Context-Aware Graph Modeling for Object Search and Retrieval in a Wide Area Camera Network

Santhoshkumar Sunderrajan, Jiejun Xu*, B.S. Manjunath
Department of Electrical and Computer Engineering,
HRL Laboratories*
University of California, Santa Barbara
{santhosh,manj}@ece.ucsb.edu, jxu@hrl.com

***Abstract*—This paper addresses the problem of context-aware object search and retrieval in a wide area distributed camera network. With the proliferation of smart cameras in urban networks, it is a challenge to process this big data in an efficient manner. A novel graph based model is proposed to represent relationships, and for search and retrieval tasks. This representation exploits the fact that objects occurring in close spatial-temporal proximity are not completely independent and serve as context for each other. Additional information such as appearance and scene context can also be encoded into the graph model to improve the overall accuracy. A manifold ranking strategy is used to order the items based on similarity with an emphasis on diversity. Extensive experimental results on a ten camera network are presented.**

***Keywords*—Context-aware camera networks, Graph based ranking, Object search and Retrieval.**

I. Introduction

The advent of cheap and network enabled cameras has given opportunities for large scale deployment of smart cameras in real world applications. However, most of these applications are limited in capability due to processing power of cameras and/or network bandwidth constraints. In a wide area network for surveillance, it is expensive and inefficient to send every video frame captured at individual nodes to the central node. At the same time, a human analyst at the base station would require video frames or information in real-time to take necessary actions during crisis. In this paper, we propose a novel de-centralized graph based approach to query and retrieve objects of interest without necessitating large data flow to the central node.

Consider a fixed camera network deployed over a wide area (see Figure 8). We assume that each of the individual cameras has sufficient storage and computational capacity to store and perform basic video analysis such as object detection and tracking. We consider the following two scenarios introduced by [15]:

1) **Spatial-Temporal Browsing** (see Figure 1(a)): A user initiates a spatial-temporal browsing query by marking a region on the image plane and specifies a time interval for querying. An example query would be "Find object occurrences *related* to region A (specified by a bounding box) from camera 1 between time 9:32am and 9:34am".

2) **Object Searching** (see Figure 1(b)): Given the set of records (objects) from the browsing query, the user could select an object of interest and query the system to identify

Fig. 1: **Sample System Queries** (a) Spatial region along with the time-interval is specified as an initial query to the system. (b) Object of interest is marked by a green bounding box.

the same or related objects possessing similar spatial-temporal and appearance information. For example, "Find all objects *related* to the selected object at region C from camera 1 at time 9:33:01am". This problem is closely related to object re-identification.

To accomplish the above-mentioned goals, most of the existing methodologies assume that global trajectories of targets are available [9], [10], [12]. With the existing state of the art object detectors and trackers [18], [14], availability of accurate global trajectories is not a reasonable assumption. Using appearance based features alone, it is a challenging task to re-identify objects in multiple views due to changes in viewpoint angle, illumination and background clutter. Hence, additional spatial-temporal constraints along with contextual information are needed for re-identifying objects.

Towards achieving the above objectives, we propose a distributed surveillance system that globally models the entire network using a context-aware graph and enables users to find a set of "representative" and "diverse" snapshots to understand the network events. Objects are detected and tracked in individual camera frames. At every frame, each of the individual cameras sends an abstracted record comprising of spatial, temporal and visual appearance information about an object to the central node. At the central node, we break long object tracks into several smaller tracklets. Tracklet serves as a basic atomic unit for our analysis. The central node builds a context based time-evolving graph of the network using these tracklets. Most importantly, we propose a strategy to encode contextual information such as appearance, spatial-temporal, and scene contexts. Also, past historical data is used to model the spatial-temporal transition between two camera views. Finally, we use an off-the-shelf graph ranking algorithm to find relevant records based on the user query. Following are the main contributions of this work:

1. A robust *context-aware network graph modeling* and extensive experimentation on a large scale camera network.
2. A novel *non-parametric spatial-temporal network topology modeling*.

The rest of the paper is organized as follows: Section II describes related works on context based object re-identification and a graph-based method for visual searching. Section III describes the proposed methodology in detail. Section IV illustrates context based modeling of the network. Section V highlights the manner in which the spatial-temporal modeling of the network is performed. Section VI describes graph based ranking methodology for retrieving objects. Finally, Section VII presents experimental results from a real 10-camera outdoor network and Section VIII concludes this paper.

II. Related Work

The proposed system is closely related to the problem of object re-identification or reacquisition. There have been several distributed camera systems [8] that collect observations from remote cameras and assign a unique global object ID based on appearance and/or spatial information. Most of the appearance based object re-identification methods operate by finding the best matching criterion. For example, [9] works by jointly modeling motion and appearance while [10] learns a low-dimensional subspace of brightness transfer functions for matching. In [2], authors propose a pca-sift based vocabulary tree for re-identifying objects in a camera network setup. Interestingly, all these methods operate by comparing observations from different views in a pair wise manner. [15] proposes an approach to model the network graph using appearance and spatial temporal features. However, appearance based observation modeling does not take context into account and it suffers from viewpoint changes.

Oliva et al. [16] demonstrate that context plays a significant role in the human visual system. Ali et al. [1] proposed a tracking system that uses motion and appearance context of co-occurring objects to improve the tracking accuracy. Zhu et al. [22] proposed a context-aware activity recognition and anomaly detection system leveraging spatial-temporal and scene contexts. This paper extends our earlier work [20] where we first introduced contextual links into a graph model. We propose a graph based methodology to represent relationships among camera observations. Contextual information such as appearance, spatial and scene are used to weight the graph edges. Graph based modeling provides an efficient tool for combining information from multiple sources. Tong et al. [19] provide an efficient strategy to fuse information from multiple sources such as image level features and text based annotations. Similar to [15], [20], we pose the problem of user interaction as a unified ranking problem for identifying *representative* and *diverse* snapshots that match the user query.

III. Proposed Method

Consider a network of C distributed static cameras with embedded storage and computing power. It is assumed that each camera node can detect and track objects in its field of view, and send the appearance and other relevant object information to a central node. The central node builds a dynamic graph as it receives information from camera nodes. The time-evolving graph models spatial and temporal relationship between different cameras that can then be used to answer queries such as: *"Find observations related to region A of camera 1 between time t_1 and t_2"*, the central node performs a ranking based on the time-evolving graph built at that time instance and, requests candidate frames for visualization. In this way, no raw image or video data is sent back and forth over the network and also the system operates within the network bandwidth constraints.

A. Real-time Distributed Detection and Tracking

At each camera node, moving objects are detected using connected component analysis based on the foreground pixels obtained using a background subtraction technique [13]. These foreground blobs are individually tracked using a mean-shift based object tracker [5]. A unique object ID is assigned to each of the tracked objects. For each frame, an individual camera generates a record for the detected/tracked object and sends it to the central node over the network [15]. Each record is represented using the following information: camera ID, time, object's bounding box on the image plane, a 64-bin normalized hue histogram as appearance representation, object type, speed and direction of motion.

B. Graph Based Modeling of a Camera Network

At time instance "t", let N be the total number of detected/tracked objects in the entire camera network and more objects could be added as they are detected/tracked. We propose a graph based framework wherein we construct a network graph $G = (V, E)$ of the tracked objects. The nodes $v_i \in V, 1 \leq i \leq N$ are the set of tracklets extracted from the entire network. We assumed that long-term tracking is not possible due to various reasons appearance changes, illumination effects and hence we break object trajectories into many shorter temporal windows of size $F = 4$ frames. The edges $e_{ij} \in E$ represent connections between the nodes. The edge weights $W_{i,j}$ are computed by taking contextual and spatial-temporal topology information of the network into account. At first, the proposed methodology is developed with a snapshot of the graph at time t, more nodes can be added to this time evolving graph in a trivial manner. If two nodes v_i and v_j are from the same camera and belong to the same object, we set $W_{ij} = \kappa$, where κ is a constant (in our experiments, we set $\kappa = 1$). If the nodes belong to tracklets from two different cameras, context-aware observation and spatial-temporal topology information are used to compute the weights.

$$W_{i,j} = W_{i,j}^{Observation} + \eta W_{i,j}^{Topology} \tag{1}$$

IV. Context-aware Observation Modeling

We use appearance, spatial-temporal and scene contexts to compute the observation weight:

$$\begin{aligned} W_{i,j}^{Observation} = {} & \alpha W_{i,j}^{Appearance} + \beta W_{i,j}^{Spatial-Temporal} \\ & + \gamma W_{i,j}^{Scene} \end{aligned} \tag{2}$$

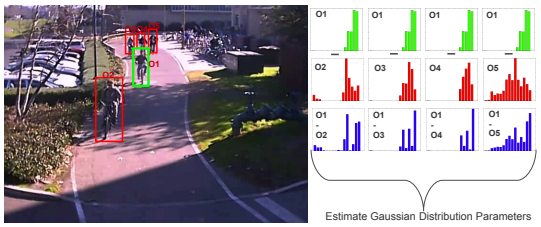

Fig. 2: **Appearance Context:** An object of interest is marked with a green bounding box. Color histogram is computed for every object in the scene and a feature difference matrix is computed with respect to the object of interest.

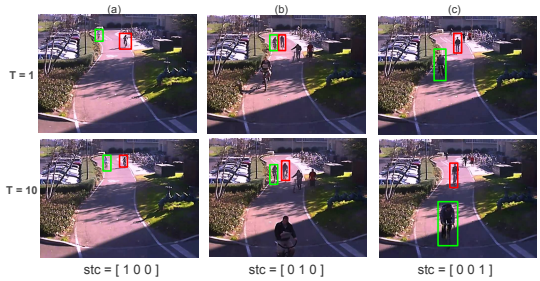

Fig. 3: **Spatial-Temporal Context:** An object of interest is marked with a green bounding box and a co-occurring object is marked with a red bounding box. (a) Object of interest moves towards the co-occurring object. (b) Object of interest moves along with the co-occurring object. (c) Object of interest moves away from the co-occurring object.

A. Appearance Context

Appearance context models the inter-object appearance variations. Object association is computed in a robust manner by using the knowledge of appearance variations with respect to other co-occurring objects in the scene when the appearance cues are weak and inconsistent between two different views. Let $O = \{O_i\}_{i=1}^{M}$ be the set of tracklets in the first camera view and M be the number of objects. Let each tracklet be represented by a $D = 64$-dimensional normalized hue histogram $H_i = [h_1 \dots h_D]^T$. At a given time t, we compute the feature difference matrix Λ_i for a tracklet O_i with respect to other tracklets $\{O_j\}_{j \neq i}$ in a given view:

$$\Lambda_i = [|H_i - H_1| \dots\dots |H_i - H_j|] \tag{3}$$

This matrix Λ_i captures the inter-object appearance variations similar to [1]. We assume that the difference matrix follows Gaussian distribution and estimate the mean (μ_i) and the covariance (Σ_i) using standard techniques. In the neighboring camera view, let $O^{'} = \{O_i^{'}\}_{i'=1}^{M^{'}}$ be the set of observed tracklets and $M^{'}$ is the number of tracklets. We find the feature difference matrix $\Lambda_{i'}^{'}$ and the corresponding Gaussian distribution parameters ($\mu_{i'}$ and $\Sigma_{i'}$). Figure 2 demonstrates the appearance context. We assume that the objects in different views have a similar feature difference distribution and the appearance based weighting is given by:

$$W_{i,i'}^{Appearance} = \frac{1}{2}\left(\sum_{d=1}^{D}\sqrt{H_i(d)H_{i'}(d)} + \phi(\Lambda_i, \Lambda_{i'}^{'})\right) \tag{4}$$

where $\phi(\Lambda_i, \Lambda_{i'}^{'})$ is the multi-variate Bhattacharya distance between the two feature difference distributions in two different views and it is given by:

$$\begin{aligned}\phi(\Lambda_i, \Lambda_{i'}^{'}) =& \frac{1}{8}(\mu_i - \mu_{i'})^T\Sigma^{-1}(\mu_i - \mu_{i'}) + \\ & \frac{1}{2}\ln\left(\frac{\det(\Sigma)}{\sqrt{\det(\Sigma_i)\,\det(\Sigma_{i'})}}\right)\end{aligned} \tag{5}$$

where $\Sigma = \frac{(\Sigma_i + \Sigma_{i'})}{2}$.

B. Spatial-Temporal Context

Spatial context refers to the spatial inter-relationship between co-occurring objects. There exist other co-occurring objects in the scene that exhibit strong motion correlation with the object of interest. Relative configurations between these sets of observed objects in the given view can serve as a strong cue for association between two neighboring camera views. Similarly, the temporal context captures relative changes in two tracklets with respect to time. Two or more objects maintaining a constant distance between each other or converging towards each other or diverging away from each other are some of the examples of spatial-temporal context.

1) Spatial-Temporal Context based graph model: Given the set of tracklets $O = \{O_i\}_{i=1}^{M}$, let F be the temporal size of the tracklets (in our experiments, we set $F = 4$ frames), we compute the pair-wise Euclidean distance $d_{ij}^{(f)}$ between the tracklets at time t where $j \neq i$. We use three bins to represent spatial-temporal context at frame f i.e. $stc_{ij}(f) = \left[\mathcal{I}(d_{ij}^{(f)} < d_{ij}^{(f-1)}), \mathcal{I}(d_{ij}^{(f)} = d_{ij}^{(f-1)}), \mathcal{I}(d_{ij}^{(f)} > d_{ij}^{(f-1)})\right]$ where $\mathcal{I}$ is an indicator function. The spatial-temporal context between two tracklets is given by $STC_{ij} = \frac{1}{(F-1)}\sum_{f=2}^{F} stc_{ij}(f)$. The overall spatial context for the tracklet O_i with respect to other tracklets in the scene is given by:

$$\Psi_i = \frac{1}{(M-1)}\sum_{j} STC_{ij} \tag{6}$$

Figure 3 illustrates spatial-temporal context. Note that the elements of unnormalized Ψ_i sum up to the number of co-occurring tracklets and each bin specifies the number of tracklets having certain spatial-temporal behavior with respect to the tracklet of interest. We compute a similar spatial context vector $\Psi_{i'}^{'}$ for tracklets in the neighboring views with respect to the tracklet $O_{i'}$. Thus,

$$W_{i,i'}^{Spatial-Temporal} = 1 - emd(\Psi_i, \Psi_{i'}^{'}) \tag{7}$$

where *emd* is the Earth mover's distance.

The spatial context based weighting has a tendency to give a strong confidence value if the same set of objects behave similarly in neighboring views. This might not be true in case of the camera views that are geographically far apart and objects might disappear and new objects might appear in the long run. Also, spatial-temporal context is more relevant when

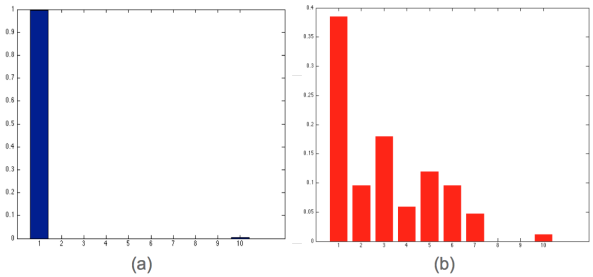

Fig. 4: (a) Average histogram of velocity magnitudes for pedestrians. (b) Average histogram of velocity magnitudes for bikers. As illustrated, pedestrians and bikers have distinctive velocity distributions.

there is sufficient overlap between two camera views. With our earlier assumption that the entire network is time synchronized, we set $W^{ST}_{i,i'} = 0$ if the time delay $T_d(i,i') > \tau_2$, where τ_2 is the time delay threshold (in our experiments we set $\tau_2 = 300$ seconds obtained from median of time delays between two geographically nearby camera views).

C. Scene Context

Scene context models the relationship between objects and the scene. The object dynamics at various parts of the scene is controlled by the various scene components. We define various scene related attributes to incorporate scene context into the graph model. For each tracklet, the histogram of optical flow (HOOF) is computed as described in [3]. h^{of}_i is a $D1$ dimensional histogram of optical flow for the tracklet O_i (in our experiments, we set $D1 = 8$). We estimate the direction θ_i and speed α_i (i.e., optical flow magnitude) of the tracklet O_i by:

$$\theta_i = \arg \max_{j \in 1,2,..D1} h^{of}_i(j)$$
$$\alpha_i = h^{of}_i(\theta_i). \quad (8)$$

1) Scene Context based graph model: The set of attributes ($G = 4$) defined in Table I is computed for each tracklet O_i. The object category per view is learned using average histogram of velocity magnitudes from the training data and used histogram intersection kernel to classify. Figure 4 shows the average normalized velocity histogram for pedestrians and bikers in one of the camera views. Let $A = \{A_g\}$ be the set of attributes, where $g \in \{1 \dots G\}$ is the attribute index and each of the sub-attributes takes a value of 1 (true) or 0 (false). Let n_g be the number of sub-attributes. For example, $A1$ has two sub-attributes i.e., speed greater than a predefined threshold or speed lesser than a predefined threshold. In our experiments, the aspect ratio threshold was to 0.5 (we chose the aspect ratio threshold to be the mid point with the maximum being 1.0) and speed threshold was set to 1.0 (the speed threshold was chosen by splitting the training data into two halves). For a tracklet O_i, the scene context attribute vector is given by $sc_i(g,a) = \mathcal{I}(A_g, a)$ where $a \in \{1 \dots n_g\}$. The overall scene context is obtained by concatenating individual attributes i.e. $sc_i = [sc_i(1) \oplus sc_i(2) \dots \oplus sc_i(G)]$. The scene context based weighting between sc_i and $sc_{i'}$ is given by:

TABLE I: List of Scene Context Attributes.

Attribute	Description
$A1$	Speed is greater than a predefined threshold; speed is smaller than a predefined threshold.
$A2$	Aspect ratio is greater than a predefined threshold; aspect ratio is smaller than a predefined threshold.
$A3$	Tracklet of interest is a biker; tracklet of interest is a pedestrian.
$A4$	Direction of motion [north, north-east, east, south-east, south, south-west, west, north-west].

$$W^{Scene}_{i,i'} = \frac{1}{G} \sum_{g=1}^{G} \frac{1}{n_g} \sum_{a=1}^{n_g} \mathcal{I}(sc_i(g,a) = sc_{i'}(g,a)) \quad (9)$$

V. Spatial-Temporal Topology Modeling between Camera Views

The image plane is divided into 8×6 blocks to model the spatial-temporal topology between two cameras. Let T_d be the time delay for an object to travel across any two blocks between two cameras. Let $\mathbf{z} = [X_i, X_{i'}, T_d]$, where X_i and $X_{i'}$ are the block centroids in $2d$ image plane. Assuming that the ground truth associations are available between two different views, a five dimensional model is built to estimate the transition probability density function $p^{ST}(X_i, X_{i'}, T_d)$ between two blocks. We non-parametrically model the pdf $p^{ST}(\mathbf{z})$ using a Kernel density estimator with K training samples:

$$p^{ST}(\mathbf{z}) = \frac{1}{K} |\mathbf{H}|^{-\frac{1}{2}} \sum_{k=1}^{K} \mathbf{K}(\mathbf{H}^{-\frac{1}{2}}(\mathbf{z} - \mathbf{z}_k)) \quad (10)$$

where $\mathbf{K}$ is the kernel function given by the product of Gaussian kernels in each dimension and $\mathbf{H}$ is a symmetric positive definite bandwidth matrix [7], [17]. Figure 6 demonstrates spatial-temporal topology modeling between two camera views. Given this topology model, we can calculate the probability that tracklets O_i and $O'_{i'}$ in two different views belong to the same object based on the time delay between $block(O_i)$ and $block(O'_{i'})$.

$$W^{Topology}_{i,i'} = p^{ST}(\mathbf{z}) \quad (11)$$

This models the spatial temporal topology of the network in a data driven manner and it is independent of the object type (biker vs pedestrian). The kernel density is learnt from a training data with approximately $K = 100,000$ samples between every pair of views.

VI. Graph Based Query Ranking

Given the network graph of camera observations, an off-the-shell graph ranking method is used to rank items based on the user query. Among the different ranking algorithms, VisualRank [11] is best suited for our scenario and it focuses more on centrality (importance). However, VisualRank does not guarantee diversity, i.e. two similar images have similar ranks. Several ranking algorithms have been proposed to enforce diversity such as absorbing random walks [21], decayed DivRank [6], and manifold ranking with sink points (MSRP) [4]. A manifold ranking with sink points is used

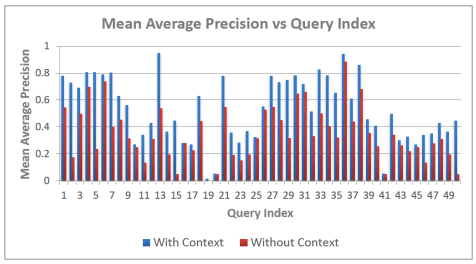

Fig. 5: Mean average precision (mAP) with and without context for 50 different search queries.

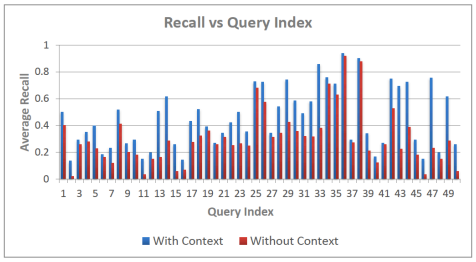

Fig. 7: Recall with and without context for 50 different search queries.

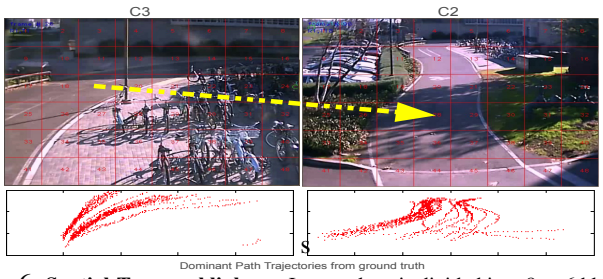

Fig. 6: **Spatial-Temporal linkage:** Image plane is divided into 8×6 blocks. Kernel density estimator models the transition probability of moving from one block to another block in a different view within a specified time delay T_d.

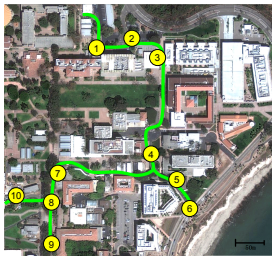

Fig. 8: An aerial map of a wide area camera network with 10 cameras along the bike path.

to rank items based on a user query with an emphasis on diversity [4]. A user interacts with the system by instantiating a query for which the system initializes a query vector $r \in R^{N\times 1}$ whose elements correspond to the nodes (i.e. tracklets), where r is the preference vector.

Algorithm 1: Ranking camera observations using Manifold Ranking with Sink Points

Input: Graph weight matrix W and preference vector $\mathbf{r}$.
Output: Top-ranked vertices $\{m_1, m_2, m_3, \ldots\}$.

1: Initialize the sink point sets $\mathcal{X}_s = \emptyset$
2: Initialize the set of points to be ranked $\mathcal{X}_r$ with the set of nodes in the graph.
3: Symmetrically normalize the weight matrix such that $\mathbf{S} = D^{-1/2}WD^{-1/2}$, W is the adjacency matrix representing the constructed global graph; D is a diagonal matrix whose (i,i)-entry equal to the sum of the i-th row of W.
4: **while** $|\mathcal{X}_s| < S$ **do**
5: Iterate $f^{(t+1)} = \mu \mathbf{S} I_f f^{(t)} + (1-\mu)r$ until convergence where $0 \leq \mu < 1$ and I_f is diagonal indicator matrix with (i,i)-entry equal to zero if the item belongs to sink set $\mathcal{X}_s$.
6: Rank points $x_r \in \mathcal{X}_r$ according to their ranking scores f^*.
7: Pick top ranked points $\{x_t\} \in \mathcal{X}_r$ and turn them into new sink points by moving them from $\mathcal{X}_r$ to $\mathcal{X}_s$.
8: **end while**
9: Return sink points in the order in which they were selected into $\mathcal{X}_s$ from $\mathcal{X}_r$.

Consider the query 'find objects related to region B of camera c between time t_1 and t_2" (Spatial-Temporal Browsing) (see Figure 1(a)). First, the system will find those tracklets that are closely related to the supplied query. Let N_q be the number of matching query nodes and $\{\mathcal{M}_q\}$ be the set of matching nodes. The system assigns a uniform matching scores for the nodes in the query set, i.e., $r_i = 1/N_q$ if $i \in \{\mathcal{M}_q\}$, and $r_i = 0$ otherwise. For an object search query (see Figure 1(b)), user supplies a query similar to find "object instance appears at time t of camera 3", which corresponds to j^{th} node in the graph. The preference vector is set as $r_j = 1$ with all other entries set to 0. In the following, we discuss the manner in which manifold ranking with sink points is used to rank items in the network graph using the preference vector r. As this is an iterative algorithm, it can be applied to extremely large graphs in a scalable manner.

A. *Manifold Ranking with Sink Points*

Let $\mathcal{X}_r$ be the set of nodes (points) to be ranked and $\mathcal{X}_s$ be the set of sink points. Let $f \in R^{N\times 1}$ be the vector containing the ranking scores for the graph nodes in $\mathcal{X}_r$. We use the strategy proposed in [4] to rank the items in $\mathcal{X}_r$. Algorithm 1 summarizes the proposed strategy for ranking items with an emphasis on diversity. Intuitively, at every iteration, the algorithm tries to spread the scores from the query nodes, i.e. $r_i \neq 0$ to the rest of the nodes in the graph by simulating a gradient walk. Ranking scores f are assigned based on the likelihood of visiting other nodes. The parameter μ controls the step size of the gradient walk (in our experiments, we set $\mu = 0.01$). For diversity, a set of sink points are introduced (S is the number of sink points), whose ranking scores are fixed at a minimum value (say zero) during the ranking. Hence the sink points will stop the spreading of scores to their neighbors. The final rank that is obtained after the ranking process is: i) Relevant to the user query, ii) Importance (i.e., centrality) and iii) Diverse compared to all the other nodes in the graph.

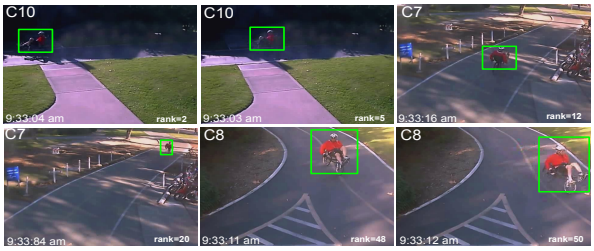

Fig. 9: Top ranked items are shown according to the rank order. An object of interest is highlighted by a green bounding box. For the ease of demonstration, some of the results are not shown. The object traverses from Camera C10 to C8 and then to C7. The corresponding global timestamp is also shown.

TABLE II: Mean Average Precision/Recall/F-measure for 50 different queries

	Mean Average Precision	Mean Recall	F-measure
With context	**0.52**	**0.45**	**0.48**
Without context	0.34	0.30	0.32

VII. Experiments

We evaluated the proposed methodology in a wide area camera network consisting of ten cameras along a bike path (See Figure 8). Videos (640x480, about 20 frames per second with variable frame rate) are captured for several hours in an uncontrolled environment with complex shape and appearance changes in objects, wireless packet losses and irregular illumination variations. This is a challenging dataset for which most of the off-the-shelf object detection and tracking algorithms fail due to various reasons [20]. For all the application scenarios presented in this section, we set the confidence parameters $\alpha = 0.20$, $\beta = 0.40$, $\gamma = 0.40$ and $\eta = 1$. Since the proposed problem is relatively new, there are not many algorithms to compare with. Therefore we demonstrate the robustness of the proposed methodology using some application scenarios and we use Mean Average Precision (mAP) and Recall for comparison. Figure 1(a) shows a sample Browsing query. A query consisting of a spatial region is marked (a red bounding box) along with a time interval (9:32am to 9:34am) is given as an initial browsing query to the system. The system returns initial candidate frames that match the browsing query and an object of interest is selected as a search query (shown in Figure 1(b)). Figure 9 shows the visual retrieval results for the sample search query (shown in Figure 1(b)).

A. Effect of Contextual Information

In order to demonstrate the role of contextual information, we compared the mean precision and recall obtained for 50 different queries with and without context. For experiments without context, we set confidence parameters $\alpha = 0$, $\beta = 0$, $\gamma = 0$ and $\eta = 1$. Figure 5 shows the mean average precision obtained with and without context for various levels of retrieval depth ($depth = [5, 10, 15, 20, 25, 30, 35, 40, 45, 50]$). Figure 7 shows recalls obtained with and without context for a retrieval depth of 50. Use of contextual information significantly improves the overall accuracy of the retrieval results. Table II shows the mean average precision and mean recall

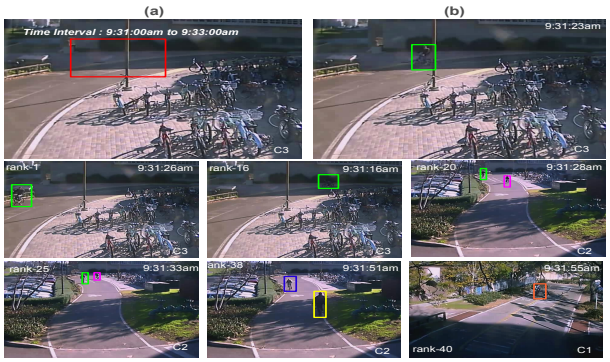

Fig. 10: (a) A spatial region along with the time interval is specified as an initial query to the system. (b) An object of interest is marked with a green bounding box. Last two rows displays top ranked items. Objects that co-occurred (in space and time) are highlighted by pink, blue, yellow and orange bounding boxes respectively.

TABLE III: Mean Average Precision with different parameters.

Parameter	1	2	3	4	5
Mean Average Precision	0.47	0.41	0.38	0.343	0.33

for 50 different queries with and without various contextual information. We obtain 50% improvement in results (in terms of F-measure) by using contextual information.

B. Effect of Parameters

To highlight the role of different contextual information, we performed a set of experiments with 10 different search queries. We used 5 different parameter settings: *Parameter-1*= $[\alpha = 0.33, \beta = 0.33, \gamma = 0.33, \eta = 1]$ (all contextual information), *Parameter-2*= $[\alpha = 0, \beta = 0.5, \gamma = 0.5, \eta = 1]$ (no appearance context), *Parameter-3*= $[\alpha = 0.5, \beta = 0, \gamma = 0.5, \eta = 1]$ (no spatial-temporal context), *Parameter-4*= $[\alpha = 0.5, \beta = 0.5, \gamma = 0, \eta = 1]$ (no scene context) and *Parameter-5*= $[\alpha = 0.33, \beta = 0.33, \gamma = 0.33, \eta = 0]$ (no spatial topology modeling). Table III shows mean average precision obtained with different parameters. Spatial temporal topology modeling plays a significant role in improving the accuracy. The object transition model based on historical data provides a strong cue in order for object association. The spatial-temporal context is more meaningful for the cameras with overlapping field of views and approximately similar direction of sensing. Also, scene and spatial-temporal contexts contribute significantly to object retrieval process. Of all the contextual information, appearance context does not contribute significantly in this dataset since most of the objects appeared dark with very little color information for discrimination. Also, we do not take color drifts into account. Figure 11 shows mean precision at various levels of retrieval depth.

C. Emphasis on Diversity

In this set of experiments, we demonstrate the role of diversity in our graph ranking algorithm. The last two rows of Figure 10 show the retrieval results for a browsing query (Figure 10(a)) and its corresponding search query (Figure 10(b)). Objects that are closely related in space and time to

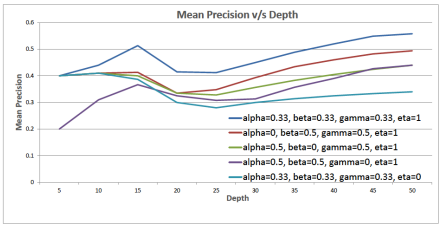

Fig. 11: Mean precision for 10 different queries with various parameters.

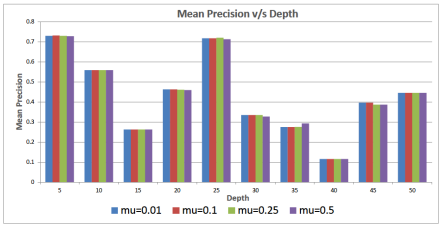

Fig. 12: Mean precision for different values of step parameter i.e. μ=$[0.01, 0.1, 0.25, 0.5]$

the search query are retrieved by emphasizing diversity in the graph ranking. For example, in Figure 10, objects highlighted by pink, blue, yellow and orange bounding boxes co-occurred in time and also traveled in close spatial proximity with the object of interest. Figure 12 shows the effect of parameter μ on mean precision at various retrieval depths. μ signifies relative contribution of neighbors and initial ranking scores to the final ranking scores. By emphasizing diversity, change in parameter μ had little effect on the overall precision.

D. Discussion on Graph Size

At the central node, we build a time-evolving graph using the tracklets received from different camera views and it has a tendency to inflate over the time. In our experiments, the network graph consisted of 7125 nodes with 44196344 edges (approximately 87% connectivity compared to a complete graph). For memory optimization, we can ignore nodes that were added before a certain period of time (say 2 hours). Also, we can neglect the connectivity between the nodes from distant camera views.

VIII. Conclusion

This paper proposes a novel context-aware graph based system to assist human analysts to efficiently browse and search objects in a large wide area camera network. We provide a novel strategy to encode contextual information such as appearance, scene and spatial-temporal contexts to increase the overall accuracy of retrieval. The proposed methodology is validated with extensive experiments on some real-world large scale camera network dataset. With contextual information, the accuracy of the proposed system is increased by approximately 50% (in terms of F-measure).

Acknowledgments

This work was supported by ONR grant #N00014-12-1-0503.

References

[1] S. Ali, V. Reilly, and M. Shah. Motion and appearance contexts for tracking and re-acquiring targets in aerial videos. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–6. IEEE, 2007. 2, 3

[2] C. Arth, C. Leistner, and H. Bischof. Object reacquisition and tracking in large-scale smart camera networks. In *Distributed Smart Cameras, 2007. ICDSC '07. First ACM/IEEE International Conference on*, pages 156–163, 2007. 2

[3] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal. Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1932–1939. IEEE, 2009. 4

[4] X.-Q. Cheng, P. Du, J. Guo, X. Zhu, and Y. Chen. Ranking on data manifold with sink points. *Knowledge and Data Engineering, IEEE Transactions on*, 25(1):177–191, 2013. 4, 5

[5] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):603–619, 2002. 2

[6] P. Du, J. Guo, and X.-Q. Cheng. Decayed divrank : Capturing relevance, diversity and prestige in information networks. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1239–1240, 2011. 4

[7] A. G. Gray and A. W. Moore. Very fast multivariate kernel density estimation via computational geometry. In *Joint Stat. Meeting*, 2003. 4

[8] O. Javed, Z. Rasheed, O. Alatas, and M. Shah. Knightˆm: A real time surveillance system for multiple overlapping and non-overlapping cameras. In *IEEE Conference on Multi media and Expo*, pages 6–9, 2003. 2

[9] O. Javed, Z. Rasheed, K. Shafique, and M. Shah. Tracking across multiple cameras with disjoint views. In *International Conference on Computer Vision*, pages 952–960. IEEE Computer Society, 2003. 1, 2

[10] O. Javed, K. Shafique, and M. Shah. Appearance modeling for tracking in multiple non-overlapping cameras. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 26–33, 2005. 1, 2

[11] Y. Jing and S. Baluja. Visualrank: Applying pagerank to large-scale image search. *IEEE Transaction Pattern Analysis and Machine Intelligence*, 30(11):1877–1890, 2008. 4

[12] V. Kettnaker and R. Zabih. Bayesian multi-camera surveillance. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 252–259, 1999. 1

[13] L. Li, W. Huang, I. Y. H. Gu, and Q. Tian. An improved adaptive background mixture model for real-time tracking with shadow detection. In *ACM International Conference on Multimedia*, pages 2–10, 2003. 2

[14] Z. Ni, S. Sunderrajan, A. Rahimi, and B. Manjunath. Particle filter tracking with online multiple instance learning. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 2616–2619. IEEE, 2010. 1

[15] Z. Ni, J. Xu, and B. Manjunath. Object browsing and searching in a camera network using graph models. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 7–14. IEEE, 2012. 1, 2

[16] A. Oliva, A. Torralba, et al. The role of context in object recognition. *Trends in cognitive sciences*, 11(12):520–527, 2007. 2

[17] I. Saleemi, K. Shafique, and M. Shah. Probabilistic modeling of scene dynamics for applications in visual surveillance. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(8):1472–1485, 2009. 4

[18] S. Sunderrajan, S. Karthikeyan, and B. Manjunath. Robust multiple object tracking by detection with interacting markov chain monte carlo. In *Image Processing (ICIP), 2013 20th IEEE International Conference on*. IEEE, sept. 2013. 1

[19] H. Tong, J. He, M. Li, C. Zhang, and W.-Y. Ma. Graph based multi-modality learning. In *ACM International Conference on Multimedia*, 2005. 2

[20] J. Xu, V. Jagadeesh, Z. Ni, S. Sunderrajan, and B. Manjunath. Graph-based topic-focused retrieval in a distributed camera network. *IEEE Transaction on Multimedia*, Dec 2013. 2, 6

[21] X. Zhu, A. B. Goldberg, J. Van Gael, and V. G. Andrzejewski. Improving diversity in ranking using absorbing random walks. In *The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 97–104, 2007. 4

[22] Y. Zhu, N. Nayak, and A. Roy-Chowdhury. Context-aware activity recognition and anomaly detection in video. *Selected Topics in Signal Processing, IEEE Journal of*, 7(1):91–101, 2013. 2