

Learning Scene Semantics Using Fiedler Embedding

Jingen Liu

Dept. of EECS, University of Michigan at Ann Arbor
liujg@eecs.umich.edu

Saad Ali

Robotics Institute, Carnegie Mellon University
saad@cs.cmu.edu

Abstract

We propose a framework to learn scene semantics from surveillance videos. Using the learnt scene semantics, a video analyst can efficiently and effectively retrieve the hidden semantic relationship between homogenous and heterogeneous entities existing in the surveillance system. For learning scene semantics, the algorithm treats different entities as nodes in a graph, where weighted edges between the nodes represent the "initial" strength of the relationship between entities. The graph is then embedded into a k -dimensional space by Fiedler Embedding.

1. Introduction

Learning scene semantics is of fundamental importance for modern intelligence surveillance system. Specifically, the discovery of explicit and implicit relationships between entities such as cameras, entry-exit points, dominant paths, classes of objects, time of the day, etc. (as shown in Fig. 1), is critical for understanding semantics of the scene. The semantics of the given scene consists of homogenous and heterogeneous relationships resulting from non-trivial correlations between these entities. For example, a non-trivial semantic in the scene shown in Fig. 1 can be that cars entering from the entry points N1 and N2 in the Field of View (FOV) of camera 5 always leave from the exit point X5. This implies that N1 and N2 are semantically related to each other. While other entry points, say N1 and N3, may not have such a semantic relationship as cars using these entry points always use different exit points. It is of prime importance to discover such relationships, if one wants to have a deeper understanding of what is happening in the scene.

Much of the previous research has defined the problem of learning scene semantics as the problem of locating entry and exit points, dominant paths, junctions, stop zones etc. in the given scene, which collectively define the structure of the scene. A detailed review of models that have been employed for learning and understanding scene activities in this way is presented in [2]. Many methods have been proposed to learn dominant paths or trajectories in a scene [3-10]. For instance, [3] used a vector quantization based neural network framework to learn physical paths taken by the pedestrians in a scene. And [6-9] proposed to cluster trajectories in different ways. In addition

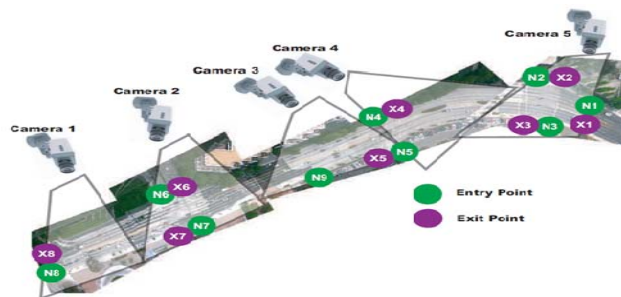


Figure 1: Representation of diverse set of entities that are present in a multi-camera surveillance system.

to learning dominant paths, detection of entry and exit points is also important for a detail description of the scene structure. In this regard, [11] proposed a more detailed approach capable of locating not only the dominant paths, but also the entry exit points and junctions, in an unsupervised manner. In [12-14], a hidden Markov model based scheme for learning sources and sinks is presented.

Note that all these approaches use the traditional definition of "scene semantics" which assumes it to be equivalent to the "scene structure". However, we contest that semantics of a visual scene does not consist solely of "scene structure". Rather, it also encapsulates the interactions between scene entities such as dominant paths, entry and exit points, junctions, cameras' FOVs etc. Unfortunately, a considerably less amount of research has been undertaken to provide surveillance systems with the ability to learn these interactions. Another important point is that, none of the above mentioned methods can handle all types of scene entities within the same framework. Usually ad-hoc practices or heuristics are employed to manually encode semantics or interpretation rules.

In this paper our aim is to seek a framework that will allow learning of explicit and implicit relationships between *different classes* of entities in a principled manner. We achieve this by proposing an algorithm based on the concept of Fiedler Embedding [1], which is an algebraic method that explicitly optimizes the closeness criteria. Unlike LSA, which can only deal with two classes of entities, our framework can embed multiple different types of (heterogeneous) entities into a common Euclidean space, and thus enables the use of simple Euclidean distances for

discovering relationships. To the best of our knowledge, this will be the first such algorithm.

2. Multi-Camera Surveillance Dataset

To show the applicability of our framework, we make use of the multi-camera dataset generated by the Next Generation Simulation (NGSIM) program [17]. It contains videos of traffic flow at a portion of Lankershim Boulevard in LA, California. The data was collected using five cameras that are mounted on a building. The FOV of each camera has a small overlap with the FOV of the adjacent cameras. The cameras are numbered 1 to 5, with the camera 1 recording the southernmost, and camera 5 recording the northernmost section, as shown in Fig. 1.

Once the videos are collected, vehicle trajectory data is transcribed using the Next Generation Vehicle Interaction and Detection Environment for Operations software [17]. This program was used to automatically detect and track most vehicles from the video and transcribe the trajectory data to a database. Vehicles maintain their labels across different cameras. The data provides X, Y coordinates of each vehicle in relative space and in the California State Plane Coordinate System, Zone 5, NAD83. The transcribed data represents 32 minutes of video in total, segmented into two periods. For each tracked vehicle we have 24 pieces of information, but the ones that are relevant to our algorithm include: vehicle id, trajectory of the vehicle in state plane coordinate system, vehicle type (1 - motorcycle, 2 - auto, 3 - truck), lanes used by the vehicle, entry and exit points of each vehicle and camera FOV in which the vehicle was visible at any time during its passage through the scene.

3. Our Proposed Framework

3.1. Creating Graph Network of Scene Entities

Let $G = (V, E)$, where V is a set of vertices and E is a set of edges, represents a graph consisting of nodes representing different classes of. If two features i and j are related to each other, then we have an edge (i, j) with a non-negative weight $w_{i,j}$ between them. The more similar the entities are to each other, the higher the weight. Our goal is to embed this graph into a low-dimensional Euclidean space, so that vertices with a high weight between them become closer to each other. As a result, spatial proximity can be used as a way to identify vertices that are similar to each other even if they do not have a direct edge between them (the implicit relationship). The L Matrix of this graph can be described as:

$$L(i, j) = \begin{cases} -w_{ij} & \text{if } e_{ij} \in E \\ \sum_k w_{ik} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where L is symmetric and positive semi-definite. Note that L is nothing but the negative of the matrix of weights with diagonal values chosen to make the row-sums zero. In the following sub-section we address the details of L matrix of the scene graph network.

3.2 Constructing Scene Laplacian Matrix

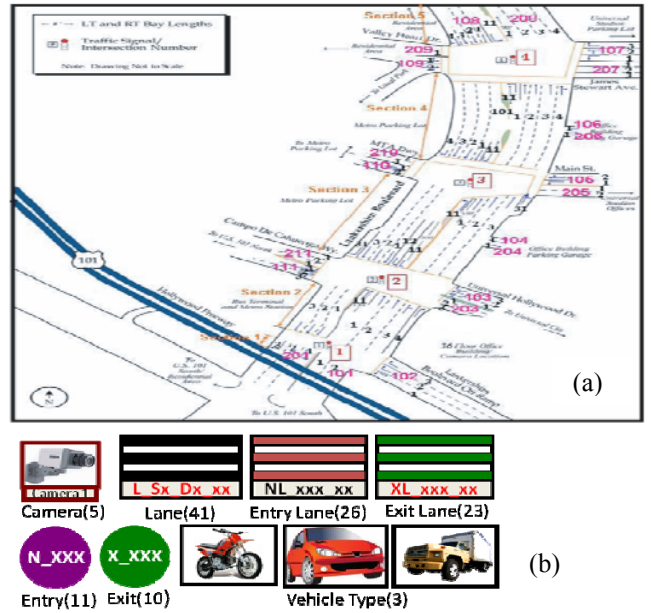


Figure 2: (a) The schematic diagram of the portion of the Lankershim Blvd. observed by five cameras. The schematics are provided by the FHWA. Different entities that are embedded into a k -dimensional space are labeled as: (1) Entry points: 101~111, (2) Exit points: 201~211, (3) FOVs: “Section 1~5” and (4) lane numbers are incremented from left most lane to the right most. (b) The graphical symbols used to represent the scene entities. The label of the symbol determines the identity of the entity. The caption underneath each symbol shows the total number of that type of entity present in the scene. We employ these symbols to qualitatively demonstrate the semantic relationships between scene entities.

For the scene under-consideration, the semantics are defined in terms of the following entities:

Entry Points: specific locations in a scene used by vehicles to enter the FOVs. There are 11 entry points observed by five cameras, labeled with number 101 through 111. These entry points are shown in the schematic diagram (Fig. 2) of the study area. We use symbol N_XXX to represent an entry point, where XXX takes values between 101 and 111.

Exit Points: specific locations in a scene used by vehicles to exit the FOVs. There are 10 exit points in the given scene which are numbered 201 through 211. They are also shown in the schematic diagram (Fig. 2) of the study area. Its symbol is X_XXX , where $XXXX$ takes values between 201 and 211.

FOV of Cameras: the portions of the scene observed by each camera. There are five such entities labeled Section 1 through 5 in Fig. 2. All tracked vehicles are assigned the label of the FOV in which they are visible during their passage through the scene. We use symbol $Camera-X$ for representing the FOV.

Lane: the lanes of Lankershim Boulevard. Intuitively, the lanes can be considered as the dominant paths present in the scene that are used by vehicles. Within each section (FOV), the lane number is incremented from the left most side as shown in Fig. 2. We use symbol $L_SX_DX_XX$ for representing a lane, where SX (X varies from 1 to 5) represents the section number, DX (X is either 0 or 1) represents whether the lane is north-bound and south-

bound, and XX (varies from 01 to 31) represents the lane number within each section.

Entry (Exit) Lane: the lane number used by each vehicle to enter (exit) into the scene. It is considered different from the ‘‘Lane’’ entity, as it has a many to one relationship with the ‘‘entry (exit) points’’ of the scene. Symbol NL_XXX_XX (XL_XXX_XX) is used to represent an entry (exit) lane, where XXX (101 ~ 111 for entry and 201 ~ 211 for exit) is the identity of the entry (exit) point while XX (01 ~ 03) is the entry (exit) lane.

Vehicle Type: Vehicles are classified into three categories: 1) automobile, 2) truck or buses, and 3) motorcycles. We use symbol VX for vehicle type.

The graphical symbols used to represent these entities are shown in Fig 3(b). For embedding the scene entities into a common k -dimensional space, we construct the scene Laplacian matrix L_S . For the scenario at hand, L_S is a 7×7 symmetric block matrix represented as follows:

$$L_S = \begin{pmatrix} D_1 & S_{(X,N)}^T & S_{(F,N)}^T & S_{(L,N)}^T & S_{(NL,N)}^T & S_{(XL,N)}^T & S_{(V,N)}^T \\ S_{(X,N)} & D_2 & S_{(F,X)}^T & S_{(L,X)}^T & S_{(NL,X)}^T & S_{(XL,X)}^T & S_{(V,X)}^T \\ S_{(F,N)} & S_{(F,X)} & D_3 & S_{(L,F)}^T & S_{(NL,F)}^T & S_{(XL,F)}^T & S_{(V,F)}^T \\ S_{(L,N)} & S_{(L,X)} & S_{(L,F)} & D_4 & S_{(NL,L)}^T & S_{(XL,L)}^T & S_{(V,L)}^T \\ S_{(NL,N)} & S_{(NL,X)} & S_{(NL,F)} & S_{(NL,L)} & D_5 & S_{(XL,NL)}^T & S_{(V,NL)}^T \\ S_{(XL,N)} & S_{(XL,X)} & S_{(XL,F)} & S_{(XL,L)} & S_{(XL,NL)} & D_6 & S_{(V,XL)}^T \\ S_{(V,N)} & S_{(V,X)} & S_{(V,F)} & S_{(V,L)} & S_{(V,NL)} & S_{(V,XL)} & D_7 \end{pmatrix}$$

Each block matrix is encoding pair-wise similarities between two entities. Here D_i is block matrix of homogeneous relationships, such as *entry point-entry point*, *lane-lane*, and so on. And, off-diagonal matrices (S) are capturing pair-wise similarities between the heterogeneous entities. Symbols X, N, F, L, NL, XL, and V in the subscript refer to *entry point*, *exit point*, *FOV*, *lane number*, *entry lane*, *exit lane*, and *vehicle type*, respectively. The similarity values in the diagonal matrices are computed using *histogram intersection*. For this purpose we compute a 119-bin histogram for each entity. The breakdown of the bins of the histogram is as follows: 11 entry points, 10 exit points, 26 entry lanes, 23 exit lanes, 41 lanes, 5 cameras and 3 vehicle types. The value in each bin is populated by using the co-occurrence between the entities in terms of vehicle identities that they share. Zeros are inserted into the bins corresponding to the co-occurrence between the entities of the same class. The similarity values in the off-diagonal matrices are computed by using the direct count of the vehicles that have used both entities under consideration. For instance, the similarity between the ‘‘Entry Point 1’’ and ‘‘Camera-2’’ will be the number of cars entering from the ‘‘Entry Point 1’’ and passing through the FOV of ‘‘Camera-2’’.

3.3. Fiedler Embedding

When posing the geometric graph embedding problem as an algebraic minimization problem, Fiedler Embedding algorithm seeks points in a k -dimensional space that minimize the weighted sum of the square of the edge lengths. If p_r and p_s are locations of nodes r and s , the function can be written as,

$$\text{Minimize } \sum_{(r,s) \in E} w_{rs} |p_r - p_s|^2, \quad (2)$$

where $w_{r,s}$ represents the weight between the nodes r and s . If the graph has n nodes, and the target space has dimensionality k , then the positions of the vertices can be represented by an $n \times k$ matrix X . This will imply that p_r or p_s ($r,s=1,2,\dots,n$) is a k -dimensional vector indicating the coordinate of the node in the k -dimensional space. We can prove that minimizing the above function is equivalent to minimizing the trace of $X^T L X$. Thus, our final minimization problem in terms of matrices L and X is

$$\text{Minimize Trace}(X^T L X) \quad (3)$$

with constrains (i) for $i=1,\dots,k$, $X_i^T \mathbf{1}^n = 0$, (ii) $X^T X = \Delta$. The first constraint makes the median of point sets in the embedding space to be at the origin, while the second constraint avoids the trivial solution of placing all the vertices at the origin. In the above equation $\mathbf{1}^n$ is a vector of n ones while Δ is a diagonal matrix of δ_i which are some positive values. As shown in [1], the solution to the above minimization is $X = \Delta^{1/2} [Q_2, \dots, Q_{k+1}]$, where Q is a matrix of normalized eigenvectors of L sorted in a non-decreasing order based on the eigenvalues λ_i of L . This implies that the coordinates of vertex i are just the i -th entries of eigenvectors $2, \dots, k+1$ of L . This solution is referred to as the Fiedler Embedding of the graph.

4. Experiment Results

The purpose of this experiment is to qualitatively show that our algorithm is able to discover high level semantic relationships between scene entities. The data set described in Section 3 used for this purpose. The embedding process starts by building the scene Laplacian L_S using the vehicle trajectory data provided along with the data set. The data used for computing histogram intersections and other similarity measures is summarized in the form of co-occurrence (in terms of vehicles) tables that are uploaded as part of the supplemental material. Next, the embedding is carried out by finding $k=20$ eigen vectors of L_S and computing the new coordinates of each entity.

For qualitative analysis, we retrieved the semantically similar entities by using queries based on different entity types, and using Euclidean distance as the similarity measure. The idea is that if the embedding space is meaningful then the returned nearby entities should have a verifiable semantic relationship with each other. Fig. 3 shows the results for different query runs. In Fig. 3(a), the entry point 102 (N_102) is used as the query entity, and we retrieved 10 nearest scene entities from the k -dimensional space. As per this result, the query entry point is semantically very similar to the entry point 101 in terms of the behavior of the traffic coming from these two locations. This can be verified from the schematic diagram (Fig. 2) of the scene. The north bound traffic is entering onto the Lankershim Blvd. from the entry point N_101, while N_102 is the entry point for traffic coming from Lankershim Blvd. Off-Ramp. The off-ramp has two

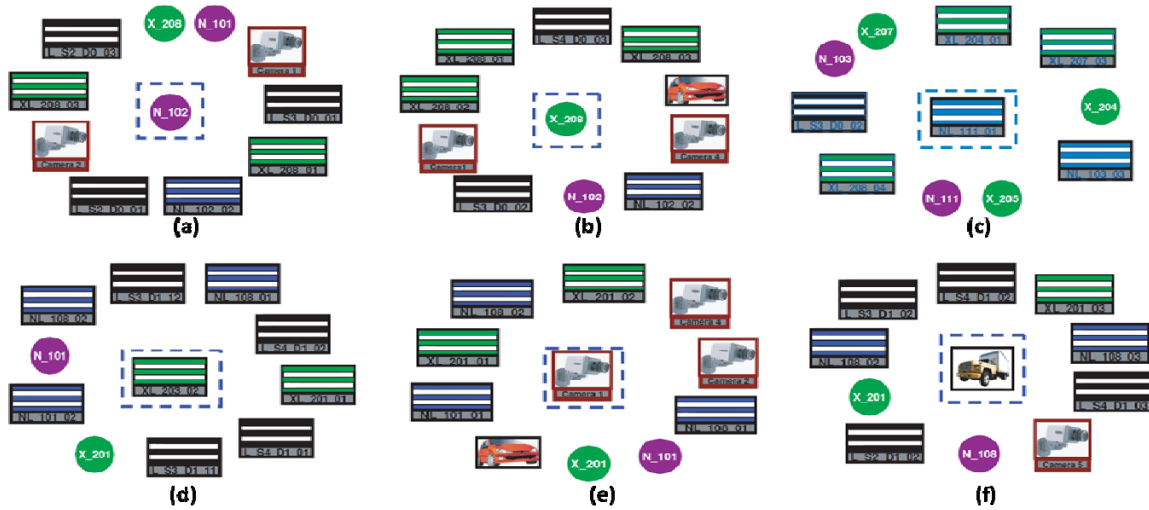


Figure 3: The results of different combinations of the query-result entities for semantic scene learning. The Query for (a) to (f) are Entry point, Exit point, Entry lane, Exit lane, Vehicle Type, and FOV of Camera respectively. The Outputs are 10 nearest scene entities.

right turn only lanes, which means most of the traffic will follow the same road conditions as the traffic coming from N_101. Thus it shows that the discovered relationship is semantically meaningful and has a correct interpretation for the given physical scene.

Furthermore, entry point N_101 is closer to exit point X_208, because most of the traffic travels straight on at this portion of the Leadership Blvd. and therefore exits from the opposite end of the study area. Further still, N_101 is closer to Camera-1 and Camera-2 in the embedding space as most of the vehicles entering from N_101 pass through the FOV of Camera-1 and Camera-2. Since there are multiple exit points before vehicles reach Camera-3, the relationship between N_101 and rest of the cameras is not that strong. Similar interpretation is associated with the other retrieved entities.

Due to the space limitation, we will discuss one other result, after which hopefully it will be easier for the reader to develop an interpretation of the remaining results using a similar logic. The next qualitative result is shown in Fig 3(f). The query is a type of a vehicle which is truck or bus (V3). The returned results show that in the given scene, trucks are semantically more related with the entry point N_101 and exit point X_201. This means that the south-bound side of Lankershim Blvd. is used more by trucks or buses during this hour of the day, as N_101 and X_201 is used by the through traffic. While within the south-bound lane trucks mostly use lane 2 and 3 as represented by the proximity of entities L_S4_D1_03, L_S4_D1_02, L_S3_D1_02, and L_S2_D1_02 in the embedding space. The reason is that the Lane 1 is for the fast moving traffic and trucks often avoid those lanes. Now if more and more trucks suddenly started using lane 1 that can be flagged as an atypical behavior requiring attention. In summary, the quality of results for semantic scene learning demonstrates that the constructed k -dimensional space is providing us meaningful information. This information can be used from abnormal event detection.

5. Conclusion

We present a principled framework to embed heterogeneous entities of a surveillance video to discover the semantic relationship between the entities. Our frame is able to retrieve all semantically similar entities given a query entity, which is of prime importance for surveillance scene understanding. We have tested our framework on NGSIM data set, and will do more experiments on a more complicated surveillance system.

References

- [1] B. Hendrickson, Latent Semantic Analysis and Fiedlder Retrieval, SIAM Linear Algebra and its Application, 2007.
- [2] H. Buxton, Generative Models for Learning and Understanding Dynamic Scene Activity, In Generative Model Based Vision Workshop, 2002.
- [3] N. Johnson et. al., Learning the Distribution of Object Trajectories for Event Recognition, IVC, 14(8), 2003.
- [4] J. H. Fernyhough et. al., Generation of Semantic Regions from Image Sequences, ECCV, 1996.
- [5] R. J. Howard et. al., Analogical Representation of Spatial Events for Understanding Traffic Behavior, ECAI, 1992.
- [6] E. B. Koller-Meier et. al., Modeling and recognition of human actions using a stochastic approach, Eur. Workshop on Advanced Video-Based Surveillance Systems, 2001.
- [7] E. Grimson et. al., Using Adaptive Tracking to Classify and Monitor Activities in a site, CVPR, 1998.
- [8] C. Stauffer et. al., Learning Patterns of Activity Using Real Time Tracking, PAMI, 22(8), 2000.
- [9] I. Junejo et. al., Multi-Feature Path Modeling for Video Surveillance, ICPR, 2004.
- [10] X. Wang et. al., Learning Patterns of Activity Using Real-Time Tracking, ECCV, 2006.
- [11] D. Makris et. al., Learning Semantic Scene Models from Observing Activities in Visual Surveillance, IEEE Transactions on Systems, Man and Cybernetics, 35(3), 2005.
- [12] C. Stauffer, Estimating tracking sources and sinks, In Proc. Event Mining Workshop, 2003.
- [13] P. Remagnino et. al., Classifying Surveillance Events from Attributes and Behavior, BMVC, 2001.
- [14] M. Walter et. al., Learning Prior and Observation Augmented Density Models for Behavior Recognition, BMVC, 1999.
- [15] <http://www.ngsim.fhwa.dot.gov/>