Story-Driven Summarization for Egocentric Video

Zheng Lu and Kristen Grauman
University of Texas at Austin
luzheng@cs.utexas.edu, grauman@cs.utexas.edu

Abstract

We present a video summarization approach that discovers the story of an egocentric video. Given a long input video, our method selects a short chain of video subshots depicting the essential events. Inspired by work in text analysis that links news articles over time, we define a random-walk based metric of influence between subshots that reflects how visual objects contribute to the progression of events. Using this influence metric, we define an objective for the optimal k-subshot summary. Whereas traditional methods optimize a summary's diversity or representativeness, ours explicitly accounts for how one sub-event "leads to" another—which, critically, captures event connectivity beyond simple object co-occurrence. As a result, our summaries provide a better sense of story. We apply our approach to over 12 hours of daily activity video taken from 23 unique camera wearers, and systematically evaluate its quality compared to multiple baselines with 34 human subjects.

1. Introduction

Digital video recorders and media storage continue to decrease in cost, while usage continues to climb. Much of the data consists of long-running, unedited content—for example, surveillance feeds, home videos, or video dumps from a camera worn by a human or robot. There is information in the data, yet most of it cannot possibly be reviewed in detail. Thus, there is a clear need for systems that assist users in accessing long videos. *Video summarization* addresses this need by producing a compact version of a full length video, ideally encapsulating its most informative parts. The resulting summaries can be used to enhance video browsing, or to aid activity recognition algorithms.

Summarization methods compress the video by selecting a series of keyframes [26, 27, 10, 16] or subshots [19, 13, 18, 6] that best represent the original input. Current methods use selection criteria based on factors like diversity (selected frames should not be redundant), anomalies (unusual events ought to be included), and temporal spacing (coverage ought to be spread across the video). Typically they rely on low-level cues such as motion or color [26, 19, 13], or else track pre-trained objects of interest [16, 6, 14]. Overall, existing methods offer sophisticated ways to sample from the original video, reducing the time required for a human to view the contents.

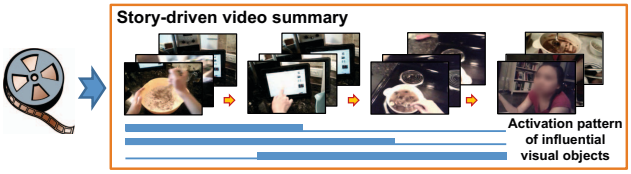

Figure 1. Our method generates a story-driven summary from an unedited egocentric video. A good story is defined as a coherent chain of video subshots in which each subshot influences the next through some (active) subset of influential visual objects.

However, we contend that defining video summarization as a *sampling* problem is much too limiting. In particular, traditional approaches fail to account for *how the events in the video progress from one to another.* As a result, they may omit important (but short) sub-events, yet include redundant (though visually diverse) ones that do not impact the overall narrative. While a problem for any video source, this limitation is especially pronounced for egocentric video summarization. Egocentric video captured with a wearable camera is long and unstructured, and its continuous nature yields no evident shot boundaries; yet, the raw data inherently *should* tell a story—that of the camera wearer's day.

Our goal is to create *story-driven* summaries for long, unedited videos. What makes a good visual story? Beyond displaying important persons, objects, and scenes, it must also convey how one thing "leads to" the next. Specifically, we define a good story as a coherent chain of video subshots[1] in which each subshot influences the next through some subset of key visual objects.

Critically, *influence* is distinct from *inclusion*. For example, in the "story" of visiting the bookstore, a book plays an important role in linking the actions of browsing the shelves

[1]Throughout, we use *subshot* and *keyframe* interchangeably; the proposed method can produce summaries based on either unit.

and seeing the cashier; in contrast, a book on the coffee table at home—though present when both watching TV and then greeting a friend—is not influential to the progress of those events. Therefore, simply counting object overlap between two sub-events will not reveal their connections in the story. Instead, inspired by work in text analysis that links news articles over time given their words [24], we devise a metric to quantify how connected two sub-events are given their objects. The metric builds a bipartite graph between subshots and objects, and then scores the impact each object has on the stationary probability for a random walk starting from one of the subshots and ending in another.

Our approach works as follows. First, we segment the input video into subshots using a novel *static-transit* grouping procedure well-suited for unstructured egocentric video. Then, we detect which entities appear in each subshot, where an entity is either some familiar object category (phone, toothbrush, etc.) or else an object-like visual word region, depending on whether previously trained object models are available. Next, for each subshot, we estimate its individual importance as well as its influence on every *other* subshot in the original sequence, given their objects/words. Finally, we optimize an energy function that scores a candidate chain of k selected subshots according to how well it preserves both influence over time and individually important events. To compose the final summary, we devise a simple method to select the best k per broad event given the neighboring events.

Contributions Our main contribution is the idea of *story-driven* video summarization; to our knowledge, ours is the first summarization work to explicitly model the influence between sub-events. To accomplish this, our technical contributions are: 1) we adapt a text analysis technique that connects news articles [24] to the visual domain, transferring the problem finding a path through chronological documents to one finding a chain of video subshots that conveys a fluid story; 2) we show how to estimate the influence of one visual event on another given their respective objects; 3) we introduce a novel temporal segmentation method uniquely designed for egocentric videos; 4) we show how to exploit our influence estimates to discover a video's most influential objects; and 5) we perform a large-scale, systematic evaluation to compare the proposed approach to several baselines on 12 hours of challenging egocentric video. Our user study results confirm that our method produces summaries with a much better sense of story.

2. Related Work

We review prior work in video summarization, egocentric video analysis, and influence discovery in text mining.

Video summarization Keyframe-based methods select a sequence of keyframes to form a summary, and typically use low-level features like optical flow [26] or image differences [27]. Recent work also uses high-level information such as object tracks [16] or "important" objects [14], or takes user input to generate a storyboard [10].

In contrast, video skimming techniques first segment the input into subshots using shot boundary detection. The summary then consists of a selected set of representative subshots. Features used for subshot selection include motion-based attention [19], motion activity [18], or spatio-temporal features [13]. User interaction can help guide subshot selection; for example, the user could point out a few interesting subshots [6], or provide keyframes for locations in a map-based storyboard [21]. For static cameras, dynamic summaries simultaneously show multiple actions from different timepoints in the video, all overlaid on the same background [22].

Both types of methods mostly focus on selecting good individual frames or shots, ignoring the relationship between them. In contrast, our approach models the influence between subshots, which we show is vital to capture the story in the original video. Prior methods that do account for inter-keyframe relationships restrict the criterion to low-level cues and pairwise terms [17, 27, 14], which can lead to an unintended "toggling", where the summary includes redundant views at every other frame. In contrast, our model uses higher-order constraints enforcing that objects enter and leave the summary with some coherent structure.

Egocentric video analysis Due to the small form factor of today's egocentric cameras, as well as expanding application areas, vision researchers are actively exploring egocentric video analysis. Recent work uses supervised learning to recognize activities [25, 7, 20, 9, 8], handled objects [23], and novel events [1]. Unsupervised methods include scene discovery [11], sports action recognition [12], keyframe selection [5], and summarization [14]. Unlike any prior work, we aim to recover a story-driven summary, and we explicitly capture shot-to-shot influence.

Influence in news articles Both our influence metric as well as the search strategy we use to find good chains are directly inspired by recent work in text mining [24]. Given a start and end news article, that system extracts a coherent chain of articles connecting them. For example, the method could try to explain how the decline in home prices in 2007 led to the health care debate in 2009. Adapting their model of influence to video requires defining analogies for documents and words. For the former, we develop a novel subshot segmentation method for egocentric data; for the latter, we explore both category-specific and category-independent models of visual objects. Finally, we find that compared to news articles, egocentric video contains substantial redundancy, and subshot quality varies greatly.

Thus, whereas the model in [24] scores only the influence of selected documents, we also model chain quality in terms of predicted importance and scene diversity.

3. Approach

Our approach takes a long video as input and returns a short video summary as output. First, we segment the original video into a series of n subshots, $\mathcal{V} = \{s_1, \ldots, s_n\}$. For each subshot, we extract the set of objects or visual words appearing in it. We define our novel segmentation method and object extraction procedure in Sec. 3.1.

Consider the subshots as nodes in a 1D chain, and let $S = \{s_{k_1}, \ldots, s_{k_K}\}$, $S \subset \mathcal{V}$ denote an order-preserving chain of K selected nodes. Our goal is to select the optimal K-node chain S^*:

$$S^* = \arg\max_{S \subset \mathcal{V}} \; Q(S), \tag{1}$$

where $Q(S)$ is a three-part quality objective function

$$Q(S) = \lambda_s \, \mathcal{S}(S) + \lambda_i \, \mathcal{I}(S) + \lambda_d \, \mathcal{D}(S) \tag{2}$$

that reflects the *story*, *importance*, and *diversity* captured in the selected subshots, and the constants λ weight their respective influence. We define each component in turn in Sec. 3.2. For now, suppose the length K is user-specified; we will return to the issue of how to select K below.

To optimize the objective, we use an efficient priority queue approach to search for good candidate chains (Sec. 3.3). Finally, we compose the final summary by selecting and linking together a "chain of chains" computed from multiple broad chunks of the source video (Sec. 3.4).

3.1. Egocentric Subshot Representation

Subshot extraction is especially challenging for egocentric video. Whereas traditional approaches rely on shot boundary detection (e.g., detecting an abrupt change to the color histogram), egocentric videos are continuous. They offer no such dramatic cues about where the scene or activity has changed. Thus, we introduce a novel subshot segmentation approach tailored to egocentric data. Our key insight is to detect generic categories of ego-activity that typically align with sub-events. Specifically, we learn to predict whether the camera wearer is *static*, meaning not undergoing significant body or head motion, *in transit*, meaning physically traveling from one point to another, or *moving the head*, meaning changing his attention to different parts of the scene. We manually label 4577 total training frames from various videos.

To represent each frame, we extract features based on optical flow and blurriness, which we expect to characterize the three classes of interest. For example, flow vectors emanating from the center of the frame are indicative of forward travel (e.g., walking down the hall), while motion blur occurs when the camera wearer moves his head quickly (e.g., to pick up a pot). Specifically, we compute dense optical flow [15], and quantize the flow angles and magnitudes into eight bins. Then we form a histogram of flow angles weighed by their magnitude, concatenated with a histogram of magnitudes. To compute blur features, we divide the frame into a 3×3 grid and score each cell by its blurriness, using [4]. We train one-vs.-rest SVM classifiers to distinguish the three classes.

Given a novel input video, we first apply the classifier to estimate class likelihoods for each frame. Then, we smooth the labels using a Markov random field (MRF), in which each frame is connected to its neighbors within a temporal window of 11 frames. The unary potentials are the class likelihoods, and the pairwise potential is a standard Ising model where consecutive frames receiving different labels are penalized according to the similarity of their color histograms. The resulting smoothed labels define the subshots: each sequence of consecutive frames with the same label belongs to the same subshot. Thus, the number of subshots n will vary per video; in our data (described below) a typical subshot lasts 15 seconds and a typical 4-hour video has $n = 960$ total subshots. This yields an "oversegmentation" of the video, which is useful for later processing.

We represent a subshot s_i in terms of the visual objects that appear within it. We extract objects in one of two ways. For videos coming from a *known environment* in which models can be pre-trained for the primary objects of interest, we use a bank of object detectors, and record all confidently detected objects. For example, for egocentric video capturing daily living activities in the living room and kitchen [20], the object bank could naturally consist of household objects like fridge, mug, couch, etc. On the other hand, for videos coming from a more *uncontrolled setting*, a preset bank of object detectors is insufficient. In this case, we take an unsupervised approach, where the "objects" are visual words created from "object-like" windows. Specifically, we generate 100 object-like windows [2] for each frame and represent each one with a HOG pyramid [3]. Since the windows can vary substantially in aspect ratio, we first quantize them into five aspect ratio groups using k-means. Then, we quantize the HOG descriptors per aspect ratio into 200 visual words, yielding 1000 total visual words. We find that with this procedure the same visual word often represents the same object.

In the following, let $O = \{o_1, \ldots, o_N\}$ denote the set of N possible objects; that is, o_i refers to one of N detectable object categories or else one of $N = 1000$ discovered visual words. We will use *object* interchangeably to mean a "true" object or a visual word.

3.2. Scoring a Candidate Chain of Subshots

Now we define each of the three terms of the scoring function in Eqn. 2.

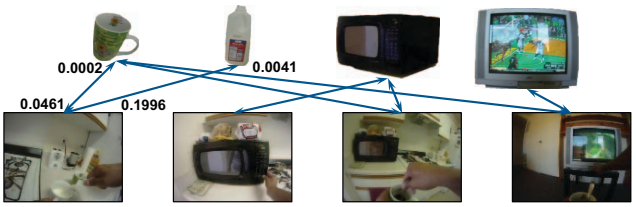

Figure 2. Sketch of the bipartite graph for influence calculation. Top row: object nodes, bottom row: subshot nodes. Edges denote that an object appears in the subshot, and probabilities on the directed edges reflect their association. (Not all are shown.)

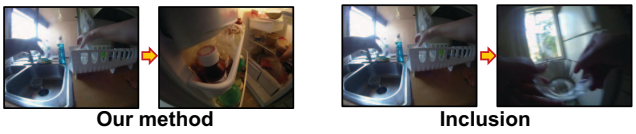

Figure 3. Illustration of the effect of influence vs. inclusion. In the story of making cereal, our influence measure can capture *grabbing a dish* leading to *fetching the milk* (left). In contrast, an object inclusion metric cannot discover this connection, since the subevents share no objects (right).

Story progress between subshots The first term $\mathcal{S}(S)$ captures the element of *story*, and is most crucial to the novelty of our approach. We say a selected chain S tells a good story if it consists of *a coherent chain of visual objects, where each strongly influences the next in sequence*. The *influence* criterion means that for any pair of subshots selected in sequence, the objects in the first one "lead to" those in the second. The *coherency* criterion tempers this objective, favoring chains where objects do not drop in and out repeatedly; this aspect helps avoid chains that either a) toggle between related objects or b) satisfy the weakest link objective below by repeating objects in $n-1$ subshots followed by a big "hop" to some unrelated content. In the following, our definitions for influence and coherency are directly adapted from [24], where we draw an analogy between the news articles and words in that work, and the subshots and visual objects in our work.

Suppose we were considering influence alone for the story term $\mathcal{S}(S)$. Then we'd want the chain that maximizes:

$$\mathcal{S}'(S) = \min_{j=1,\ldots,K-1} \sum_{o_i \in O} \textsc{Influence}(s_j, s_{j+1}|o_i), \quad (3)$$

that is, the chain whose weakest link is as strong as possible.

To compute the influence between two subshots requires more than simply counting their shared objects, as discussed above. To capture this notion, we use a random-walk approach to score influence conditioned on each object. We construct a bipartite directed graph $G = (V_s \cup V_o, E)$ connecting subshots and objects. The vertices V_s and V_o correspond to the subshots and objects, respectively. For every object o that appears in subshot s, we add both the edges (o, s) and (s, o) to E. The edges have weights based on the association between the subshot and object; we define the weight to be the frequency with which the object occurs in that subshot, scaled by its predicted egocentric importance, using [14]. We normalize the edge weights over all objects/subshots to form probabilities. See Figure 2.

Intuitively, two subshots are highly connected if a random walk on the graph starting at the first subshot vertex frequently reaches the second one. Furthermore, the object o plays an important role in their connection to the extent that walks through that object's vertex lead to the second subshot. Using this idea, we measure influence in terms of the difference in stationary distributions for two variants of the graph G. Let $P(u|v)$ denote the probability of reaching vertex v from vertex u (as recorded by the edge weight on (u, v)). The chance of a random walk starting from s_i being at any node v is given by the stationary distribution:

$$\prod\nolimits_i(v) = \varepsilon \cdot \mathbb{1}(v = s_i) + (1-\varepsilon) \sum\nolimits_{(u,v)\in E} \prod\nolimits_i(u) P(v|u), \quad (4)$$

where ε is the random restart probability (defined later). Let $\prod_i^o(v)$ be computed the same way, but over a modified graph where object o is made to be a sink node, with no outgoing edges. The influence is computed as the difference between the two stationary probabilities at s_j:

$$\textsc{Influence}(s_i, s_j|o) = \prod\nolimits_i(s_j) - \prod\nolimits_i^o(s_j). \quad (5)$$

Intuitively, the score is high if object o is key to the influence of subshot s_i on s_j—that is, if its removal would cause s_j to no longer be reachable from s_i. As desired, this metric of influence captures relationships between subshots even when they do not share objects. For example, in the "story" of making cereal, taking a dish from the plate holder leads to grabbing the milk from the fridge. Our influence measure can capture these two events' ties, whereas a metric measuring object *inclusion* (e.g., cosine similarity on bag-of-objects) cannot, since they contain no shared objects. Instead, the inclusion measure can only capture links less essential to the story, such as grabbing and holding a dish. See Figure 3.

To account for coherency as well as influence, we also enforce preferences that only a small number of objects be "active" for any given subshot transition, and that their activation patterns be smooth in the summary. This is done by adding an activation variable specifying which objects are active when, yielding the story-based objective term:

$$\mathcal{S}(S) = \max_{\boldsymbol{a}} \min_{j=1,\ldots,K-1} \sum_{o_i \in O} \boldsymbol{a}_{i,j} \textsc{Influence}(s_j, s_{j+1}|o_i), \quad (6)$$

where $\boldsymbol{a}$ is an indicator variable of length $N \times n$ reflecting which objects are active in which subshots, $\boldsymbol{a}_{i,j}$ denotes its value for object i and subshot j, and $\sum_{i,j} \boldsymbol{a}_{i,j}$ is bounded by γ. By relaxing $\boldsymbol{a}$ to take real-valued strengths, the above is formulated as a linear program (see [24] for details).

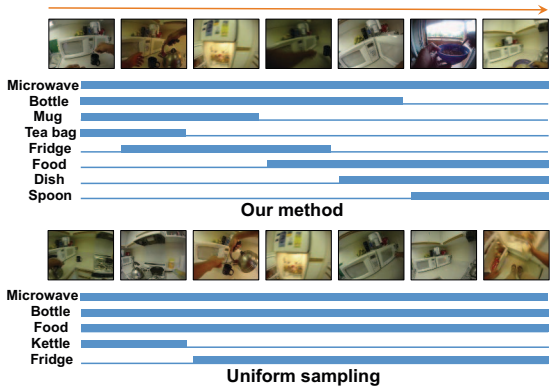

Figure 4. Activation pattern of objects for our summary (top) and a uniform sampling baseline (bottom) for the same input video. Bars indicate the objects activated in the keyframes above them.

Figure 4 shows an example of the activation pattern over a chain of subshots for our method and a baseline that uniformly samples frames throughout the original video. Our result shows how the story progresses though the objects (i.e., making tea, getting food from the fridge, getting dishes to put the food in, and eating the cereal). In contrast, the baseline shows only a portion of the relevant objects and repeats them throughout, failing to capture how one event leads to the next.

Importance of individual subshots The second term of our objective (Eqn. 2) accounts for the individual quality of each selected subshot. Subshots containing an identical set of objects can still vary in perceived quality, depending on how prominent the objects are in the scene, the camera angle, the amount of blur, etc. We use an importance criterion similar to [14], which is a learned function that takes a region as input and returns a scalar importance score as output. It exploits cues specific to egocentric data, such as nearness of the region to the camera wearer's hands, its size and location, and its frequency of appearance in a short time window. We define

$$\mathcal{I}(S) = \sum_{j=1}^{K} \textsc{Importance}(s_j), \tag{7}$$

where the importance of a subshot s_j is the average of importance scores for all its regions. Note our influence computation also uses importance to weight edges in G above; however, the normalization step discards the overall importance of the subshot that we capture here.

Diversity among transitions The third term in Eqn. 2 enforces scene diversity in the selected chain. We compute an affinity based the distribution of scenes present in two adjacent subshots. Specifically, we extract Gist descriptors and color histograms for each frame, then quantize them to one of 55 scene types (as identified with mean-shift clustering). A subshot's scene distribution is a histogram over those scene types. Then, we score diversity as:

$$\mathcal{D}(S) = \sum_{j=1}^{K-1} \left(1 - \exp(-\frac{1}{\Omega}\chi^2(s_j, s_{j+1}))\right), \tag{8}$$

where Ω is the mean of χ^2-distances among all nodes, and s_j refers to its scene histogram. Note this value is high when the scenes in sequential subshots are dissimilar.

3.3. Searching for the Optimal Chain of Subshots

A naive search for the optimal chain would involve computing Eqn. 2 for all possible chains. While importance and scene diversity can be computed quickly, the story-driven term (Eqn. 6) is more expensive, as it uses linear programming. To efficiently find a good chain, we use the approximate best-first search strategy given in [24], modified to account for our full objective. The basic idea is to use a priority queue to hold intermediate chains, and exploit the fact that computing the story term $\mathcal{S}$ for a single-link chain is very efficient.

Briefly, it works as follows. The priority queue is initialized with a single node chain. Each chain in the priority queue is either associated with its $Q(S)$ score or an approximate score that is computed very efficiently. The approximate score computes $\mathcal{I}(S) + \mathcal{D}(S)$ for the new chain, and adds the minimum of the $\mathcal{S}(S)$ scores for the current chain and the newly added link. At each iteration, the top chain in the priority queue is scored by $Q(S)$ and reinserted if the chain is currently associated with its approximate score; otherwise the chain is expanded to longer chains by adding the subsequent subshots. Then each newly created chain is inserted in the priority queue with its approximate score. In this way unnecessary $\mathcal{S}(S)$ computation is avoided. The algorithm terminates when the chain is of desired length. The authors provide approximation guarantees for this approach [24]; they are also applicable for our case since our objective adds only pairwise and individual node terms.

3.4. Selecting a Chain of Chains in Long Videos

For long egocentric video inputs, it is often ill-posed to measure influence across the boundaries of major distinct events (such as entirely different physical locations). For example, having dinner in a restaurant has little to do with watching TV later on at home—at least in terms of *visual* cues that we can capture. Based on this observation, we pose the final summarization task in two layers. First, we automatically decompose the full video into major events. We compute an affinity matrix A over all pairs of subshots, based on both their color similarity and mutual influence:

$$A_{m,n} = \alpha_1 \exp(-\frac{1}{\Omega}\chi^2(s_m, s_n)) + \alpha_2 \sum_o \textsc{Influence}(s_m, s_n | o),$$

where α_1 and α_2 are weights, Ω is as above, and s_m refers to its color histogram in the first term. To compute a boundary score for each subshot, we sum the affinity between that

subshot and all others within a small temporal window, and normalize that value by the affinity of all pairs in which one subshot is either before or after the current window. Event boundaries are the local minima in this boundary measurement.

Next, for each major event, we generate multiple candidate chains as in Sec. 3.2 by varying K. The final video summary is constructed by selecting one chain from the candidates per event, and concatenating the selected chains together. We simply select the chain with the highest importance among those for which the minimum diversity term is higher than a threshold τ.

For a typical 4-hour video with 7 major events, it takes $30-40$ minutes to generate the final summary. Note that our system is implemented in Matlab without any optimization. The run time could be improved by using a faster LP implementation and caching $Q(S)$ when generating chains.

4. Results

We now analyze our method compared to multiple alternative summarization approaches. Since judging the quality of a summary is a complex and subjective task, we conduct a substantial user study to quantify its performance.

Datasets We use two datasets: the UT Egocentric (UTE) dataset[2] [14] and the Activities of Daily Living (ADL) dataset[3] [20]. UTE contains 4 videos from head-mounted cameras, each about $3-5$ hours long, captured in a very uncontrolled setting. The camera wearers travel through multiple locations, eating, shopping, walking, driving, cooking, etc. We use visual words for this data; the objects present are so diverse that pre-specifying a bank of detectors would be inadequate. ADL contains 20 videos from chest-mounted cameras, each about $20-60$ minutes long. The camera wearers perform daily activities in the house, like brushing teeth, washing dishes, or making a snack. The data are annotated with a set of $N = 42$ relevant objects (e.g., mug, fridge, TV), which we use to demonstrate how our method performs using familiar "true" objects. We use the provided ground truth bounding boxes rather than raw detector outputs, in an effort to focus on summarization issues rather than object detector issues. For ADL we use keyframes rather than subshots due to their shorter duration.

Implementation details For the pHOG visual words, we use 8×8 blocks, a 4 pixel step size, and 2 scales per octave. We set $\varepsilon = 0.25$ for influence computation, following [24]. We set γ to constrain the total number of activated objects to 80 and 15 for UTE and ADL, respectively, reflecting the datasets' differing total number of objects. We weigh the objective terms as $\lambda_s = 1$, $\lambda_i = 0.5$, and $\lambda_d = 0.5$, to emphasize the story-based criterion. For event boundary detection, we set $\alpha_1 = 0.1$ and $\alpha_2 = 0.9$. For each event, we use $K = 4, \ldots, 8$ to generate the candidate chains. We set the minimum scene diversity to be $\tau = 0.6$ for UTE and $\tau = 0.45$ for ADL after visually examining a few examples. We process 1 fps for efficiency.

Being that our approach is unsupervised, validating parameter settings is of course challenging. We stress that nearly all were set simply based on intuitions given above, and not tuned. We did observe some trade-offs in two parameters, however—the range for K and scene diversity threshold τ. If K is too high or τ too low, the summaries contain more redundancies. A user of our system would likely inspect a couple examples in order to adjust them, just as we have done here. We fix all parameters for all results, and use the same final K for all compared methods.

Baselines We compare to three baselines: (1) **Uniform sampling**: We select K subshots uniformly spaced throughout the video. This is a simple yet often reasonable method. (2) **Shortest-path**: We construct a graph where all pairs of subshots have an edge connecting them, and the edge is weighted by their bag-of-objects distance. We then select the K subshots that form the shortest path connecting the first and last subshot. This baseline has the benefit of the same object representation we use and should find a smooth path of sub-events, but it lacks any notion of influence. (3) **Object-driven**: We apply the state-of-the-art egocentric summarization method [14] using the authors' code. Because it produces keyframe summaries, we map its output to a video skim by including the 15 frames surrounding each selected keyframe. For ADL, only the first two baselines are compared, since the object-driven approach [14] would require additional annotation of important objects for training, which is outside the scope of this paper.

Evaluating summary quality We perform a "blind taste test" in which users report which summary best captures the original story. The test works as follows. We first show the users a sped-up version of the entire original video, and ask them to write down the main story events. The latter is intended to help them concentrate on the task at hand. Then, we show the subject two summaries for that original video; one is ours, one is from a baseline method. We do not reveal which is which, and we order them randomly. After viewing both, the subject is asked, *Which summary better shows the progress of the story?* We also emphasize that the subjects should pay attention to the relationship among sub-events, redundancy, and representativeness of each sub-event. The supp. file shows the complete interface.

The final set shown to subjects consists of 5 hours and 11 events for UTE and 7 hours and 37 events for ADL.[4] We

[2] http://vision.cs.utexas.edu/projects/egocentric/

[3] http://deepthought.ics.uci.edu/ADLdataset/adl.html

[4] To mitigate the cost of our user studies, we omit events not meeting the minimum scene diversity value (they are monotonous and so trivial to summarize), as well as those shorter than 3 minutes in ADL.

Data	Uniform sampling	Shortest path	Object-driven [14]
UTE	90.9%	90.9%	81.8%
ADL	75.7%	94.6%	N/A

Table 1. User study results. Numbers indicate percentage of users who prefer our method's summary over each of the three baselines.

enlisted 34 total subjects. They range from 18-60 years old, and about half have no computer vision background. We show our summary paired separately with each baseline to five different users, and take a vote to robustly quantify the outcome. This makes $11 \times 3 \times 5 = 165$ comparisons for UTE and $37 \times 2 \times 5 = 370$ comparisons for ADL, for a total of 535 tasks done by our subjects. We estimate each task required about 5 minutes to complete, meaning the study amounts to about 45 hours of user time. To our knowledge, this ranks among the most extensive user studies performed to systematically evaluate a summarization algorithm.

Table 1 shows the results. A strong majority of the subjects prefer our summaries over any of the baselines'. This supports our main claim, that our approach can better capture stories in egocentric videos. Furthermore, in 51% of the comparisons all five subjects prefer our summary, and only in 9% of the comparisons does our summary win by one vote.

Inspecting the results, we find that our advantage is best when there is a clear theme in the video, e.g., buying ice cream or cooking soup. In such cases, our model of coherent influence finds subshots that give the sense of one event leading to the next. In contrast, the state-of-the-art approach [14] tends to include subshots with important objects, but with a less obvious thread connecting them. When a video focuses on the same scene for a long time, our method summarizes a short essential part, thanks to our importance and scene diversity terms. In contrast, both uniform sampling and shortest-path tend to include more redundant subshots. In fact, we believe shortest-path is weakest relative to our approach on the ADL data because it contains many such activities (e.g., using a computer, watching TV).

On the other hand, our method does not have much advantage when the story is uneventful, or when there are multiple interwoven threads (e.g., cooking soup and making cookies at the same time). In such cases, our method tends to select a chain of subshots that are influential to each other, but miss other important parts of the story. In a sense, such multi-tasking is inherently breaking the visual storyline.

Example summaries Figures 6 and 7 show all methods' summaries for example UTE and ADL inputs. See captions for explanations. Please see our website for video result examples[2].

Discovering influential objects Finally, we demonstrate how our influence estimates can be used to discover the objects most influential to the story. For a given video, we sort the objects $o_i \in O$ by their total influence scores across all its subshot transitions (Eqn. 5). Figure 5 shows ROC curves for our discovered objects on the ADL data, compared to a baseline that ranks the objects by their frequency within the video. To obtain ground truth, we had 3 workers on MTurk identify which of the $N = 42$ objects they found central to the story per video, and took the majority vote. The results show our method's advantage; the most influential objects need not be the most frequent. We stress that our method is unsupervised, and discovers the central objects looking at a *single* video—as opposed to a supervised approach that might exploit multiple labeled videos to find typical objects. This application of our work may be useful for video retrieval or video saliency detection applications.

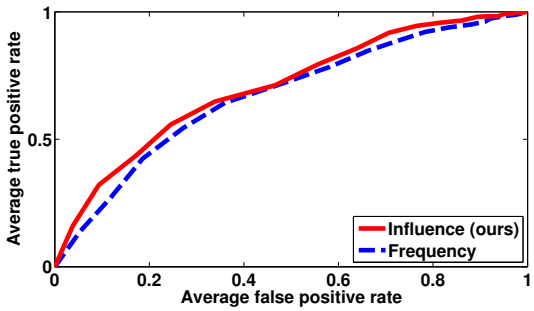

Figure 5. Comparing our method to a frequency-based baseline for the task of discovering influential objects. See text.

5. Conclusion

Our work brings the notion of "story" into video summarization, making it possible to link sub-events based on the relationships between their objects, not just their co-occurring features. Towards this goal, we have developed a novel subshot segmentation method for egocentric data, and a selection objective that captures the influence between subshots as well as shot importance and diversity. Our large-scale user study indicates the promise of our approach. The results also suggest how our unsupervised technique might assist in other vision tasks, such as discovering the objects central for human activity recognition.

We are interested in our method's use for egocentric data, since there is great need in that domain to cope with long unedited video—and it will only increase as more people and robots wear a camera as one of their mobile computing devices. Still, in the future we'd like to explore visual influence in the context of other video domains. We also plan to extend our subshot descriptions to reflect motion patterns or detected actions, moving beyond the object-centric view.

Acknowledgements We thank all the user study subjects, and Yong Jae Lee for helpful discussions. This research is supported in part by ONR YIP N00014-12-1-0754.

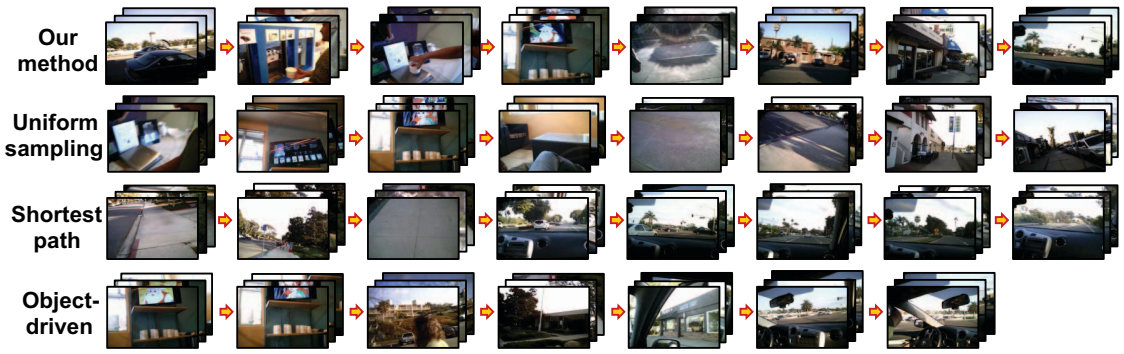

Figure 6. Example from UTE data comparing our summary (top row) to the three baselines. Our method clearly captures the progress of the story: serving ice cream leads to weighing the ice cream, which leads to watching TV in the ice cream shop, then driving home. Even when there are no obvious visual links for the story, our method captures visually distinct scenes (see last few subshots in top row). The shortest-path approach makes abrupt hops across the storyline in order to preserve subshots that smoothly transition (see redundancy in its last 5 subshots). While the object-driven method [14] does indeed find some important objects (e.g., TV, person), the summary fails to suggest the links between them. Note that object-driven method sometimes produces shorter summaries (like this example) depending on number of unique important objects discovered in the video. See supplementary file for videos.

Figure 7. Example from ADL data. While here uniform sampling produces a plausible result, ours appears to be more coherent. Objects such as *milk* and *cup* connect the selected keyframes and show the progress of the story—preparing a hot drink and enjoying it by the TV. Shortest-path produces the weakest result due to its redundant keyframes. This is often the case if the input has many similar frames, since it accounts for the sum of all link weights' *similarity*, without any notion of *influence*. See supplementary file for videos.

References

[1] O. Aghazadeh, J. Sullivan, and S. Carlsson. Novelty detection from an egocentric perspective. In *CVPR*, 2011.

[2] B. Alexe, T. Deselaers, and V. Ferrari. What is an object? In *CVPR*, 2010.

[3] A. Bosch, A. Zisserman, and X. Munoz. Representing shape with a spatial pyramid kernel. In *CIVR*, 2007.

[4] F. Crete-Roffet, T. Dolmiere, P. Ladret, and M. Nicolas. The blur effect: Perception and estimation with a new no-reference perceptual blur metric. In *SPIE*, 2007.

[5] A. Doherty, D. Byrne, A. Smeaton, G. Jones, and M. Hughes. Investigating keyframe selection methods in the novel domain of passively captured visual lifelogs. In *CIVR*, 2008.

[6] M. Ellouze, N. Boujemaa, and A. M. Alimi. Im(s)2: Interactive movie summarization system. *J VCIR*, 21(4):283–294, 2010.

[7] A. Fathi, A. Farhadi, and J. Rehg. Understanding egocentric activities. In *ICCV*, 2011.

[8] A. Fathi, J. K. Hodgins, and J. M. Rehg. Social interactions: A first-person perspective. In *CVPR*, 2012.

[9] A. Fathi, Y. Li, and J. M. Rehg. Learning to recognize daily action using gaze. In *ECCV*, 2012.

[10] D. B. Goldman, B. Curless, and S. M. Seitz. Schematic storyboarding for video visualization and editing. In *SIGGRAPH*, 2006.

[11] N. Jojic, A. Perina, and V. Murino. Structural epitome: A way to summarize one's visual experience. In *NIPS*, 2010.

[12] K. Kitani, T. Okabe, Y. Sato, and A. Sugimoto. Fast unsupervised ego-action learning for first-person sports video. In *CVPR*, 2011.

[13] R. Laganiere, R. Bacco, A. Hocevar, P. Lambert, G. Pais, and B. E. Ionescu. Video summarization from spatio-temporal features. In *Proc of ACM TRECVid Video Summarization Wkshp*, 2008.

[14] Y. J. Lee, J. Ghosh, and K. Grauman. Discovering important people and objects for egocentric video summarization. In *CVPR*, 2012.

[15] C. Liu. Beyond pixels: Exploring new representations and applications for motion analysis. In *MIT press*, 2009.

[16] D. Liu, G. Hua, and T. Chen. A hierarchical visual model for video object summarization. *PAMI*, 32(12):2178–2190, 2010.

[17] T. Liu and J. R. Kender. Optimization algorithms for the selection of key frame sequences of variable length. In *ECCV*, 2002.

[18] J. Nam and A. H. Tewfik. Event-driven video abstraction and visualization. *Multimedia Tools Application*, 16(1):55–77, 2002.

[19] C.-W. Ngo, Y.-F. Ma, and H.-J. Zhang. Automatic video summarization by graph modeling. In *ICCV*, 2003.

[20] H. Pirsiavash and D. Rmanan. Detecting activities of daily living in first-person camera views. In *CVPR*, 2012.

[21] S. Pongnumkul, J. Wang, and M. Cohen. Creating map-based storyboards for browsing tour videos. In *UIST*, 2008.

[22] Y. Pritch, A. Rav-Acha, A. Gutman, and S. Peleg. Webcam synopsis: Peeking around the world. In *ICCV*, 2007.

[23] X. Ren and C. Gu. Figure-ground segmentation improves handled object recognition in egocentric video. In *CVPR*, 2010.

[24] D. Shahaf and C. Guestrin. Connecting the dots between news articles. In *KDD*, 2010.

[25] E. Spriggs, F. D. la Torre, and M. Hebert. Temporal segmentation and activity classification from first-person sensing. In *CVPR Wkshp on Egocentric Vision*, 2009.

[26] W. Wolf. Key frame selection by motion analysis. In *ICASSP*, 1996.

[27] H.-J. Zhang, J. Wu, D. Zhong, and S. W. Smoliar. An integrated system for content-based video retrieval and browsing. *Pattern Recognition*, 30(4):643–658, 1997.