Ego2Top: Matching Viewers in Egocentric and Top-view Cameras

Shervin Ardeshir and Ali Borji

Center for Research in Computer Vision at University of Central Florida

Abstract. Egocentric cameras are becoming increasingly popular and provide us with large amounts of videos, captured from the first person perspective. At the same time, surveillance cameras and drones offer an abundance of visual information, often captured from top-view. Although these two sources of information have been separately studied in the past, they have not been collectively studied and related. Having a set of egocentric cameras and a top-view camera capturing the same area, we propose a framework to identify the egocentric viewers in the top-view video. We utilize two types of features for our assignment procedure. Unary features encode what a viewer (seen from top-view or recording an egocentric video) visually experiences over time. Pairwise features encode the relationship between the visual content of a pair of viewers. Modeling each view (egocentric or top) by a graph, the assignment process is formulated as spectral graph matching. Evaluating our method over a dataset of 50 top-view and 188 egocentric videos taken in different scenarios demonstrates the efficiency of the proposed approach in assigning egocentric viewers to identities present in top-view camera. We also study the effect of different parameters such as the number of egocentric viewers and visual features.

Keywords: Egocentric Vision, Surveillance, Spectral Graph Matching, Gist, Video Understanding

1 Introduction

The availability of large amounts of egocentric videos captured by cellphones and wearable devices such as GoPro cameras and Google Glass has opened the door to a lot of interesting research in computer vision [1–3]. At the same time, videos captured with top-down static cameras such as surveillance cameras in airports and subways, unmanned aerial vehicles (UAVs) and drones, provide us with a lot of invaluable information about activities and events taking place at different locations and environments. Relating these two complementary, but drastically different sources of visual information can provide us with rich analytical power, and help us explore what can not be inferred from each of these sources taken separately. Establishing such a relationship can have several applications. For example, athletes can be equipped with body-worn cameras, and their egocentric videos together with the top-view videos can offer new data useful for better

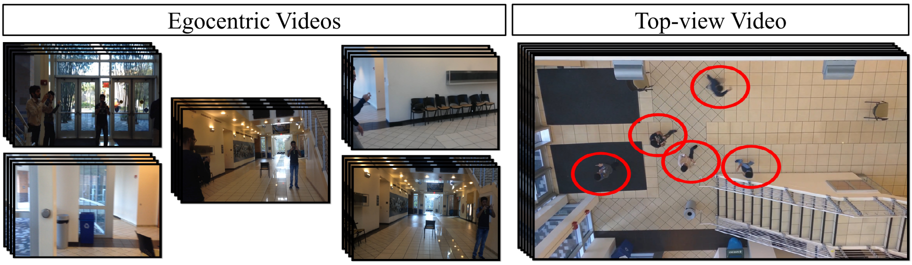


Fig. 1: Left shows a set of 5 egocentric videos. Right shows a top-view video capturing the whole scene. The viewers are highlighted using red circles in the top-view video. We aim to answer the two following questions: 1) Does this set of egocentric videos belong to the viewers visible in the top-view video? 2) Assuming they do, which viewer is capturing which egocentric video?

technical and tactical sport analysis. Moreover, due to the use of wearable devices and cameras by law enforcement officers, finding the person behind an egocentric camera in a surveillance network could be a useful application. Furthermore, fusing these two types of information can result in better 3D reconstruction of an environment by combining the top-view information with first person views.

The first necessary step to utilize information from these two sources, is to establish correspondences between the two views. In other words, a matching between egocentric cameras and the people present in the top-view camera is needed. In this effort, we attempt to address this problem. More specifically, our goal is to localize people recording egocentric videos, in a top-view reference camera. To the best of our knowledge, such an effort has not been done so far. In order to evaluate our method, we designed the following setup. A dataset containing several test cases is collected. In each test case, multiple people were asked to move freely in a certain environment and record egocentric videos. We refer to these people as ego-centric *viewers*. At the same time, a top-view camera was recording the entire scene/area including all the egocentric viewers and possibly other intruders. An example case is illustrated in Figure 1.

Given a set of egocentric videos and a top-view surveillance video, we try to answer the following two questions: 1) Does this set of egocentric videos belong to the viewers visible in the top-view camera? 2) If yes, then which viewer is capturing which egocentric video? To answer these questions, we need to compare a set of egocentric videos to a set of viewers visible in a single top-view video. To find a matching, each set is represented by a graph and the two graphs are compared using a spectral graph matching technique [4]. In general, this problem can be very challenging due to the nature of egocentric cameras. Since the camera-holder is not visible in his own egocentric video leaving us with no cues about his visual appearance.

In what follows we briefly mention some challenges concerning this problem and sketch the layout of our approach.

In order to have an understanding of the behavior of each individual in the top-view video, we use a multiple object tracking method [5] to extract the

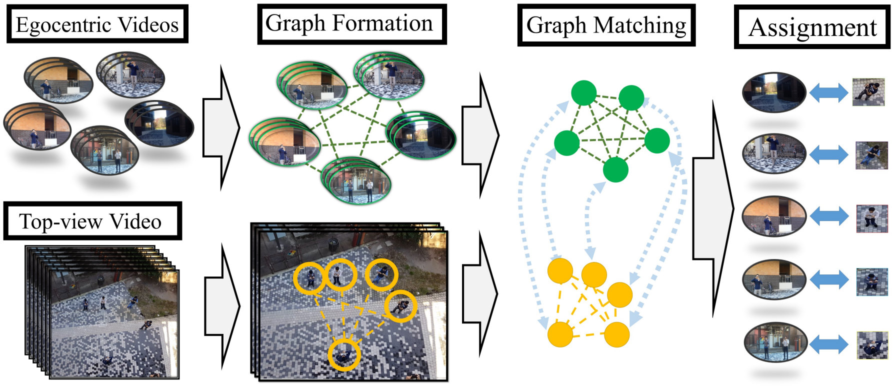


Fig. 2: The input to our framework is a set of egocentric videos (in this case 5 videos), and one top-view video. The goal is defined as assigning the egocentric videos to the people recording them. A graph is formed on the set of egocentric videos (each node being one of the egocentric videos), and the other graph is formed on the top-view video (each node being one of the targets present in the video). Using spectral graph matching, a soft assignment is found between the two graphs, and using a soft-to-hard assignment, each egocentric video is assigned to one of the viewers in the top-view video. This assignment is our answer to the second question in 1.

viewer's trajectory in the top-view video. Note that an egocentric video captures a person's field of view rather than his spatial location. Therefore, the content of a viewer's egocentric video, a 2D scene, corresponds to the content of the viewer's field of view in the top-view camera. For the sake of brevity, we refer to a viewer's top-view field of view as Top-FOV in what follows. Since trajectories computed by multiple object tracking do not provide us with the orientation of the egocentric cameras in the top-view video, we employ the assumption that for the most part humans tend to look straight ahead and therefore shoot videos from the visual content in front of them. Note that this is not a restrictive assumption as most ego-centric cameras are body worn (Please see Figure 4). Having an estimate of a viewer's orientation and Top-FOV, we then encode the changes in his Top-FOV over time and use it as a descriptor. We show that this feature correlates with the change in the global visual content (or Gist) of the scene observed in his corresponding egocentric video.

We also define pairwise features to capture the relationship between two egocentric videos, and also the relationship between two viewers in the top-view camera. Intuitively, if an egocentric viewer observes a certain scene and another egocentric viewer comes across the same scene later, this could hint as a relationship between the two cameras. If we match a top-view viewer to one of the two egocentric videos, we are likely to be able to find the other viewer using the mentioned relationship. As we experimentally show, this pairwise relationship significantly improves our assignment accuracy. This assignment will lead to defining a score measuring the similarity between the two graphs. Our exper-

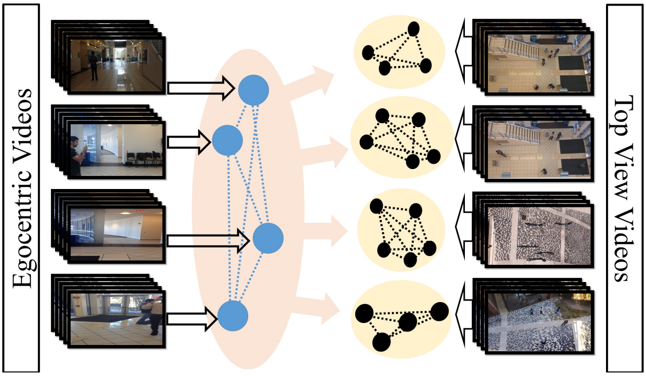


Fig. 3: Adapting our method for evaluating top-view videos. We form a graph on the set of egocentric videos and compare this graph to other graphs built on different top-view videos. The top-view videos are ranked based on how similar their graph is to the egocentric graph. The performance of this ranking helps us answer our first question.

iments demonstrate that the graph matching score could be used for verifying if the top-view video is in fact, capturing the egocentric viewers (See the diagram shown in Figure 7a).

The rest of this work is as follows. In section 2, we mention related works to our study. In section 3, we describe the details of our framework. Section 4 presents our experimental results followed by discussions and conclusions in Section 5.

2 Related Work

Visual analysis of egocentric videos has recently became a hot topic in computer vision [6, 7], from recognizing daily activities [2, 1] to object detection [8], video summarization [9], and predicting gaze behavior [10–12]. In the following, we review some previous work related to ours spanning *Relating static and egocentric*, *Social interactions among egocentric viewers*, and *Person identification and localization*.

Relating Static and Egocentric Cameras: Some studies have addressed relationships between moving and static cameras. Interesting works reported in [13, 14] have studied the relationship between mobile and static cameras for the purpose of improving object detection accuracy. [15] fuses information from egocentric and exocentric vision (other cameras in the environment) and laser depth range data to improve depth perception in 3D reconstruction. [16] predicts gaze behavior in social scenes using first-person and third-person cameras.

Social Interactions among Egocentric Viewers: To explore the relationship among multiple egocentric viewers, [17] combines several egocentric videos to achieve a more complete video with less quality degradation by estimating the importance of different scene regions and incorporating the consensus among several egocentric videos. Fathi et al., [18] detect and recognize the type of social interactions such as dialogue, monologue, and discussion by detecting human faces and estimating their body and head orientations. [19] proposes a multi-task clustering framework, which searches for coherent clusters of daily actions using the notion that people tend to perform similar actions in certain environments such as workplace or kitchen. [20] proposes a framework that discovers static and movable objects used by a set of egocentric users.

Person Identification and Localization: Perhaps, the most similar computer vision task to ours is person re-identification [21–23]. The objective here is to find the person present in one static camera, in another overlapping or non-overlapping static camera. However, the main cue in human re-identification is visual appearance of humans, which is absent in egocentric videos. Tasks such as human-identification and localization in egocentric cameras have been studied in the past. [24] uses the head motion of an egocentric viewer as a biometric signature for determine which videos have been captured by the same person. [25] identifies egocentric observers in other egocentric videos, using their head motion. Relating geo-spatial location to user shared visual content has also been explored. [3] localizes the field of view of an egocentric camera by matching it against a reference dataset of videos or images (such as Google street view), and [26] refines the geo-location of images by matching them against user shared images. Landmarks and map symbols are used in [27] to perform self localization on the map. [28] use semantic cues for spatial localization, and [29] uses location information to infer semantic information.

3 Framework

The block diagram in Figure 2 illustrates different steps of our approach. First, each view (ego-centric or top-down) is represented by a graph which defines the relationship among the viewers present in the scene. These two graphs may not have the same number of nodes as some the egocentric videos might not be available, or some individuals present in the top-view video might not be capturing videos. Each graph consists of a set of nodes, each of which represents one viewer (egocentric or top-view), and the edges of the graph encode pairwise relationships between pairs of viewers.

We represent each viewer in top-view by describing his expected Top-FOV, and in egocentric view by the visual content of his video over time. This description is encoded in the nodes of the graphs. We also define pairwise relationships between pairs of viewers, which is encoded as the edge features of the graph (i.e., how two viewers' visual experience relate to each other).

Second, we use spectral graph matching to compute a score measuring the similarity between the two graphs, alongside with an assignment from the nodes of the egocentric graph to the nodes of the top-view graph.

Our experiments show that the graph matching score can be used as a measure of similarity between the egocentric graph and the top-view graph. Therefore, it can be used as a measure for verifying if a set of egocentric videos have been shot in the same environment captured by the top-view camera. In other words, we can evaluate the capability of our method in terms of answering our first question. In addition, the assignment obtained by the graph matching suggests an answer to our second question. We organize this section by going over the graph formation process for each of the views, and then describing the details of the matching procedure.

3.1 Graph Representation

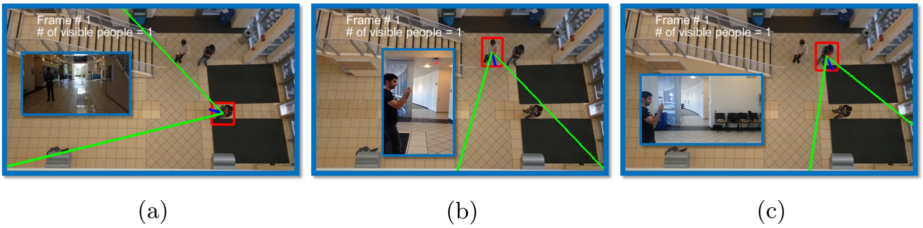
Each view (egocentric or top-view) is described using a single graph. The set of egocentric videos is represented using a graph in which each node represents one of the egocentric videos, and an edge captures the pairwise relationship between the content of the two videos.

In the top-view graph, each node represents the visual experience of a viewer being tracked (in the top-view camera), and an edge captures the pairwise relationship between the two. By visual experience we mean what a viewer is expected to observe during the course of his recording seen from the top view.

3.1.1 Modeling the Top-View Graph: In order to model the visual experience of a viewer in a top-view camera, we need to have knowledge about his spatial location (trajectory) throughout the video. We employ the multiple object tracking method presented in [5] and extract a set of trajectories, each corresponding to one of the viewers in the scene. Similar to [5], we use annotated bounding boxes, and provide their centers as an input to the multiple object tracker. Our tracking results are nearly perfect due to several reasons: the high quality of videos, high video frame rate, and lack of challenges such as occlusion in the top-view videos.

Each node represents one of the individuals being tracked. Employing the general assumption that people often tend to look straight ahead, we use a person's speed vector as the direction of his camera at time t (denoted as θ_t). We pre-process the speed vectors using a Gaussian filter to temporally smooth them and exclude outliers. This also fixes the orientation for a short standing interval between two movements. Further, assuming a fixed angle (θ_d), we expect the content of the person's egocentric video to be consistent with the content included in a 2D cone formed by the two rays emanating from the viewer's location and with angles $\theta - \theta_d$ and $\theta + \theta_d$. Figure 4 illustrates the expected Top-FOV for three different individuals present in a frame. In our experiments, we set θ_d to 30 degrees. In theory, angle θ_d can be estimated more accurately by knowing intrinsic camera parameters such as focal length and sensor size of the corresponding egocentric camera. However, since we do not know the corresponding egocentric camera, we set it to a default value.

Top-FOVs are not directly comparable to viewers' egocentric views. The area in the Top-FOV in a top-view video mostly contains the ground floor which is



(a) (b) (c)

Fig. 4: Expected field of view for three different viewers in the top-view video alongside with their corresponding egocentric frames. The short dark blue line shows the estimated orientation of the camera. The Top-FOV shown in (b) and (c) have a high overlap, therefore we expect their egocentric videos to have relatively similar visual content compared to the pairs (a,b) or (a,c) at this specific time.

not what an ego-centric viewer usually observes in front of him. However, what can be used to compare the two views is the relative change in the Top-FOV of a viewer over time. This change should correlate with the change in the content of the egocentric video. Intuitively, if a viewer is looking straight ahead while walking on a straight line, his Top-FOV is not going to have drastic changes. Therefore, we expect the viewer's egocentric view to have a stable visual content.

Node Features: We extract two unary features for each node, one captures the changes in the content covered by his FOV, and the other is the number of visible people in the content of the Top-FOV.

To encode the relative change in the visual content of viewer i visible in the top-view camera, we form the $T \times T$ matrix (T denotes the number of frames in the top-view video) U_i^{IOU} whose elements $U_i^{IOU}(f_p, f_q)$ indicate the IOU (intersection over union) of the Top-FOV of person i in frames f_p and f_q. For example, if the viewer's Top-FOV in frame 10 has high overlap with his FOV in frame 30 (thus $U_i^{IOU}(10, 30)$ has a high value), we expect to see a high visual similarity between frames 10 and 30 in the egocentric video. Two examples of such features are illustrated in the middle column of Figure 5 (a).

Having the Top-FOV of viewer i estimated, we then count the number of people within his Top-FOV at each time frame and store it in a $1 \times T$ vector U_i^n. To count the number of people, we used annotated bounding boxes. Figure 4 illustrates three viewers who have one human in their Top-FOV. A few examples of this feature are visualized in the top row of figure 6.

Edge Features: Pairwise features are designed to capture the relationship among two different individuals. In the top-view videos, similar to the unary matrix U_i^{IOU}, we can form a $T \times T$ matrix B_{ij}^{IOU} to describe the relationship between a pair of viewers (viewers/nodes i and j), in which $B_{ij}^{IOU}(f_p, f_q)$ is defined as the intersection over union of the Top-FOVs of person i in frame f_p and person j in frame f_q. Intuitively, if there is a high similarity between the Top-FOVs of person i in frame 10 and person j in frame 30, we would expect the 30th frame of viewer j's egocentric video to be similar to the 10th frame of viewer i's egocentric video. Two examples of such features are illustrated in the middle column of Figure 5 (b).

3.1.2 Modeling the Egocentric View Graph: As in the top-view graph, we also construct a graph on the set of egocentric videos. Each node of this graph represents one egocentric video. Edges between the nodes capture the relationship between two egocentric videos.

Node Features: Similar to the top-view graph, each node is represented using two features. First, we compute pairwise similarity between GIST features [30] of all video frames (for one viewer) and store the pairwise similarities in the matrix $U_{E_i}^{GIST}$, in which the element $U_{E_i}^{GIST}(f_p, f_q)$ is the GIST similarity between frame f_p and f_q of egocentric video i. Two examples of such features are illustrated in the left column of Figure 5 (a). The GIST similarity is a function of the euclidean distance of the GIST feature vectors.

$$U_{E_i}^{GIST}(f_1, f_2) = e^{-\gamma|g_{f_p}^{E_i} - g_{f_q}^{E_i}|}. \tag{1}$$

In which $g_{f_p}^{E_i}$ and $g_{f_q}^{E_i}$ are the GIST descriptors of frame f_p and f_q of egocentric video i, and γ is a constant which we empirically set to 0.5.

The second feature is a time series counting the number of seen people in each frame. In order to have an estimate of the number of people, we run a pre-trained human detector using deformable part model [31] on each egocentric frame. In order to make sure that our method is not including humans in far distances (which are not likely to be present in the top-view camera), we exclude bounding boxes whose sizes are smaller than a certain threshold (determined considering an average human height of 1.7m and distance of the radius of the area being covered in the top view video.). Each of the remaining bouding boxes, has a detection score (rescaled into the interval [0 1]) which has the notion of the probability of that bounding box containing a person. Scores of all detections in a frame are added and used as a count of people in that frame. Therefore, similar to the top-view feature, we can represent the node E_i of egocentric video i with a $1 \times T_{E_i}$ vector $U_{E_i}^{n}$. A few examples of this feature are visualized in the bottom row of figure 6.

Edge Features: To capture the pairwise relationship between egocentric camera i (containing T_{E_i} frames) and egocentric camera j (containing T_{E_j} frames), we extract GIST features from all of the frames of both videos and form a $T_{E_i} \times T_{E_j}$ matrix B_{ij}^{GIST} in which $B_{ij}^{GIST}(f_p, f_q)$ represents the GIST similarity between frame f_p of video i and frame f_q of video j.

$$B_{ij}^{GIST}(f_p, f_q) = e^{-\gamma|g_{f_p}^{E_i} - g_{f_q}^{E_j}|}. \tag{2}$$

Two examples of such features are illustrated in the left column of Figure 5 (b).

3.2 Graph Matching

Our goal in this section is to find a binary assignment matrix $x_{N^e \times N^t}$, in which N^e is the number of egocentric videos and N^t is the number of people in the top-view video. $x(i,j)$ equal to 1 means that egocentric video i has been matched to viewer j in the top-view camera. To capture the similarities between the elements of the two graphs, we define the affinity matrix $A_{N^eN^t \times N^eN^t}$ in which $a_{ik,jl}$ is the affinity of edge ij in the egocentric graph with edge kl in the top-view graph. Reshaping matrix x as a vector $x_{N^eN^t \times 1} \in \{0,1\}^{N^eN^t}$, the assignment problem is defined as maximizing the following objective function:

$$\underset{x}{\text{argmax}}\ x^T A x. \tag{3}$$

We compute $a_{ik,jl}$ based on the similarity between the feature descriptor of edge ij in the egocentric graph B_{ij}^{GIST} and the feature descriptor for edge kl in the top-view graph B_{kl}^{IOU}.

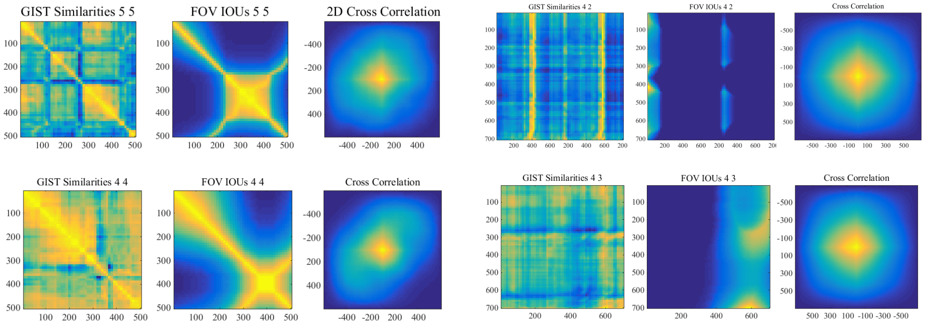
As described in the previous section, each of these features is a 2D matrix. B_{ij}^{GIST} is a $T_{E_i} \times T_{E_j}$ matrix, T_{E_i} and T_{E_j} being the number of frames in egocentric videos i and j, respectively. On the other hand, B_{kl}^{IOU} is a $T_t \times T_t$ matrix, T_t being the number of frames in the top-view video. However, expecting B_{ij}^{GIST} and B_{kl}^{IOU} to be comparable is not reasonable due to two reasons. First, the two matrices are not of the same size (the videos do not necessarily have the same length). Second, the absolute time in the videos do not correspond to each other. Note that videos are not time-synchronized. For example, the relationship between viewers i and j in the 100th frame of the top-view video does not correspond to frame number 100 of the egocentric videos. Instead, we expect to see a correlation between the GIST similarity of frame $100 + d_i$ of egocentric video i and frame $100 + d_j$ of egocentric video j, and the intersection over union of in Top-FOVs of viewers k and l in frame 100. d_i and d_j are the time delays of egocentric videos i and j with respect to the top-view video.

As a result, we need to define an affinity score which is able to handle this misalignment. To this end, we compute the affinity between the two 2D matrices as the maximum value of their 2D cross correlation. Hence, if egocentric videos i and j have d_i and d_j delays with respect to the top-view video, the cross correlation between B_{ij}^{GIST} and B_{kl}^{IOU} should be maximum when B_{ij}^{GIST} is shifted d_i units in the first, and d_j units in the second dimension.

$$A_{ikjl} = \max(B_{ij}^{GIST} * B_{kl}^{IOU}). \tag{4}$$

where $*$ denotes cross correlation. For the elements of A for which $i = j$ and $k = l$, the affinity captures the compatibility of node i in the egocentric graph, to node k in the top-view graph. The compatibility between the two nodes is computed using 2D cross correlation between U_k^{IOU} and $U_{E_i}^{GIST}$ and 1D cross correlation between U_k^n and $U_{E_i}^n$. The overall compatibility of the two nodes is a weighted linear combination of the two:

$$A_{ikik} = \alpha \max(U_{E_i}^{GIST} * U_k^{IOU}) + (1-\alpha)\max(U_{E_i}^n * U_k^n), \tag{5}$$



(a) (b)

Fig. 5: (a) shows two different examples of the 2D features extracted from the **nodes** of the graphs, for which the values are color-coded. Left column shows the 2D matrices extracted from the pairwise similarities of the GIST feature descriptors U^{GIST}, middle shows the 2D matrices computed by intersection over union of the FOV in the top-view camera U^{IOU}, and the rightmost column shows the result of the 2D cross correlation between the two. (b) shows the same concept, but between two **edges**. Again, the leftmost figure shows the pairwise similarity between GIST descriptors of one egocentric camera to another B^{GIST}. Middle, shows the pairwise intersection over union of the FOVs of the pair of viewers B^{IOU}, and the rightmost column illustrates their 2D cross correlation. The similarities between the GIST and FOV matrices in fact capture the affinity of two nodes/edges in the two graphs.

where α is a constant between 0 and 1 specifying the contribution of each term. In our experiments, we set α to 0.9. Figure 5 illustrates the features extracted from some of the nodes and edges in the two graphs.

Soft Assignment We employ the spectral graph matching method introduced in [4] to compute a soft assignment between the set of egocentric viewers and top-view viewers. In [4], assuming that the affinity matrix is an empirical estimation of the pairwise assignment probability, and the assignment probabilities are statistically independent, A is represented using it's rank one estimation which is computed by $\underset{p}{\operatorname{argmin}} |A - pp^T|$. In fact, the rank one estimation of A is no different than it's leading eigenvector. Therefore, p can be computed either using eigen decompositon, or estimated iteratively using power iteration. Considering vector p as the assignment probablities, we can reshape $p_{N^eN^t \times 1}$ into a $N^e \times N^t$ soft assignment matrix P, for which after row normalization $P(i, j)$ represents the probability of matching egocentric viewer i to viewer j in the top-view video.

Hard Assignment Any soft to hard assignment method can be used to convert the soft assignment result (generated by spectral matching) to the hard binary assignment between the nodes of the graphs. We used the well known Munkres (also known as Hungarian) algorithm to obtain the final binary assignment.

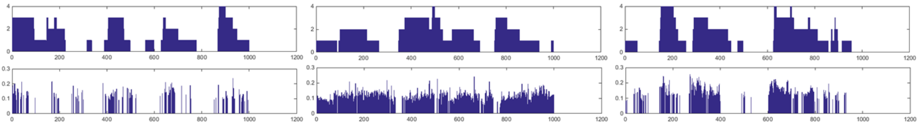


Fig. 6: Examples of the one dimensional features capturing the number of humans in different frames of the videos. The top row shows the number of visible people in each viewer's Top-FOV over time. The second row shows the summation of the detection scores at every frame from egocentric videos. The similarity between the two patterns shows the discriminative power of this feature in some cases. However, our experiments show that in most cases this feature by itself does not results in a high assignment accuracy.

4 Experimental Results

In this section, we will mention details of our experimental setup and collected dataset, the measures we used to evaluate the performance of our method, and the performance of our proposed method alongside with some baselines.

4.1 Dataset

We collected a dataset containing 50 test cases of videos shot in different indoor and outdoor conditions. Each test case, contains one top-view video and several egocentric videos captured by the people visible in the top-view camera. Depending on the subset of egocentric cameras that we include, we can generate up to 2,862 instances of our assignment problem (will be explained in more detail in section 4.2.4). Overall, our dataset contains more than 225,000 frames. Number of people visible in the top-view cameras varies from 3 to 10, number of egocentric cameras varies from 1 to 6, and the ratio of number of available egocentric cameras to the number of visible people in the top-view camera varies from 0.16 to 1. Lengths of the videos vary from 320 frames (10.6 seconds) up to 3132 frames (110 seconds). Please see supplementary material for more details on our data and sample sequences.

4.2 Evaluation

We evaluate our method in terms of answering the two questions we asked. First, given a top-view video and a set of egocentric videos, can we verify if the top-view video is capturing the egocentric viewers? We analyze the capability of our method in answering this question in section 4.2.1.

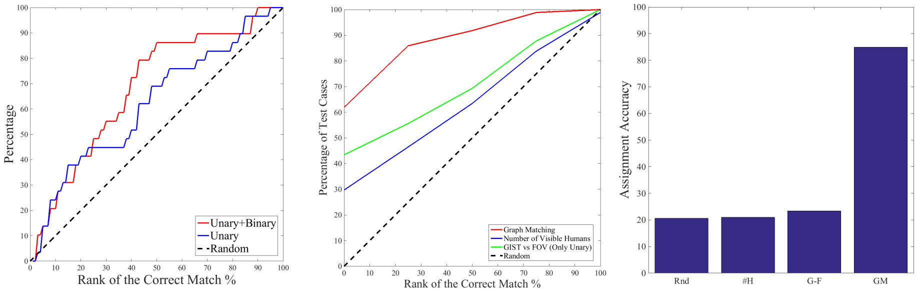
Second, knowing that a top-view video contains the viewers recording a set of egocentric videos, can we determine which viewer has recorded which video? We answer this question in sections 4.2.2 and 4.2.3.

4.2.1. Ranking Top-view Videos: We design an experiment to evaluate if our graph matching score is a good measure for the similarity between the set of egocentric videos and a top-view video. Having a set of egocentric videos from the same test case (recorded in the same environment), and 50 different top-view videos (from different test cases), we compare the similarity of each of the top-view graphs to the egocentric graph. After computing the hard assignment for each top view video(resulting in the assignment vector x), the score $x^T Ax$ is associated to that top-view video. This score is effectively the summation of all the similarities between the corresponding nodes and edges of the two graphs. Using this score rank all the top-view videos. The ranking accuracy is measured by measuring the rank of the ground truth top-view video, and computing the cumulative matching curves shown in figure 7(a). The blue curve shows the ranking accuracy when we compute the scores only based on the unary features. The red curve shows the ranking accuracy when we consider both the unary and pairwise features for performing graph matching. The dashed black line shows the accuracy of randomly ranking the top-view videos. It can be observed that both the blue and red curves outperform the random ranking. This shows that our graph matching score is a meaningful measure for estimating the similarity between the two graphs. In addition, the red curve, outperforming the blue curve shows the effectiveness of our pairwise features. In general, this experiment answers the first question. We can in fact use the graph matching score as a cue for narrowing down the search space among the top-view videos, for finding the one corresponding to our set of the egocentric cameras.

4.2.2. Viewer Ranking Accuracy: We evaluate our soft assignment results, in terms of ranking capability. In other words, we can look at our soft assignment as a measure to sort the viewers in the top-view video based on their assignment probability to each egocentric video. Computing the ranks of the correct matches, we can plot the cumulative matching curves to illustrate their performance.

We compare our method with three baselines in figure 7 (b). First, random ranking (dashed black line), in which for each egocentric video we randomly rank the viewers present in the top-view video. Second, sorting the top-view viewers based on the similarities of their 1D unary features to the 1D unary features of each egocentric camera (i.e., number of visible humans illustrated by the blue curve). Third, sorting the top-view viewers based on their 2D unary feature (GIST vs. FOV, shown by the green curve). Note that here (the blue and green curves), we are ignoring the pairwise relationships (edges) in the graphs. The consistent improvement of our method (red curve) over the baselines, justifies the effectiveness of our representation, and shows the contribution of each stage.

4.2.3. Assignment Accuracy: In order to answer the second question, we need to evaluate the accuracy of our method in terms of node assignment accuracy. Having a set of egocentric videos and a top-view video, which we know contains the egocentric viewers, we evaluate the percentage of the egocentric videos which were correctly matched to their corresponding viewer. We evaluate the hard-assignment accuracy for our method and compare it with three baselines in figure 7(c). First, random assignment (Rnd), in which we randomly assign each egocentric video to one of the visible viewers in the top-view video. Second, performing Hungarian bipartite matching only on the 1D unary feature which is the count of visible humans over times denoted as H. Third, performing



(a) (b) (c)

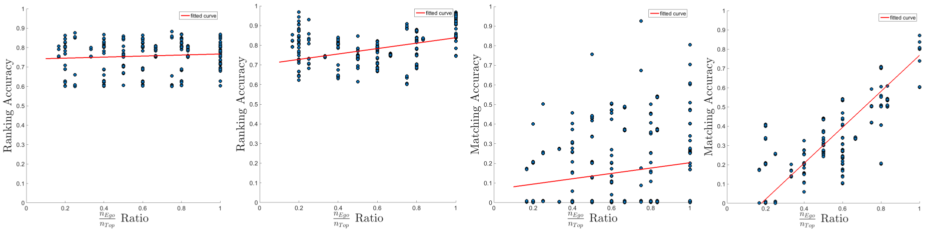
Fig. 7: (a) shows the cumulative matching curve for ranking top-view videos. The blue curve shows the accuracy if we only consider the node similarities. The red curve shows the accuracy if we consider both node, and edge similarities in the graph matching and therefore the ranking process. (b) shows the cumulative matching curve for ranking the viewers in the top-view video. The red, green and blue curves belong to ranking based on spectral graph matching scores, cross correlation between the 2D, and cross correlation between the 1D unary scores, respectively. The dashed black line shows random ranking accuracy (c) shows the assignment accuracy based on randomly assigning, using the number of humans, using unary features, and using spectral graph matching.

Hungarian bipartite matching only on the 2D unary feature (GIST vs. FOV, denoted as G-F), ignoring the pairwise relationships (edges) in the graphs.

The consistent improvement of our method using both unary and pairwise features in graph matching (denoted as GM) over the baselines shows the significant contribution of pairwise features in the assignment accuracy. As a result, the promising accuracy acquired by graph matching answers the second question. Knowing a top-view camera is capturing a set of egocentric viewers, we can use visual cues in the egocentric videos and the top-view video, to decide reliably which viewer is capturing which egocentric video.

4.2.4. Effect of Number of Egocentric Cameras: In sections 4.2.2 and 4.2.3, we evaluated the performance of our method given all the available egocentric videos present in each set as the input to our method. In this experiment, we compare the accuracy of our assignment and ranking framework as a function of the completeness ratio ($\frac{n_{Ego}}{n_{Top}}$) of our egocentric set. Each of our sets contain $3 < N^t < 11$ viewers in the top-view camera, and $2 < N^e < 8$ egocentric videos. We evaluated the accuracy of our method and baselines when using different subsets of the egocentric videos. A total of $2^{N^e} - 1$ non-empty subsets of egocentric videos is possible depending on which egocentric video out of N^e are included (all possible non-empty subsets). We evaluate our method on each subset separately.

Figure 8 illustrates the assignment and ranking accuracies versus the ratio of the available egocentric videos to the number of visible people in the top-view camera. It shows that as the completeness ratio increases, the assignment accuracy drastically improves. Intuitively, having more egocentric cameras gives more information about the structure of the graph (by providing more pairwise terms) which leads to improvement in the spectral graph matching and assignment accuracy.



(a) (b) (c) (d)

Fig. 8: Effect of the relative number of egocentric cameras referred to as completeness ratio ($\frac{n_{Ego}}{n_{Top}}$). (a) shows the ranking accuracy vs $\frac{n_{Ego}}{n_{Top}}$, only using the unary features. (b) shows the same evaluation using the graph matching output. (c) shows the accuracy of the hard assignment computed based on Hungarian bipartite matching on top of the unary features, and (d) shows the hard-assignment computed based on the spectral graph matching.

5 Conclusion and Discussion

In this work, we addressed two main questions regarding relating multiple egocentric videos to a single top-view video. First, can we tell if a set of egocentric videos belong to a set of humans present in a top-view video? And second, If we know they do, can we identify them? We proposed a framework to explore these possibilities.

Our experiments suggest that capturing the pattern of change in the content of the egocentric videos, along with capturing the relationships among them can help to identify the viewers in top-view. To do so, we utilized a spectral graph matching technique. We showed that the graph matching score, is a meaningful criteria for narrowing down the search space in a set of top-view videos. Further, the assignment found by our framework is capable of associating egocentric videos to the viewers in the top-view camera. We conclude that meaningful features can be extracted from single, and pairs of egocentric camera(s), simply based on global scene gist of the content of the camera and incorporating the temporal information of the video(s).

Our work helps relate two sources of information which so far have been studied in isolation and infer new insights about the visual world from different perspectives. For future, we will consider a more general case of this problem by assigning multiple egocentric viewers to viewers in multiple top-view cameras. We will also investigate more diverse scenarios and conditions.

References

1. Fathi A, Farhadi A, R.J.: Understanding egocentric activities. Computer Vision (ICCV), 2011 IEEE International Conference on. IEEE (2011)
2. Fathi A, Li Y, R.J.: Learning to recognize daily actions using gaze. Computer VisionECCV (2012)
3. Bettadapura, Vinay, I.E., Pantofaru., C.: Egocentric field-of-view localization using first-person point-of-view devices. Applications of Computer Vision (WACV), IEEE Winter Conference on. (2015)
4. Egozi, Amir, Y.K., Guterman., H.: A probabilistic approach to spectral graph matching. Pattern Analysis and Machine Intelligence, IEEE Transactions on (2013)
5. Dicle, Caglayan, O.C., Sznaier., M.: The way they move: Tracking multiple targets with similar appearance. Proceedings of the IEEE International Conference on Computer Vision (2013)
6. Kanade, T., Hebert., M.: First-person vision. Proceedings of the IEEE 100.8 (2012)
7. Betancourt A, Morerio P, R.C.R.M.: The evolution of first person vision methods: A survey. Circuits and Systems for Video Technology, IEEE Transactions on (2015)
8. Fathi, Alireza, X.R., Rehg., J.M.: Learning to recognize objects in egocentric activities. Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference On (2011)
9. Lu, Z., Grauman., K.: Story-driven summarization for egocentric video. Computer Vision and Pattern Recognition (CVPR), IEEE Conference On (2013)
10. Li, Y., Fathi, A., Rehg, J.: Learning to predict gaze in egocentric video. In: Proceedings of the IEEE International Conference on Computer Vision. (2013) 3216–3223
11. Polatsek, P., Benesova, W., Paletta, L., Perko, R.: Novelty-based spatiotemporal saliency detection for prediction of gaze in egocentric video
12. Borji, A., Sihite, D.N., Itti, L.: What/where to look next? modeling top-down visual attention in complex interactive environments. Systems, Man, and Cybernetics: Systems, IEEE Transactions on **44**(5) (2014) 523–538
13. Alahi, Alexandre, M.B., Kunt., M.: Object detection and matching with mobile cameras collaborating with fixed cameras. Workshop on Multi-camera and Multimodal Sensor Fusion Algorithms and Applications-M2SFA2 (2008)
14. Alahi A, Marimon D, B.M.K.M.: A master-slave approach for object detection and matching with fixed and mobile cameras. InImage Processing, 2008. ICIP 2008. 15th IEEE International Conference (2008)
15. Ferland F, Pomerleau F, L.D.C.M.F.: Egocentric and exocentric teleoperation interface using real-time, 3d video projection. InHuman-Robot Interaction (HRI), 2009 4th ACM/IEEE International Conference on (2009)
16. Park, Hyun, E.J., Sheikh., Y.: Predicting primary gaze behavior using social saliency fields. Proceedings of the IEEE International Conference on Computer Vision. (2013)
17. Hoshen, Yedid, G.B.A., Peleg., S.: Wisdom of the crowd in egocentric video curation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2014 (2014)
18. Fathi, Alireza, J.K.H., Rehg., J.M.: Social interactions: A first-person perspective. Computer Vision and Pattern Recognition (CVPR), IEEE Conference on. (2012)
19. Yan, Yan, e.a.: Egocentric daily activity recognition via multitask clustering. Image Processing, IEEE Transactions on (2015)

20. Damen D, Leelasawassuk T, H.O.C.A.M.C.W.: You-do, i-learn: Discovering task relevant objects and their modes of interaction from multi-user egocentric video. BMVC (2014)
21. Cheng DS, Cristani M, S.M.B.L.M.V.: Custom pictorial structures for re-identification. BMVC (2011)
22. Bak S, Corvee E, B.F.T.M.: Multiple-shot human re-identification by mean riemannian covariance grid. InAdvanced Video and Signal-Based Surveillance (AVSS), 8th IEEE International Conference on (2011)
23. Bazzani L, Cristani M, M.V.: Symmetry-driven accumulation of local features for human characterization and re-identification. omputer Vision and Image Understanding. (2013)
24. Cheng DS, Cristani M, S.M.B.L.M.V.: Head motion signatures from egocentric videos. InComputer Vision–ACCV. Springer International Publishing. (2014)
25. Yonetani, Ryo, K.M.K., Sato., Y.: Ego-surfing first person videos. Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on. IEEE, (2015)
26. Zamir, A.R., Ardeshir, S., Shah, M.: Robust refinement of gps-tags using random walks with an adaptive damping factor. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR). (2014)
27. Kiefer, Peter, I.G., Raubal., M.: Where am i? investigating map matching during selflocalization with mobile eye tracking in an urban environment. Transactions in GIS 18.5 (2014)
28. Shervin Ardeshir, Amir Roshan Zamir, A.T., Shah., M.: Gis-assisted object detection and geospatial localization. In European Conference on Computer VisionECCV (2014) 602–617
29. Shervin Ardeshir, K.M.C.S., Shah., M.: Geo-semantic segmentation. (2015) Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.
30. Torralba, A.: Contextual priming for object detection. In: International Journal of Computer Vision, Vol. 53(2), 169-191. (2003)
31. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part based models. IEEE Transactions on Pattern Analysis and Machine Intelligence **32**(9) (2010) 1627–1645