (RFS 2021) in Artificial Intelligence & Computer Science. He has published book chapters in Springer and River Publishers and several research papers in prominent conferences including ICAPR, PReMI, IEEE CALCON, TENSYMP, TENCON and ACM WWW.

Swarnabja Bhaumik received his B.Tech degree in Computer Science and Engineering from Meghnad Saha Institute of Technology, Kolkata, India in 2020. He has two years of experience in the IT corporate sector and is currently working as an Analytics and Cognitive Python Data Engineer in Deloitte India (Offices of the US). He enjoys recreational mathematics and has research interests in computer vision, image processing and machine learning.

Partha Pratim Mohanta received Ph. D. in Engineering from Jadavpur University, Kolkata, India in 2014. Prior to that he received the B.Sc. and M.Sc. degree in Statistics from University of Kalyani, West Bengal, India in 1995 and 1997 respectively and obtained PGDCA from Regional Computer Centre, Calcutta, West Bengal, India in 1998. Currently, he is Associate Scientist 'C' in Electronics and Communication Sciences Unit of the Indian Statistical Institute. He has successfully completed a number of research project on video summarization, video event recognition and video captioning. His research interests include Image Processing, Video Processing, Pattern Recognition and Machine Learning. He has published more than 20 articles in the internationally reputed journals, conferences and book chapters.