

Concept Drift Detection for Multivariate Data Streams and Temporal Segmentation of Daylong Egocentric Videos

Anonymous Author(s)

Submission Id: 1702



Figure 1: The above figure shows the complexity of the temporal segmentation of egocentric videos. The first row shows a significant change in the scene due to head movement but there is no ground truth boundary whereas the second row shows boundary in the segment but no significant change in visuals.

ABSTRACT

The long and unconstrained nature of egocentric videos makes it imperative to use temporal segmentation as an important pre-processing step for many higher-level inference tasks. Activities of the wearer in an egocentric video typically span over hours and are often separated by slow, gradual changes. Furthermore, the change of camera viewpoint due to the wearer’s head motion causes frequent and extreme, but, spurious scene changes. The continuous nature of boundaries makes it difficult to apply traditional Markov Random Field (MRF) pipelines relying on temporal discontinuity, whereas deep Long Short Term Memory (LSTM) networks gather context only upto a few hundred frames, rendering them ineffective for egocentric videos. In this paper, we present a novel unsupervised temporal segmentation technique especially suited for day-long egocentric videos. We formulate the problem as detecting concept drift in a time-varying, non i.i.d. sequence of frames. Statistically bounded thresholds are calculated to detect concept drift between two temporally adjacent multivariate data segments with different underlying distributions while establishing guarantees on false positives. Since, the derived threshold indicates confidence in the prediction, it can also be used to control the granularity of the output segmentation. Using our technique, we report significantly improved state of the art f-measure for daylong egocentric video datasets, as well as photostream datasets derived from them: HUJI (73.01%, 59.44%), UTEgo (58.41%, 60.61%) and Disney (67.63%, 68.83%).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Qi ’20, Oct 12–16, 2020, Seattle, USA

© 2020 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM... \$15.00

<https://doi.org/10.1145/1122445.1122456>

ACM Reference Format:

Anonymous Author(s). 2020. Concept Drift Detection for Multivariate Data Streams and Temporal Segmentation of Daylong Egocentric Videos. In *Proceedings of ACM Multimedia (Qi ’20)*. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/1122445.1122456>

1 INTRODUCTION

Egocentric videos are captured from the cameras typically harnessed on wearer’s head, recording first person perspective in a hands-free, always-on manner. This makes captured videos extremely long (often spanning the whole day), and highly redundant. The natural head motion of the camera wearer causes fast and extreme changes in the viewpoint. The wild camera ego-motion, coupled with the unconstrained environment in which such videos are usually captured, makes the videos extremely hard to watch, and even more challenging to process by traditional computer vision techniques. This has motivated the computer vision community to develop novel techniques designed for analyzing egocentric videos [2–4, 9, 10, 21, 39].

The focus of this paper is on temporal video segmentation of daylong egocentric video streams. Due to the task’s utility as a pre-processing for many higher-level inference problems like indexing and summarization, the problem is a well-researched area in computer vision: both for the first person [7, 20, 37, 38, 46] as well as third person videos [27, 42, 47].

Common techniques for temporal segmentation of third person videos are based on either MRF formulation or deep neural network (DNN) with RNN/LSTM units. The former techniques [26] look for temporal discontinuities, and hence fail for egocentric videos when the segment boundaries are often slow with gradual changes in the scene. DNN based techniques [6, 15, 16, 35] use recurrent connections to capture the temporal context and do not scale well for long segments. To better understand the scales involved, a 10 minutes video segment captured at 30 frames per second (FPS) contains 18000 frames. Even with sophisticated back-propagation techniques [31], it is hard to train RNNs for such a long sequence.

Methods	Unsupervised	Multivariate Data	Scalability to Long Sequences	Customized Granularity	Works with Extremely Shaky Videos
TCFPN [17]	✗	✓	✗	✗	✗
ADWIN [7]	✓	✗	✓	✓	✓
SR-Clustering [14]	✓	✓	✗	✓	✓
CES [12]	✓	✓	✗	✗	✗
Ours	✓	✓	✓	✓	✓

Table 1: Comparison of state of the art with the proposed method on various criteria important for applicability to egocentric videos.

Multi-scale network designs [15, 17, 29] are possible but compromise temporal resolution to gain long term context. Fig 1 shows few challenges in the temporal segmentation of egocentric videos.

For temporal segmentation of egocentric videos, researchers have suggested to use both generic (e.g. RGB, Optical flow, etc.) as well as egocentric specific cues (e.g. hand pose, handled object, etc.). However the techniques are often limited to either short segments [23] or segmentation based on long term activities but with short term signatures [6, 37, 38]. For example, to detect long term ‘walking’ activity, [38] independently classifies a video clip of 4 secs.

In this paper, we propose to formulate the problem of temporal segmentation as concept drift detection in multivariate time series data. In a concept drift detection task, one maintains two adjacent temporal windows of fixed size and estimate statistical summary (e.g. average) of the two windows separately. If the summary is significantly different for the two windows, the algorithm declares concept drift. The key challenges to use the formulation for temporal segmentation are: (1) Choosing window length for the statistical summary, as different activity/event lengths may require different temporal windows, and (2) Choosing threshold to declare a boundary, as real boundaries may have smooth visual changes, whereas sharp head motion may cause significant visual changes in non-boundary regions. We emphasize that the proposed formulation can incorporate various other cues suggested for temporal segmentation of egocentric videos viz optical flow, hand pose, and other objects present in the scene, etc. Our primary contribution is in suggesting a way to deal with smooth changes in the features at the real boundaries compared to sharper changes at the spurious boundaries as illustrated in Fig. 1.

Bifet and Gavalda [7] have suggested a technique, called ADWIN, to segment i.i.d. univariate sequences. Their method maintains an adaptive window, and for each of its various partitions into two sub-windows, a threshold is calculated based upon the harmonic mean of the length of the two sub-windows. A boundary is declared if the difference of the statistical summaries of the two sub-windows is larger than this threshold. The threshold is based on the Hoeffding’s inequality and is valid for all probability distributions. ADWIN gives probabilistic bounds on the boundary detection error and works for univariate sequences with slow as well as abrupt changes.

In this paper, we propose a technique for concept drift detection in multivariate, and non-i.i.d. sequences such as egocentric videos, which can be used for temporal segmentation of such videos. Table 1 compares the key strengths of our approach with state of the art. The specific contributions of this paper are as follows:

- (1) To the best of our knowledge, we are the first to suggest formulating the problem of temporal segmentation of extremely long egocentric videos as detecting concept drift in a time series data.
- (2) We use a multivariate generalization of Hoeffding’s bound to compute distribution invariant segmentation threshold for multivariate time series arising out of a given frame sequence.
- (3) Hoeffding’s bound as such assumes i.i.d. samples and can not be used for video sequences with a large correlation between temporal neighbors. We suggest a simple heuristic of jump factor to get around the problem.
- (4) In our experiments on both day-long egocentric videos, as well as benchmark photo-stream datasets, the proposed technique successfully copes with two key egocentric specific challenges viz continuous as well as extreme viewpoint variations, and long segments. Our technique gives significantly improved f-score of 59.44%, on HUJI [38], in comparison to current state of the art of 45.70% by [12].

2 RELATED WORK

Related Tasks: We note that the solution to action localization as well as scene segmentation results in the temporal segmentation of videos. Action localization refers to predicting the temporal bounds of pre-specified action categories in an input video and is typically accomplished using supervised learning techniques. Researchers have looked at the problem in both third person [5, 8, 11, 15, 18, 32, 35, 41], as well as the first person contexts [2, 6, 9, 10, 23, 25]. On the other hand, the focus of our paper is on unsupervised segmentation with no prior knowledge of output categories. Similarly, in a scene segmentation task, one is looking at the boundaries separating two visually different scenes. In a scene segmentation scenario the boundaries are usually sharp, which is not true for the case of egocentric videos. Besides, as described earlier, wearer’s head motion and the resulting sharp viewpoint changes may induce false segmentation using a typical scene segmentation technique.

Deep Learning Techniques for Temporal Segmentation: In the last decade, DNNs have emerged as a leading technique for several computer vision problems, including the temporal video segmentation [1, 12, 16, 20, 27, 38]. Temporal Convolutional Networks (TCNs) and its variants [16, 17, 29] harness local motion information and use a hierarchy of temporal convolutional filters to capture longer range patterns. [16] has proposed a hybrid of LSTM and TCN to capture local motion as well as longer term context. [36]

uses Siamese Neural Network to detect context change between two consecutive low-resolution images for egocentric photostreams. [13] and [12] use LSTM based generative model to predict the future context and track their evolution to decide the event boundaries in continuous photostreams. [1] uses a self-supervised perceptual predictive model for contextual event segmentation. All these methods fail to scale for hours long egocentric video segments, as the gradients during backpropagation vanish beyond a few hundred-time steps [31]. Besides, most of the techniques are supervised and require a large amount of training data, which is extremely hard in privacy-sensitive egocentric context.

Traditional Techniques for Temporal Segmentation: Traditional techniques for temporal segmentation of third person videos [18, 22, 24, 40, 41, 45] utilize variations of fixed-size sliding window approach to generate the start and end times of all the events in a video. These methods generally specify windows of different sizes and slide them across a video to generate event proposals of corresponding sizes. The overlapping proposals generated are further processed to remove overlap and select only the most relevant proposals. These methods are computationally expensive and require a large scale space search to handle events with significantly varying lengths, making them impractical for egocentric videos. For instance, in Disney egocentric dataset, events can be less than 5 minutes (for social interactions), to more than 30 minutes (for lunch).

Adaptive Windowing: To handle events of variable lengths one can use adaptive windowing [7], and maintain the size of the window dynamically. Such techniques keep on growing the window if the current event is long, and drop a sub-window from the tail if an event boundary is detected. [14] combines low-level features with high-level semantic labels, and has suggested a graph cut technique to look for the trade-off between the adaptive windowing [7] and agglomerative clustering.

3 PROPOSED APPROACH

We start this section with our theoretical contributions. We first describe Matrix Hoeffding's bound and adapt it for multivariate data. Then we use the derived bound for our novel concept drift detection formulation in multivariate data sequences. While the discussion until here will assume the input samples (frames in our case) to be independent, we end the section with details on how to deal with temporally correlated data streams.

3.1 Multivariate Hoeffding's Bound

The standard result for Hoeffding's inequality for random symmetric matrices may be given as the following [34]:

LEMMA 3.1. Consider a finite sequence Z_i of independent, random, symmetric matrices with dimension d , and a sequence of fixed symmetric matrices P_i , such that $\mathbb{E}[Z_i] = 0$ and $Z_i^2 \leq P_i^2$, almost surely. Here, \leq denotes the semi-definite order on symmetric matrices. Then for all $\epsilon \geq 0$, we have:

$$\mathbb{P}\left(\left\|\sum_i Z_i\right\|_s \geq \epsilon\right) \leq d \exp\left(\frac{-\epsilon^2}{2\sigma^2}\right), \quad (1)$$

where $\sigma^2 = \frac{1}{2}\left\|\sum_i (P_i^2 + \mathbb{E}[Z_i^2])\right\|_s$, and $\|X\|_s$ denotes the spectral norm of X .

For our case, we assume that $\mathbb{E}[Z_i^2] \approx Z_i^2$, and $Z_i \approx P_i$, and hence compute σ^2 as simply $\left\|\sum_i P_i^2\right\|_s$. Note that the result as such is valid only for the symmetric matrices. We extend it to the vector data-streams using the Jordan-WieLaudt theorem [43] as described below. Consider a vector X of size $d \times 1$. Let A be a block matrix such that $A = \begin{bmatrix} 0 & X \\ X^T & 0 \end{bmatrix}$. Since, A is a symmetric matrix with dimension $(d+1) \times (d+1)$, we can use Eq. (1) for the matrix A , such that:

$$\mathbb{P}\left(\left\|\sum_i A_i\right\|_s \geq \epsilon\right) \leq (d+1) \exp\left(\frac{-\epsilon^2}{2\sigma^2}\right), \quad (2)$$

where $\sigma^2 = \left\|\sum_i A_i^2\right\|_s$. It can also be shown that: $A^2 = \begin{bmatrix} XX^T & 0 \\ 0 & X^T X \end{bmatrix}$, and that A 's non-zero eigenvalues are ± 1 times the singular values of X . Hence $\|A\|_s = \|X\|_2$, where $\|X\|_2$ denotes the ℓ_2 norm of the vector X . Using the result in the equation above:

$$\mathbb{P}\left(\left\|\sum_i X_i\right\|_2 \geq \epsilon\right) \leq (d+1) \exp\left(\frac{-\epsilon^2}{2\sigma^2}\right), \quad (3a)$$

$$\text{where } \sigma^2 = \max\left(\left\|\sum_i \mathbb{E}[X_i X_i^T]\right\|_s, \left\|\sum_i \mathbb{E}[X_i^T X_i]\right\|_s\right) \quad (3b)$$

We use the above result to compute the bound for the average as:

$$\begin{aligned} \mathbb{P}\left(\left\|\frac{1}{n} \sum_i X_i\right\|_2 \geq \epsilon\right) &= \mathbb{P}\left(\frac{1}{n} \left\|\sum_i X_i\right\|_2 \geq \epsilon\right) \\ &= \mathbb{P}\left(\left\|\sum_i X_i\right\|_2 \geq n\epsilon\right) \\ &\leq (d+1) \exp\left(\frac{-n^2\epsilon^2}{2\sigma^2}\right). \quad (\text{Using Eq. (3a)}) \end{aligned}$$

Denoting $\bar{X} = \frac{1}{n} \sum_i X_i$, and $\bar{\sigma}^2 = \sigma^2/n$

$$\mathbb{P}\left(\left\|\bar{X}\right\|_2 \geq \epsilon\right) \leq (d+1) \exp\left(\frac{-n\epsilon^2}{2\bar{\sigma}^2}\right), \quad (4)$$

Note that, if we assume the ℓ_2 norm of X as 1, then $X_i^T X_i = 1$, and σ^2 as given in Eq. (3b) is always 1. We summarize our result as the following theorem:

THEOREM 3.2. Let X_1, \dots, X_n be d dimensional, independent random vectors with $\mathbb{E}[X] = 0$, and unit ℓ_2 norm. Then:

$$\mathbb{P}\left(\left\|\bar{X}\right\|_2 \geq \epsilon\right) \leq (d+1) \exp\left(\frac{-n\epsilon^2}{2}\right), \quad (5)$$

where \bar{X} denotes the observed mean of the samples.

3.2 Concept Drift Detection for Multivariate Data

We formulate the temporal segmentation of egocentric videos as concept drift detection in a data stream. While in reality, the adjacent frames in the video stream are not conditionally independent of each other, for this section, we will assume so. In the next section, we describe our proposal to get around the assumption.

Concept Drift Detection Pipeline: For the concept drift detection, one maintains a sliding window, w , of dynamic length, n , over the sequence. Consider a hypothesis that there is a segment boundary at index t within the window, i.e., there is a particular segment, w_1 , of length n_1 , from $[0, t)$ and another segment, w_2 , of length n_2 , from $[t, n]$. We assume that the data in two segments is from two unknown

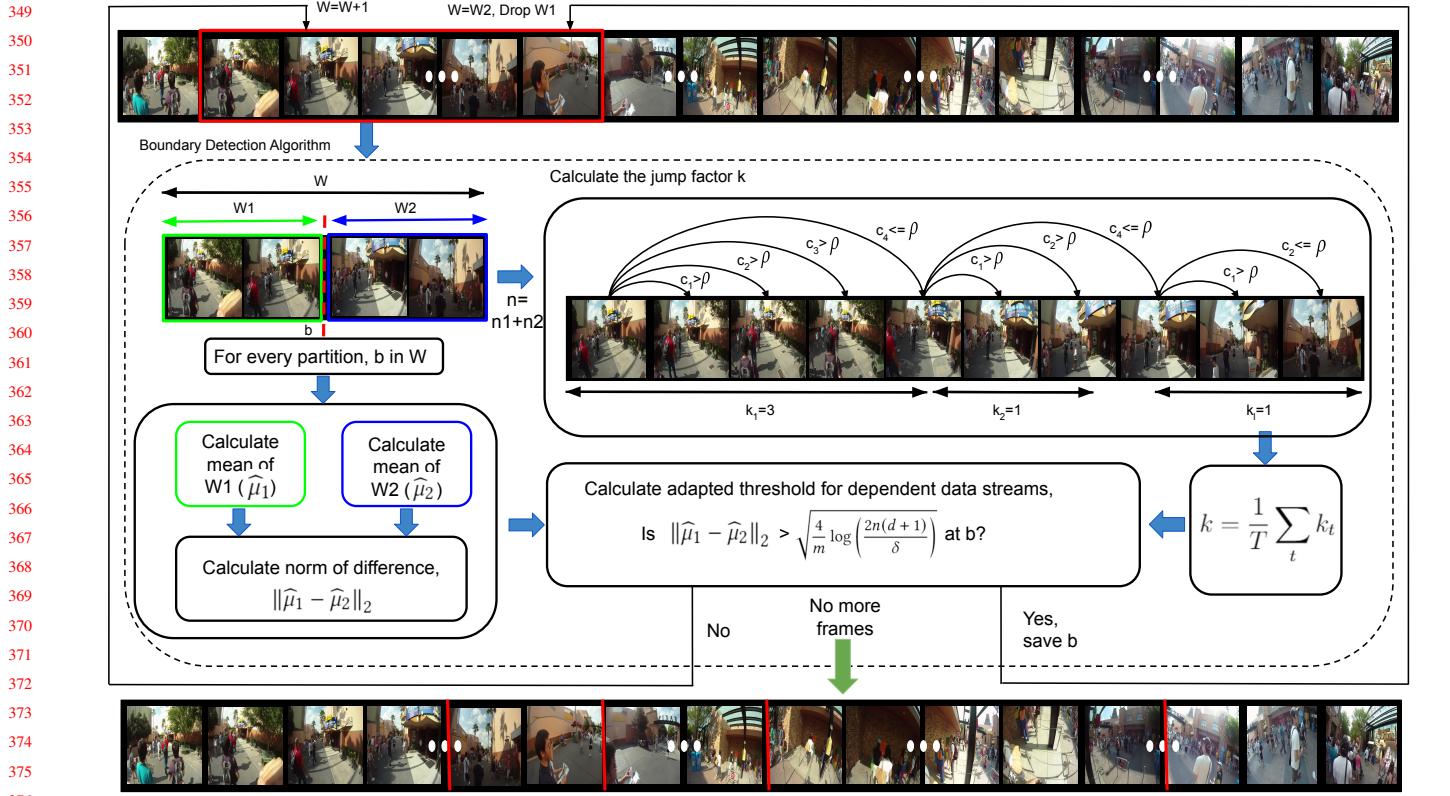


Figure 2: The block diagram describing major steps of the proposed approach. The c_i represents the correlation coefficient between the two frames. Please refer to the main paper for the details

distributions with the observed mean values of $\hat{\mu}_1$ and $\hat{\mu}_2$ respectively. If for a particular partition, the score ($\|\hat{\mu}_1 - \hat{\mu}_2\|_2$) exceeds a threshold ϵ_{cut} , we would like to declare a detected boundary at t and the segment w_1 will be dropped from w . Otherwise, a new sample is added to the current window w , and the process is repeated for this new window of size $n + 1$. For each window w , the boundary hypothesis is tested for all indices $t \in w$. Below we describe a way to compute the threshold ϵ_{cut} in a principled manner using multiple hypothesis testing.

Multiple Hypothesis Testing: One of the ways to calculate the threshold ϵ_{cut} is by bounding the error rate for declaring incorrect segment boundaries. Let us denote the observed mean of the segments, as $\hat{\mu}_1, \hat{\mu}_2$ respectively, and the true (unobserved) mean of the current window as μ_w . We perform hypothesis testing with $\hat{\mu}_1 = \hat{\mu}_2 = \mu_w$ as the null hypothesis. In other words, our null hypothesis is that the two segments come from the same, but unknown, distribution. Since we perform multiple tests in a single window for various values of t , hence as per the multiple hypothesis testing problem in the statistics, we would like to increase the threshold of accepting the hypothesis by n (size of the window or number of tests). For the hypothesis accepting the probability of δ , we would like to set the ϵ_{cut} such that:

$$\mathbb{P}(\|\hat{\mu}_1 - \hat{\mu}_2\|_2 \geq \epsilon_{cut}) \leq \frac{\delta}{n}. \quad (6)$$

The following lemma bounds the probability of difference in the observed means:

LEMMA 3.3. *For a sequence of d -dimensional random vectors, $\{X_1, \dots, X_n\}$, sampled from an unknown but stationary probability distribution, and its arbitrary partition into two subsets w_1 , and w_2 , with lengths n_1 , and n_2 , and observed means $\hat{\mu}_1$, and $\hat{\mu}_2$ respectively:*

$$\mathbb{P}(\|\hat{\mu}_1 - \hat{\mu}_2\|_2 \geq \epsilon) \leq 2(d+1) \exp\left(\frac{-m\epsilon^2}{4}\right), \quad (7)$$

where m is the harmonic mean of n_1 and n_2 .

PROOF. Consider the following three events:

- **Event A:** $\|\hat{\mu}_1 - \hat{\mu}_2\|_2 < \epsilon$.
- **Event B:** $\|\hat{\mu}_1\|_2 < k\epsilon$.
- **Event C:** $\|\hat{\mu}_2\|_2 < (1-k)\epsilon$.

Here, k is a real number $\in (0, 1)$. Further, from triangle inequality:

$$\|\hat{\mu}_1 - \hat{\mu}_2\|_2 \leq \|\hat{\mu}_1\|_2 + \|\hat{\mu}_2\|_2 \quad (8)$$

Assuming Events B and C hold:

$$\Rightarrow \|\hat{\mu}_1 - \hat{\mu}_2\|_2 < k\epsilon + (1-k)\epsilon \quad (9)$$

$$\Rightarrow \|\hat{\mu}_1 - \hat{\mu}_2\|_2 < \epsilon. \quad (10)$$

Hence, we can say that $B \cap C \subseteq A$, which implies $A^c \subseteq B^c \cup C^c$, where S^c denotes the complement of the set S . Therefore, from union

465 bound rule of the probability theory:

$$466 \quad \mathbb{P}(A^c) \leq \mathbb{P}(B^c) + \mathbb{P}(C^c) \quad (11)$$

467 Using event defintions as given above:

$$468 \quad \mathbb{P}(\|\hat{\mu}_1 - \hat{\mu}_2\|_2 \geq \epsilon) \leq \mathbb{P}(\|\hat{\mu}_1\|_2 \geq k\epsilon) + \mathbb{P}(\|\hat{\mu}_2\|_2 \geq (1-k)\epsilon)$$

469 Using Theorem 3.2

$$470 \quad \mathbb{P}(\|\hat{\mu}_1 - \hat{\mu}_2\|_2 \geq \epsilon) \leq (d+1) \exp\left(\frac{-n_1 k^2 \epsilon^2}{2}\right) \\ 471 \quad + (d+1) \exp\left(\frac{-n_2 (1-k)^2 \epsilon^2}{2}\right) \quad (12)$$

472 The equation above holds for all values of k . Hence, to get the tightest
473 upper bound of the left hand side (l.h.s.) of the above equation, we
474 minimize the right hand side (r.h.s.) with respect to k . Here, we note,
475 and also done in [7], the r.h.s. is approximately minimized when the
476 exponents of the two terms are equal:

$$477 \quad k^2 \epsilon^2 n_1 = (1-k)^2 \epsilon^2 n_2 \quad (13)$$

$$478 \quad \Rightarrow \quad k = \sqrt{(n_2/n_1)/(1 + \sqrt{(n_2/n_1)})} \quad (14)$$

479 For this value of k , we have:

$$480 \quad k^2 \epsilon^2 n_1 = (1-k)^2 \epsilon^2 n_2 = \frac{n_2 n_1}{(\sqrt{n_1} + \sqrt{n_2})^2} \epsilon^2 \quad (15)$$

$$481 \quad \leq \frac{n_2 n_1}{(n_1 + n_2)} \epsilon^2 = \frac{m}{2} \epsilon^2, \quad (16)$$

482 where m is the harmonic mean of n_1 and n_2 . We can use the values
483 to get the tightest upper bound for the l.h.s. of Eq. (12) as:

$$484 \quad \mathbb{P}(\|\hat{\mu}_1 - \hat{\mu}_2\|_2 \geq \epsilon) \leq 2(d+1) \exp\left(\frac{-m \epsilon^2}{4}\right) \quad (17)$$

485 Hence proved. \square

502 **Calculating ϵ_{cut} :** As noted in Eq. (6), and the accompanying dis-
503 cussion, we would like to choose a value of ϵ which enables us to
504 declare a concept drift and hence the segment boundary if the ℓ_2
505 norm of the difference of the observed means of the two segments
506 goes beyond ϵ . Further, the hypothesis testing framework allows us
507 to choose a value of ϵ according to the threshold of accepting the
508 hypothesis δ , which bounds the error rate for declaring incorrect
509 segment boundaries to δ . Since Lemma 3.3 bounds the probability
510 of difference of observed means exceeding ϵ , we can use it to choose
511 a value of ϵ (denoted as ϵ_{cut} hereon) such that we get the desired
512 upper bound on declaring the false boundary:

$$513 \quad 2(d+1) \exp\left(\frac{-m \epsilon_{cut}^2}{4}\right) \leq \frac{\delta}{n} \quad (18)$$

$$514 \quad \Rightarrow \quad \epsilon_{cut} \geq \sqrt{\frac{4}{m} \log\left(\frac{2n(d+1)}{\delta}\right)} \quad (19)$$

3.3 Handling Conditionally Dependent Data

523 It may be noted that the derivation of ϵ_{cut} , working backward from
524 the error bound depends upon the Hoeffding's bound. This makes the
525 derivation valid only when the data is identically and independently
526 distributed (i.i.d.). The assumption does not hold for egocentric video
527 segmentation problem when the temporal data coming from video
528 frames is highly correlated with its temporal neighbor. One way to
529 resolve the problem is by making the data conditionally independent.
530 We observe that the correlation between the frames decreases as the
531 temporal distance between them increases. We fix a threshold and
532 declare two frames independent if the correlation coefficient between
533 them is below the threshold. This is effectively sub-sampling the
534 video.

535 We discover the optimal sub-sampling rate from the data itself.
536 For the first frame t in a given window W , we find the frame $t + k_t$
537 for which the correlation coefficient is less than a threshold ρ_c . The
538 process is then repeated from frame $t = t + k_t$, and is continued until
539 the end of the window is reached. We select the sub-sampling rate,
540 k , as the average of k_t for all t .

541 We further optimize the proposed pipeline by observing that we
542 do not really need to sub-sample the video, but the effect of sub-
543 sampling can be incorporated in the threshold ϵ_{cut} itself. Consider
544 an extreme scenario, when the original samples were conditionally
545 independent, but we introduced a severe correlation by duplicating
546 a sample r times. Note that in this case, the ground truth boundary
547 should not shift but the length of the segments W_0 and W_1 just
548 increases by r times. The harmonic mean m also increases by r
549 times, thus effectively decreasing segmentation threshold ϵ_{cut} , and
550 leading to over-segmentation. We compensate for the reduction in
551 ϵ_{cut} by updating the expression to:

$$552 \quad \epsilon_{cut} \geq \sqrt{\frac{4k}{m} \log\left(\frac{2n(d+1)}{k\delta}\right)} \quad (20)$$

553 where k is the sub-sampling rate for un-correlating the input data, as
554 described earlier. Note that the exact choice of correct k is not very
555 critical, but merely helps to virtually sub-sample a video such that
556 the i.i.d. assumption starts to holds, by penalizing the effect to ϵ_{cut} .
557 However, the role of k becomes more important to normalize videos
558 taken at different temporal resolutions (frames per second). The
559 proposed approach avoids over-segmentation of a video by adjusting
560 the threshold for videos at the higher temporal resolution, leading
561 to higher accuracy in boundary prediction. Note that the discussion
562 above does not address the problems when videos are captured at
563 extremely low temporal solution, which we discuss next.

3.4 Handling Photo-stream Data

571 Imagine we had a video, and have found an optimal sub-sampling
572 rate k at which the adjacent frames become conditionally indepen-
573 dent. Note that, any larger k will also satisfy the independence
574 constraint, but will lead to under-segmentation. We observe that
575 when the input is a photo-stream, the frames are indeed condition-
576 ally independent, but they would likely be independent (as per our
577 correlation coefficient criterion) even when we insert an additional
578 frame (by interpolating neighboring frames) in between. We believe
579 that our method underestimates the length of the segment in the case



Figure 3: The segmentation granularity increases as we increase δ in our approach. The three rows in the figure show the output from our approach at $\delta = 10^{-6}$, 10^{-4} , and 10^{-2} respectively, on the ‘Alireza Day 1’ sequence from Disney dataset. The bars above each row indicates the time instance of frames chosen as a boundary, such that the length of the row shows the length of the sequence.

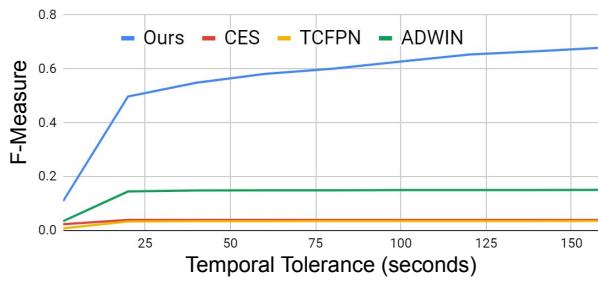


Figure 4: The figure shows the F-Measure comparison between SOTA and proposed approach for different values of temporal tolerance for the Disney dataset.

of photo-streams due to the above reason. Therefore, for the photo-streams, we suggest to look for the smallest number of k frames, which when inserted in the photo-stream still keeps the neighboring frames independent. We introduce these frames, or feature vectors as the case may be, by simply averaging the features of two consecutive frames. The process is continued until the correlation coefficient of the feature vectors remains below a user specified threshold.

However, similar to the way we handled correlated frames in the videos, we do not need to make the actual addition of frames to the dataset. We just need to know the length of the adaptive window, when the frames will be added to the window. This new window length is then used to modify the threshold. The modified threshold used for the photo-stream is as follows:

$$\epsilon_{\text{cut}} \geq \sqrt{\frac{4}{mk} \log \left(\frac{2nk(d+1)}{\delta} \right)} \quad (21)$$

Fig. 2 shows the block diagram of the proposed approach and Algorithm 1 in the supplementary material presents the pseudo-code.

4 EXPERIMENTS

We demonstrate the results of proposed approach on three extremely long egocentric video datasets, viz HUJI [37, 38], Disney [19], and UTEgo [30, 33], as well as on the standard photo-stream dataset, viz EDUB-Seg20 [14, 44]. We give a detail description of the datasets in the supplementary material. The proposed technique is implemented

on Matlab with system architecture comprising of Quadro P5000 GPU and Intel i7 processor with 4 cores (32 GB RAM). It takes approximately 2 hrs (inclusive of feature extraction) and approximately 8GB CPU RAM to segment 8 hrs long video.

4.1 Implementation Details

Feature Vector: For all the video datasets, we use the input at 5fps and use frame-wise AlexNet [28] features as used by SR-Clustering [14]. However, for a fair comparison on the photo-stream datasets, we use LSTM features similar to one used by [12]. However, since we operate in the streaming mode in our application, instead of bi-directional features as suggested in [12], we use only unidirectional features.

Frame Correlation Coefficient: As discussed earlier, to make the frames independent for meeting the requirements of our theoretical results, we use the notion of skip factor. The learned skip factor requires a hyper-parameter correlation coefficient threshold, ρ_c to declare the two frames independent. We have chosen $\rho_c = 0.95$ for video datasets. However, we observe that LSTM features used for the photo-stream datasets exhibit a high correlation. Hence, we use $\rho_c = 0.999$ for the photo-stream datasets.

Granularity: Any segmentation problem is inherently dependent upon the scale one is looking for. In our technique, the granularity at which the user wants their video to be segmented can be controlled by the δ . As seen in Fig 3, as the value of δ increases, the number of segments increases, and boundaries are detected even for smaller changes. Similarly, upon decreasing the value of δ , the number of segments decreases, corresponding to capturing large heterogeneous context in a single event. In general application of our technique, we expect that such a granularity could be taken as feedback from the end-user. However, for comparing with benchmark datasets, we do not have such user-feedback available. Hence we use average segment length as the proxy for the segmentation granularity required. We define 2500-3000, 1600-2500, and 1000-1600 frames per segment as our ranges for low, medium, and high levels of granularity respectively. We set the δ for the corresponding granularity as 10^{-2} , 10^{-4} , and 10^{-6} respectively. Similarly, for the photo-stream datasets

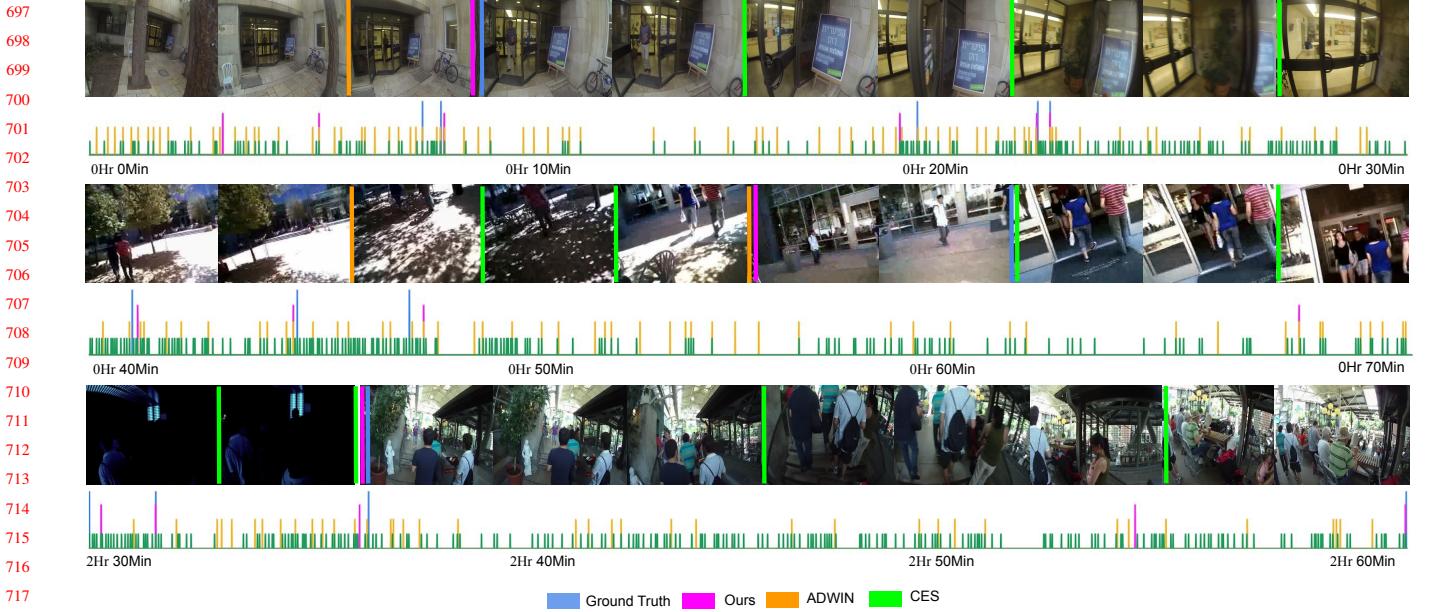


Figure 5: Temporal segmentation of long egocentric videos: The figure shows a qualitative representation of closeness of boundaries predicted by the proposed approach, ADWIN [7], CES [12] to ground truth boundaries from specific portions of Huji (first row), UTEgo (second row) and Disney (third row) datasets (better visualize in colors). Please see the text for details.

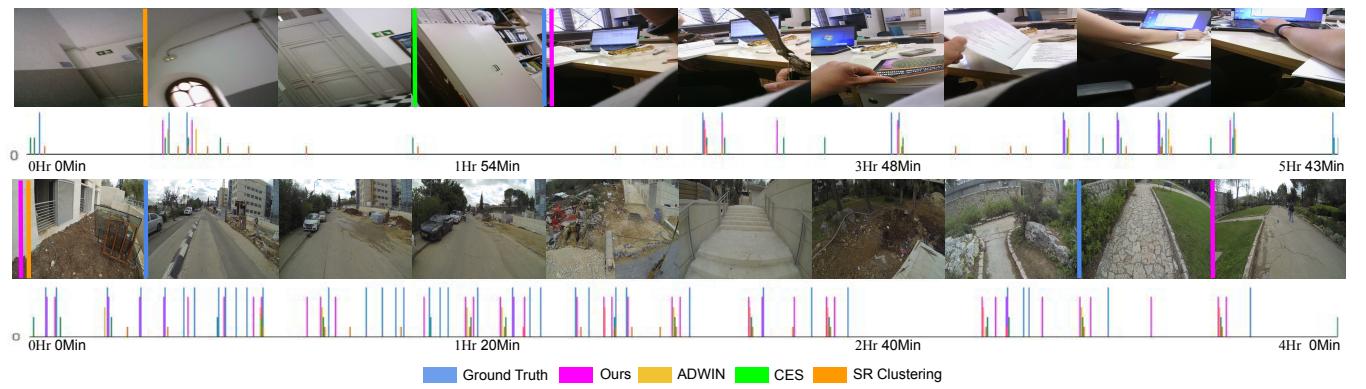


Figure 6: Temporal segmentation of photo-stream data: The figure shows a qualitative representation of the closeness of boundaries predicted by the proposed approach, ADWIN [7], CES [12] to ground truth boundaries from specific portions of EDUB-Seg (first row), and Huji (second row). Please see the text for details.

of HUJI, UTEgo, Disney, and EDUB-Seg20, we experiment with δ values of 10^{-1} , 10^{-3} , and 10^{-7} for different levels of granularity.

Boundary Tolerance: As proposed by [12], when dealing with continuous boundaries in an egocentric video, there is an inherent ambiguity in annotating the exact frame which should be marked as the boundary, and many frames in the temporal vicinity could have been marked as a boundary as well. Hence, penalizing an algorithm for marking the exact frame as a boundary may not indicate the true strength of the technique. [12] has proposed the use of temporal tolerance, which allows a technique to be rewarded if it predicts a boundary within a certain range of the ground truth. We adopt the

metric in our experiments and use a temporal tolerance (tol) of 2.5 minutes to calculate the performance (f-measure) of our technique. As expected, and as shown in Fig. 4, the boundary detection accuracy improves as the value of temporal tolerance is increased.

4.2 Evaluation Measure

We use the averaged F-measure to evaluate our performance. As proposed in [12], we consider a predicted boundary as true positive if it occurs within the tolerance(tol) neighborhood of a ground truth boundary, while taking into consideration that this ground truth boundary has not already been matched to a predicted boundary

755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811

	Methods	HUJI	UTEgo	Disney
813	TCFPN [17]	4.18	2.50	3.56
814	ADWIN [7]	12.44	0.83	15.01
815	CES[12]	4.52	9.31	3.96
816				
817	Ours	73.01	58.41	67.63
818				
819				
820				

821 **Table 2: Comparing F-Measure performance of various
822 approaches on video datasets**

Datasets	High	Medium	Low	
826	UTEgo	2m19s	3m08s	3m29s
827	Disney	1m77s	2m54s	3m87s
828				
829				

831 **Table 4: Average latency of the boundary detection during on-
832 line mode running of our technique under different temporal
833 granularity. Please see the text for details.**

836 before. Analogously, all the ground truth boundaries, for which no
837 frame within its tol range has been predicted, are referred to as
838 false negative. We also evaluate our method based on the number of
839 segments predicted. This metric is used to demonstrate the reduc-
840 tion in over-segmentation achieved for video data upon using our
841 methodology.

4.3 Comparative Evaluation

845 For comparison on video datasets, we pick two representative tech-
846 niques to compare against, viz CES [12] and TCFPN [17]. We
847 also compare against ADWIN [7] which is based on unsupervised
848 concept drift detection but does not handle multivariate data or cor-
849 related samples. For comparison with ADWIN, we pretend the data
850 is uncorrelated and convert a feature vector into a single scalar by
851 taking its ℓ_2 norm. We ignore the SR-Clustering [14] for the video
852 datasets because it doesn't scale for day-long video sequences.

853 Since many of the approaches we compare against were origi-
854 nally targeted for photo-streams and not videos, therefore, to ensure
855 a fair comparison, we prepare two configurations for each dataset. In
856 the first configuration, we resample a video at 2 frames per minute,
857 thereby making it resemble a photo-stream. In the second config-
858 uration, each input video is resampled at 5fps to match the lowest
859 temporal resolution of all the datasets. For photo-stream datasets, we
860 also compare with SR-Clustering [14]. Table 2 shows the quantita-
861 tive evaluation based on F-measure for $tol = 750$ for video datasets.
862 We notice significant performance improvement over all the state
863 of the art approaches as these techniques fail to handle the daylong
864 video sequences.

865 Fig 5 shows a qualitative visualization of the comparison be-
866 tween various state of the art techniques and the proposed approach.
867 The bar chart shows the frames selected as a boundary by different
868 techniques for a 30 minutes clip. It is clear that the state of the art
869 techniques severely over-segment all the video sequence datasets

870

	Methods	Features	EDUB	HUJI	UTEgo	Disney
871	TCFPN [17]	CNN	19.26	2.37	1.37	3.84
872	ADWIN [7]	CNN	35.37	44	11.47	23.21
873	CES [12]	LSTM-Bi	69	45.70	36.19	61.40
874	SR-Clustering [14]	CNN	49.93	44.06	9.44	55.81
875						
876	Ours	LSTM-uni	63.96	59.44	60.61	68.83
877						
878						

879 **Table 3: Comparing F-Measure performance of various approaches on
880 photo-stream datasets**

881 due to frequent scene changes accompanying the sharp head motion
882 of the wearer. The images above each of the bar charts show
883 representative frames from a short video segment from each of the
884 clips. The boundaries selected by each technique are marked by thick
885 colored lines between the frames. This is for visual comparison of
886 the frames where different techniques choose to create a boundary.
887 From the figure, we can observe that the proposed approach doesn't
888 over-segment and precisely locates the temporal boundaries.

889 Table 3 shows the F-measure for $tol = 5$ for photo-stream datasets
890 (EDUB-Seg as well as all the video datasets down-sampled to photo-
891 streams as described earlier). For photo-stream datasets also we
892 show considerable improvement. We report 13.74%, 24.42%, and
893 7.43% improvement in F-measure for HUJI, UTEgo, and Disney
894 datasets respectively, however, for EDUB-Seg20 we under-perform
895 marginally as CES [12] uses bidirectional features, whereas we
896 use uni-directional features to maintain the online streaming mode
897 property of our technique. Fig 6 shows the visualization for photo-
898 stream datasets. The first row shows the visualization for the EDUB-
899 Seg20 dataset where the CES [12] performs competitively. For the
900 HUJI dataset proposed method performs better than the CES [12].

4.4 Online Streaming vs Recorded Video

901 Our algorithm can be potentially used in the online streaming mode
902 as well. Recall that for detecting a temporal boundary, we take a
903 window w , split it at time instant t , in two windows w_1 and w_2 , and
904 then find the difference of means. Therefore, we effectively find the
905 temporal boundary at t after looking at w_2 as well. This can be seen
906 as detecting a boundary with a certain latency. Table 4 shows the
907 average latency of our algorithm vs the average segment length in
908 the video.

5 CONCLUSION

909 In this paper, we have introduced a novel, principled, and theoreti-
910 cally justified technique for temporal segmentation of egocentric
911 videos. We have adapted the univariate concept drift for i.i.d. data to
912 multivariate correlated data using the adaptive windowing technique.
913 We demonstrate the results on long videos as well as photo-stream
914 datasets to prove the efficacy of the proposed approach. We have
915 also shown that the adaptive windowing technique can generate su-
916 perior results in video temporal segmentation when compared to the
917 state-of-the-art deep CNN/LSTM models.

918
919
920
921
922
923
924
925
926
927
928

REFERENCES

- [1] Sathyaranayanan N Aakur and Sudeep Sarkar. 2019. A Perceptual Prediction Framework for Self Supervised Event Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1197–1206.
- [2] Maedeh Aghaei. 2017. Social signal extraction from egocentric photo-streams. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*. ACM, 656–659.
- [3] Maedeh Aghaei, Mariella Dimiccoli, Cristian Canton Ferrer, and Petia Radeva. 2018. Towards social pattern characterization in egocentric photo-streams. *Computer Vision and Image Understanding* 171 (2018), 104–117.
- [4] Maedeh Aghaei, Mariella Dimiccoli, and Petia Radeva. 2017. All the people around me: face discovery in egocentric photo-streams. In *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 1342–1346.
- [5] Humam Alwassel, Fabian Caba Heilbron, and Bernard Ghanem. 2018. Action search: Spotting actions in videos and its application to temporal action localization. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 251–266.
- [6] Bharat Lal Bhatnagar, Suriya Singh, Chetan Arora, CV Jawahar, and KCIS CVIT. 2017. Unsupervised Learning of Deep Feature Representation for Clustering Egocentric Actions.. In *IJCAI*. 1447–1453.
- [7] Albert Bifet and Ricard Gavaldà. 2007. Learning from time-changing data with adaptive windowing. In *Proceedings of the 2007 SIAM international conference on data mining*. SIAM, 443–448.
- [8] Shyamal Buch, Victor Escorcia, Chuanqi Shen, Bernard Ghanem, and Juan Carlos Niebles. 2017. Sst: Single-stream temporal action proposals. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2911–2920.
- [9] Alejandro Cartas, Mariella Dimiccoli, and Petia Radeva. 2017. Batch-based activity recognition from egocentric photo-streams. In *Proceedings of the IEEE International Conference on Computer Vision*. 2347–2354.
- [10] Alejandro Cartas, Estefania Talavera, Petia Radeva, and Mariella Dimiccoli. 2018. On the Role of Event Boundaries in Egocentric Activity Recognition from Photo-streams. *arXiv preprint arXiv:1809.00402* (2018).
- [11] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A Ross, Jia Deng, and Rahul Sukthankar. 2018. Rethinking the faster r-cnn architecture for temporal action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1130–1139.
- [12] Ana Garcia del Molino, Joo-Hwee Lim, and Ah-Hwee Tan. 2018. Predicting Visual Context for Unsupervised Event Segmentation in Continuous Photo-streams. *arXiv preprint arXiv:1808.02289* (2018).
- [13] Catarina Dias and Mariella Dimiccoli. 2018. Learning Event Representations by Encoding the Temporal Context. In *European Conference on Computer Vision*. Springer, 587–596.
- [14] Mariella Dimiccoli, Marc Bolaños, Estefania Talavera, Maedeh Aghaei, Stavri G Nikolov, and Petia Radeva. 2017. SR-clustering: Semantic regularized clustering for egocentric photo-streams segmentation. *Computer Vision and Image Understanding* 155 (2017), 55–69.
- [15] Li Ding and Chenliang Xu. 2017. Tricornet: A hybrid temporal convolutional and recurrent network for video action segmentation. *arXiv preprint arXiv:1705.07818* (2017).
- [16] Li Ding and Chenliang Xu. 2018. Video Action Segmentation with Hybrid Temporal Networks. (2018).
- [17] Li Ding and Chenliang Xu. 2018. Weakly-supervised action segmentation with iterative soft boundary assignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6508–6516.
- [18] Victor Escorcia, Fabian Caba Heilbron, Juan Carlos Niebles, and Bernard Ghanem. 2016. Daps: Deep action proposals for action understanding. In *European Conference on Computer Vision*. Springer, 768–784.
- [19] Aliriza Fathi, Jessica K Hodgins, and James M Rehg. 2012. Social interactions: A first-person perspective. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 1226–1233.
- [20] Antonino Furnari, Sebastiano Battiatto, and Giovanni Maria Farinella. 2018. Personal-location-based temporal segmentation of egocentric videos for lifelogging applications. *Journal of Visual Communication and Image Representation* 52 (2018), 1–12.
- [21] Pedro Herruzo, Laura Portell, Alberto Soto, and Beatriz Remesero. 2017. Analyzing First-Person Stories Based on Socializing, Eating and Sedentary Patterns. In *International Conference on Image Analysis and Processing*. Springer, 109–119.
- [22] Minh Hoai and Fernando De la Torre. 2014. Max-margin early event detectors. *International Journal of Computer Vision* 107, 2 (2014), 191–202.
- [23] Shao Huang, Weiqiang Wang, Shengfeng He, and Rynson W.H. Lau. 2017. Egocentric Temporal Action Proposals. *IEEE Transactions on Image Processing* (11 2017), 1–1.
- [24] Mihir Jain, Jan Van Gemert, Hervé Jégou, Patrick Bouthemy, and Cees GM Snoek. 2014. Action localization with tubelets from motion. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 740–747.
- [25] Kris M Kitani, Takahiro Okabe, Yoichi Sato, and Akihiro Sugimoto. 2011. Fast unsupervised ego-action learning for first-person sports videos. In *CVPR 2011*. IEEE, 3241–3248.
- [26] Adarsh Kowdle and Tsuhan Chen. 2012. Learning to Segment a Video to Clips Based on Scene and Camera Motion. In *ECCV*.
- [27] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017. Dense-captioning events in videos. In *Proceedings of the IEEE International Conference on Computer Vision*. 706–715.
- [28] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.
- [29] Colin Lea, Michael D Flynn, Rene Vidal, Austin Reiter, and Gregory D Hager. 2017. Temporal convolutional networks for action segmentation and detection. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 156–165.
- [30] Yong Jae Lee, Joydeep Ghosh, and Kristen Grauman. 2012. Discovering important people and objects for egocentric video summarization. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 1346–1353.
- [31] Shuai Li, Wanqing Li, Chris Cook, Ce Zhu, and Yanbo Gao. 2018. Independently recurrent neural network (indrnn): Building a longer and deeper rnn. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5457–5466.
- [32] Qinying Liu and Zilei Wang. [n.d.]. Progressive Boundary Refinement Network for Temporal Action Detection. ([n. d.]).
- [33] Zheng Lu and Kristen Grauman. 2013. Story-driven summarization for egocentric video. In *CVPR*.
- [34] Lester Mackey, Michael I Jordan, Richard Y Chen, Brendan Farrell, Joel A Tropp, et al. 2014. Matrix concentration inequalities via the method of exchangeable pairs. *The Annals of Probability* (2014), 906–945.
- [35] Behrooz Mahasseni, Xiaodong Yang, Pavlo Molchanov, and Jan Kautz. 2017. Budget-aware activity detection with a recurrent policy network. *arXiv preprint arXiv:1712.00097* (2017).
- [36] Francesco Paci, Lorenzo Baraldi, Giuseppe Serra, Rita Cucchiara, and Luca Benini. 2016. Context change detection for an ultra-low power low-resolution ego-vision imager. In *European Conference on Computer Vision*. Springer, 589–602.
- [37] Yair Peleg, Chetan Arora, and Shmuel Peleg. 2014. Temporal segmentation of egocentric videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2537–2544.
- [38] Yair Peleg, Ariel Ephrat, Shmuel Peleg, and Chetan Arora. 2016. Compact cnn for indexing egocentric videos. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 1–9.
- [39] Md Mostafa Kamal Sarker, Hatem A Rashwan, Estefania Talavera, Syeda Furruka Banu, Petia Radeva, and Domènec Puig. 2018. MACNet: Multi-scale Atrous Convolution Networks for Food Places Classification in Egocentric Photo-Streams. In *European Conference on Computer Vision*. Springer, 423–433.
- [40] Amr Sharaf, Marwan Torki, Mohamed E Hussein, and Motaz El-Saban. 2015. Real-time multi-scale action detection from 3d skeleton data. In *2015 IEEE Winter Conference on Applications of Computer Vision*. IEEE, 998–1005.
- [41] Zheng Shou, Dongang Wang, and Shih-Fu Chang. 2016. Temporal action localization in untrimmed videos via multi-stage cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1049–1058.
- [42] Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, and Paolo Remagnino. 2018. Superframes, A Temporal Video Segmentation. *2018 24th International Conference on Pattern Recognition (ICPR)* (2018), 566–571.
- [43] Gilbert W Stewart. 1990. Matrix perturbation theory. (1990).
- [44] Estefania Talavera, Mariella Dimiccoli, Marc Bolaños, Maedeh Aghaei, and Petia Radeva. 2015. R-clustering for egocentric video segmentation. In *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, 327–336.
- [45] Limin Wang, Yu Qiao, and Xiaou Tang. 2014. Action recognition and detection by combining motion and appearance features. *THUMOS14 Action Recognition Challenge* 1, 2 (2014), 2.
- [46] Bo Xiong, Gunhee Kim, and Leonid Sigal. 2015. Storyline representation of egocentric videos with an applications to story-based search. In *Proceedings of the IEEE International Conference on Computer Vision*. 4525–4533.
- [47] Yun Zhai and Mubarak Shah. 2005. A general framework for temporal video scene segmentation. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. 1111–1116.

929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044