Efficient Object-Based Distributed Image Search in Wireless Visual Sensor Networks

*1Seungmin Rho
1Department of Media Software, Sungkyul University, Korea
smrho@sungkyul.ac.kr

Abstract

Modern wireless visual sensor networks are a large collection of inexpensive computational sensor nodes. These nodes continuously capture images from the environment and, then, process, store, and transmit the images to one or more servers. Visual data gathered by these networks grow exponentially and present many challenges concerning their transmission, storage, and retrieval. Recent advances in sensor node architectures resulted in enhanced processing capabilities and extended local storage has created new opportunities for innovative applications of these network platforms. Visual sensor networks have emerged as inexpensive and powerful but resource constrained distributed systems. In this paper, we present a framework for distributed image search, utilizing the processing, storage, and transmission capabilities of individual sensor nodes in an efficient manner. Every sensor acts as a self-contained smart node that can prioritize captured images, extract and store discriminative and compact features from those images, and perform searching. The proposed system is evaluated for two large and challenging datasets. Results report significant improvements in both performance and efficiency over other state-of-the-art methods.

***Keywords**: Distributed image search, visual sensor network, feature vector, embedded processing, wireless sensor network*

1. Introduction

In recent years, visual sensor networks have come out to be the most common choice for security reason and surveillance. They are deployed by covering the area of interest with visual sensor nodes. The sensor node includes an image sensor, embedded processor, and wireless transceiver [1]. With the inclusion of embedded processors, modern sensor nodes are able to process data locally in a smart way in order to determine the importance of sensed data before transmission. Individual node has the ability to sense the environment, process locally and streaming video to a central visual processing hub (VPH) and base station (BS) for the operators to process manually.

Because of the recent development in low power processing, visual sensors, wireless networking and distributed sensing, we witness an increment in the amount of sensors-based applications ranging from security to smart homes. In a network of smart visual sensor nodes, the camera nodes form a large distributed system, where each sensor node is capable of processing and storing image data locally. They are able to communicate with other sensors, retrieve related data rather than showing the operators tons of unrelated data, they support users in selecting a subset of the sensed data which helps the VPH to analyze, store, and transmit more easily.

In practice, VPH is unable to receive all data streamed from the sensors due to resource constraints. Moreover, enormous surveillance videos will bring huge challenges to the analysts as they are overload. In addition, it will cost unnecessary resources to process and stream such a large video in the network. In order to overcome this problem, many researches on summarization and prioritization have been done. These methods processed images locally to decide which frame to transmit, so they can save precious resources in networks that have limited resources. In these systems, the sensors only send

* Corresponding Author
Received: Nov. 13, 2017, Revised: Nov. 29, 2017, Accepted: Dec. 21, 2017

appropriate frames to the VPH and BS where they are being processed [1]. As time pass, the data size increases exponentially which makes it difficult to analyze relevant information. The requirement for effective and more reliable ways to extract the right visual contents from such huge data is imminent.

With the inclusion of improved embedded processors and extendable flash memory, visual sensor networks have become an efficient platform to perform distributed query for visual contents. Although, distributed search in VSNs has opened new opportunities, it also carries with it several challenges. For instance, resource constraints of visual sensors restrict the use of complicated search procedures. Optimized and energy-friendly computations and storage schemes are needed for local data processing. Finally, an effective fusion scheme is required to merge results from multiple sensor nodes. This paper proposes a distributed image retrieval system, that leverages the local processing and storage capabilities of the visual sensor nodes to efficiently search relevant images in a VSN. Each node is treated as a local search engine for the data it has sensed. The proposed system is designed efficiently utilizing inexpensive-computations, low-dimensional feature vectors from segmented objects, and local indexing on the sensor node. The proposed system is simulated for a network of iMote2 sensor nodes with extended flash storage. The main contributions of our work are as follows.

1. Reduced feature computation time by eliminating background from captured images
2. Reduction in data transmissions by indexing images locally using compact feature vectors and transmitting background eliminated salient object images only
3. Distributed image search system efficiently utilizing the local processing capability of visual sensor nodes
4. Reduced packet size during query results propagation in the VSN
5. Efficient strategy for merging similarity scores returned by multiple sensor nodes

The rest of the paper is organized as: In section 2, some of the previous notable works in this field are listed, Section 3 presents the architecture of the proposed method, experiments and discussions are carried out in Section 4. Finally, section 5 gives the conclusions on the paper and proposes future directions.

2. Related Works

Distributed visual sensor networks are resource constrained and researchers dealing with VSNs are often faced with limited resources like energy, transmission capability, and storage. On the other hand, visual data processing algorithms are usually computational and data intensive. This gives rise to the challenge of achieving maximum performance on the available hardware with limited capabilities. Researchers are actively working on both fronts; i.e. building energy efficient distributed processing algorithms and reducing data transmission requirements in VSNs. The remaining of this section summarizes some of the notable works in these areas.

The authors in [2-4] attempt to transmit visual data as efficiently as possible by minimizing data size through image quality degradation. However, degrading image quality often leads to less efficient performance of the systems. Sulic et al. [5] transmit features of the visual information instead of the actual visual data. These features were used for efficient routing. The contents of the data packets were analyzed for similarities with already transmitted data. Transmission efficiency was obtained in pattern recognition problems especially in distributed environments.

Computer vision algorithms have been designed to take advantage of the distributed nature of VSNs for camera calibration [6, 7], multi-view gesture recognition [8], multi-camera surveillance system [9], object recognition [10], and tracking [11]. With all these promising applications in the domain of VSNs, it is feasible to attempt implementation of other systems on this distributed platform.

Image search is a problem that arises in every VSN when it has collected a month-long data. The heavy volume of data generated by the sensor nodes and collected at the BS, make indexing and searching, a challenging task. Yan et al. [12] performed distributed image search in a wireless visual sensor network using SIFT based features. Hierarchical k-means clustering on SIFT features were conducted to create visual words or visual terms. Each descriptor was presented by a 4 bytes' visual term. Though, they decreased the image indexing feature dimension and feature matching computation, the cost for feature extraction was still remained high. Because the method used a big vocabulary tree which need large local storage [1].

The authors in [13] attempt to reduce the transmitted messages across the network during image retrieval from a visual sensor network. Edge orientation histogram feature vectors were used to index the images. To reduce traffic across the network, feature vectors are transmitted to each horizontal node which in turn run queries across the vertical nodes in a mesh structured network. Number of messages transmitted is reduced, but the time required for a query to process is increased.

In [5], the authors propose a hierarchical feature distribution method for object based distributed image retrieval and achieve reduced transmission requirements in their proposed framework. The framework was tested on standard datasets and significant reduction in network traffic was achieved with no degradation in matching performance. Reduced feature vectors were used which require less transmission energy and less storage space. The authors in [22-25] also addressed the issues in wireless sensor networks.

It is imperative to conclude that only a few noticeable methods have been proposed for image search in VSNs. The problem of image search in VSNs is still open and significant efforts need to be taken to build efficient distributed image search systems for such networks. In this paper, we provide an efficient, cost effective and reliable image retrieval system for distributed wireless visual sensor networks.

3. Materials and Methods

Efficient utilization of the available hardware resources in a distributed VSN is the key factor towards successful deployment of an image retrieval system. Efficiency and transmission modules for the system need to be considered carefully. We have attempted to come up with less intensive computations and proposed a method to reduce the amount of data transmissions during network operations and query processing on the VSN. Our objectives are to reduce transmission data and use efficient indexing and retrieval scheme for high performance image retrieval.

3.1. Image Prioritization

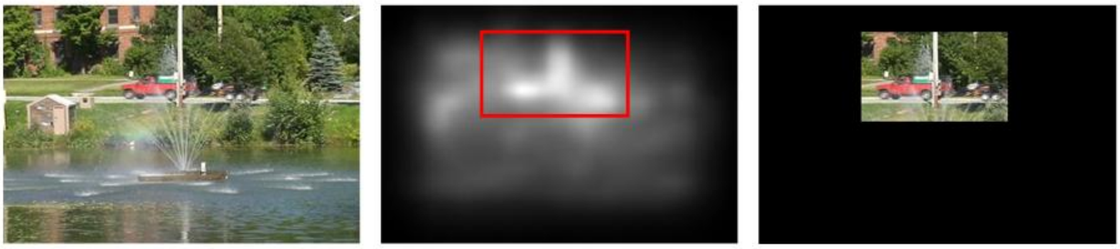
Not all the visual data seen by a visual sensor node is vital. As the result, the sensed data must be processed by efficient algorithms to filter unimportant data before transmitting the server. Transmission consumes the most energy, and therefore, minimizing data transmissions is a viable option which will eventually prolong network life time [14]. In order to achieve this, we adopted an image prioritization scheme presented in [15], which employs motion saliency to determine priorities before transmitting any visual data to the server. A lightweight and energy friendly prioritization algorithm adds to the efficiency of the overall prioritization framework which eventually results in reduced communication overhead. Low-priority and high-priority sample images are shown in Figure 1. The low priority image contains only the background with no noticeable moving actions. On the other hand, the high one includes a moving car, which is categorized as noticeable moving object by our prioritization method.

(a) (b)

Figure 1. (a) Low priority image (dynamic background image with no salient motion detected), (b) High priority image with salient object

3.2. Object Selection

The data size reduced significantly as the result of data prioritization when it is transmitted in the network. In order to reduce more data size, we propose a simple but important method. As the system only use a portion of the prioritized image (e.g. car or a person). The remaining parts are not important and can be removed. Figure 2 shows a sample high prioritized image, its saliency map, and the corresponding segmented image without background.



(a) (b) (c)

Figure 2. (a) Priority image, (b) Motion saliency map generated by [15], (c) important object with no background

The segmentation step has two benefits. It reduces the amount of data to be extracted. Moreover, the transmitted data is reduced. Finally, background removal algorithm requires few computations power. For the background removal algorithm implementation, the readers are referred to [1, 15].

3.3. Feature Extraction

For efficiency and reliability, it is crucial to capture the necessary features of visual contents for object matching in image retrieval applications. For the purpose of feature extraction in our case, we adopted a compact descriptor known as color and edge directivity descriptor (CEDD) with some modifications, which utilizes fuzzy color and texture features to represent salient visual content as histograms [16].

[17] proposed 10 color fuzzy binning scheme, which formed a 10 bin histogram for colors black, gray, white, red, orange, yellow, green, cyan, blue, and magenta. Each pixel value from the captured color image increments one or more bins of this histogram by some fraction, depending upon the adopted mapping strategy. Usually the HSV color model is used and membership values are computed from these three color channels. The dividing boundaries between the various color channels is found using coordinate logic filters (CLF) [18]. The input pixels contribute to fuzzy bins derived from various color channels as: Hue is segmented into 8 fuzzy sections representing red to orange, orange, yellow, green, cyan, blue, magenta, and blue to red segments. Similarly, saturation is divided into 2 segments which provides information regarding shade of a color based on white. The V channel segments into 3 areas. The first edge represent boundary between pure black and the rest of the shades. The remaining two separates light and dark gray areas. A set of 20 fuzzy rules map input pixels to histogram bins and increment selected bins of this fuzzy-linking histogram.

The method suggested by MPEG-7 Edge Histogram Descriptor is applied with few modifications to extract texture features from the ROI. Four local edge features corresponding to four different directions (0^o, 45^o, 90^o, and 135^o) are extracted from individual blocks of the ROI. Each of these blocks is further divided into 4 sub blocks. The edge magnitudes of four directions from each sub-block form a 5 area histogram carrying texture information. The final feature histogram is constituted by 5 texture regions where each region is further represented by a 10 bin color histogram. The final histogram includes 5 × 10 = 50 regions represented by 50 bins histogram. Each bin is quantized into 6 bits resulting a final descriptor of size 25 bytes. The fuzzy color histogram along with the texture features form a powerful discriminative representation of the image requiring only 25 bytes per color image.

3.4. Feature Indexing

All the features extracted from the segmented salient objects are stored in the local storage of the visual sensor node. The small feature database is a repository containing information about all individual salient objects seen by this node. For an image with multiple objects, multiple feature vectors are stored on the local storage. This object based feature pooling is done in order to achieve distributed object-based image retrieval in VSNs. Any object which is smaller than some threshold is ignored. An optimal threshold value is selected after experimentations.

imageID is a unique identity for each feature vector, including the date, time when the image is taken, sensor node ID, and an integer number. The server recognizes every individual object using its unique imageID across all the network.

3.5. Feature Matching

In order to perform efficient feature matching, the proposed scheme introduces a quick-reject strategy along with a lightweight feature matching formula. During the distance computation using Equation 1, if the distance D becomes greater than some threshold, the feature matching process is immediately terminated and the corresponding object is ignored. The threshold value is chosen after analyzing the feature differences among similar and dissimilar objects using the proposed feature extraction scheme.

$$D(F_q, F_i) = \sum_{D \le \tau} \left| F_q - F_i \right| \tag{1}$$

where D is the two feature vectors dissimilarity score, F_q is the query vector received using the medium communication during a query process, F_i is a feature vector stored locally. During the entire feature matching process, if the value of D becomes greater than threshold τ, the process for the particular F_i is terminated and the rest of the comparisons are skipped. This helps reduce the number of comparisons and saves precious processing time and energy. The threshold value τ is determined by taking the mean of dissimilarity scores between similar and dissimilar images using equation 2 as:

$$\tau = \frac{1}{2}\left(D_{relevant}(F_q, F_i) + D_{irrelevant}(F_q, F_i) \right) \tag{2}$$

3.6. Distributed Image Search

Initiating a query by distributing a full image across the VSN is a very expensive operation. Therefore, researchers are looking for ways to reduce burden on the network during a query. Some of the research suggest to reduce the image size before transmitting the query image [19, 20], others adapt more intelligent ways by distributing feature vectors instead of images [5], and selecting the appropriate subset of features for object re-identification [21]. We adapt the feature distribution strategy and transmit the feature vector of the query image instead of the full or resized image. The small size of the feature vector (25 bytes) make it convenient to transmit across the network without overloading communication lines of the network.

The schematics of the proposed distributed image search are illustrated in Figure 3. A query is initiated from the base station after extracting features from the segmented object image that is to be searched across the network. Feature vector is transmitted to all nodes in the VSN. Each node activates a local image searching module on the local feature-set. Image IDs of the relevant images found are sent back to the server for further processing. If no relevant images were found on a sensor, it does not send any data to the server, thus reducing the overall network traffic.

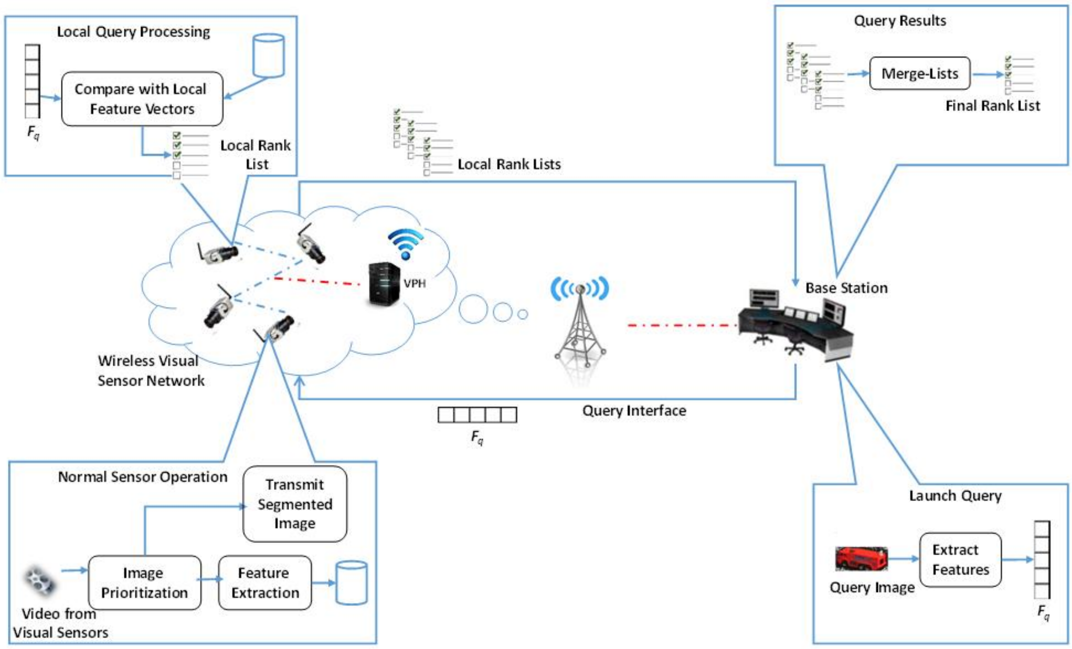


Figure 3. Schematics of the proposed framework

3.7. Local Ranking of Relevant Images

During the local searching on each sensor node, feature vector of the query image, F_q is compared with every feature vector F_i on the local storage to obtain a list L_L of potentially relevant images. This list is sorted on the basis of dissimilarity scores in ascending order to get the local rank list of images. The list containing ImageIDs and SensorIDs along with their dissimilarity scores is sent to the VPH for onward delivery to the BS. Equation 3 symbolizes the strategy for populating the list.

$$L_L = \{(ImageID, SensorID) \mid D(F_q, F_i) < \tau) \quad (3)$$

3.8. Global Ranking of Relevant Images

The locally ranked image lists received by the BS are merged to produce a global rank list L_G. Since the local lists are already sorted, a simple n-way-merge sort algorithm puts together all of the local rank lists into a global rank list.

$$L_G = Merge(L_L^i), \; i = 1, 2, 3 \ldots \mathrm{n} \quad (4)$$

After obtaining the global rank list, images in the list can be displayed to the user as final search results. These results can be made more meaningful and informative if they are displayed over a map, showing the exact location of the sensor in a geographical area along with the date, time, and view of the object as seen by that sensor. Figure 4 provides a schematic diagram of the system. Data acquired from each sensor nodes is shown at their locations using the local map monitored by the network. Additionally, the system enables object tracking over a geographical area.

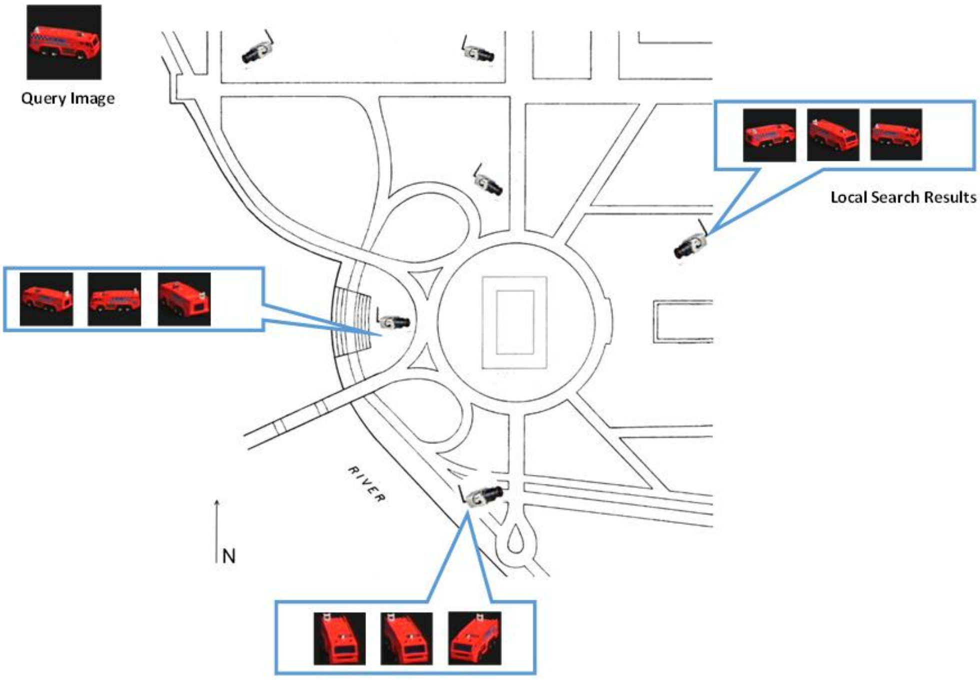


Figure 4. Localized results display interface

4. Results and Discussions

This section gives a step by step evaluation of the proposed scheme in terms of performance and efficiency. The below subsections describe the datasets, experiment design, evaluation metrics, and discussions about the results.

4.1. Datasets

Two datasets were used to evaluate the performance of the proposed method. Both of them are suitable for object based image retrieval, and are widely used to evaluate the performance of such systems. A brief description is given as follows:

a. COIL-100 Dataset:

The Columbia Object Image Library (COIL-100) is a widely used color image dataset of 100 different 3D objects. Each objects has 72 images captured with a pose interval of 5^{o}. There are a total of 7200 images in this dataset having resolution of 128×128. There are no occlusions, background clutter, and illumination variations. This dataset is specifically designed for the evaluation of systems dealing with viewpoint invariant representations of 3D objects. It is suitable in our case because there are different images captured of the same objects by the different visual sensors deployed at different places. Therefore, the proposed system can be evaluated with this dataset. Two experiments were performed using this dataset. The details of the experiments are given in sections 4.5 and 4.6.

b. Surveillance Images Dataset:

This is a more challenging dataset containing keyframes extracted from surveillance videos captured by multiple cameras in both indoor and outdoor environments. Images contain salient objects extracted using the background elimination procedure presented in [15]. A total of 1200 prioritized keyframes extracted from the surveillance videos were used. Image sizes in this dataset were 720 × 576 and 320 × 240.

4.2. Experimental Setup

To simulate the proposed retrieval framework on distributed wireless VSN, a visual sensor simulator was designed having similar storage capabilities as the iMote2 sensor nodes. The simulator was designed using MATLAB 2014a, keeping in view the capabilities and limitations of the sensors. The simulator only deals with the top layer of network communication and ignores network related technicalities including delays, bandwidths etc. A typical iMote2 sensor node is an advanced sensor node platform from Intel Research. It features a low power CPU, an integrated IEEE 802.11.4 compliant radio, along with standard and high-speed I/O interfaces. It also features a 256 KB cache, 32 MB SDRAM, and a 32 MB Flash memory.

The background removal, prioritization, retrieval processes and features extraction were conducted on MATLAB 2014a using a Windows 7 Professional PC (Intel Core i5 processor, 8 GB RAM). To evaluate the capability of the proposed framework in challenging scenarios, three different experiments were created. The corresponding subsections describe the experiments in detailed and for more details readers are referred to [1].

4.3. Evaluation Metrics

Several evaluation metrics are available for determining performance of retrieval systems on quantitative scales. We have adopted the most widely used metrics, including precision (P), recall (R), and F-measure. These can be computed as:

The recall measure for a particular query image I_q on a retrieval system is calculated as:

$$R(I_q, n) = \frac{1}{N_R} \sum_{i=1}^{|D|} \sigma(\chi(I_i), \chi(I_q) \mid \rho(I_i, I_q) \leq \mathrm{n} \mid \tag{5}$$

In which N_R is the number of related images extracted from dataset D. The precision can be calculated as given in equation 6.

$$P(I_q, n) = \frac{1}{n} \sum_{i=1}^{|D|} \sigma(\chi(I_i), \chi(I_q) \mid \rho(I_i, I_q) \leq \mathrm{n} \mid \tag{6}$$

where n represents the number of retrieved images, χ(I) shows the category of image I, ρ represents the rank at which the image I is placed at in the result-set among all images in D, and σ computes the relevancy of an image with the query image considering their categories as:

$$\sigma(\chi(\mathrm{I_i}),\ \chi(\mathrm{I}_q)) = \left\{\begin{matrix} 1, & \chi(\mathrm{I_i}) = \chi(\mathrm{I}_q) \\ 0, & \text{Otherwise} \end{matrix}\right\} \tag{7}$$

F-measure represents a consolidated score by merging precision and recall into a single quantity. It is computed using the weighted mean of precision and recall:

$$F\text{ - }Measure = 2.\frac{P \times R}{P + R} \tag{8}$$

4.4. Experiment 1

In this experiment, random queries were selected from the entire COIL dataset, and top 20 images were retrieved by the proposed method. In order to simulate the distributed scenario of the sensor nodes, images from 20 different views were placed inside separate folders. Each folder simulates the local storage of a single sensor node. Only a subset of images were stored at every separate location. So that each node only has its unique images seen by it. Hence, when a query was executed, results from only those nodes were received which contained the relevant information. The local search results were finally combined at the server for final ranking. This experiment focuses on the retrieval capability of the framework in case of no occlusions. The results of distributed queries executed with 10, 20, 30, and 50 nodes are provided in Table 1. As shown in Table 1, the proposed retrieval method outperforms others both on small and large networks. In a distributed image search system, the number of nodes can be thought of as individual threads running local searches. Increasing the number of threads will increase the search efficiency as well. Therefore, it is important to mention here that the query times were reduced in large networks comparing to smaller networks in our simulations.

4.5. Experiment 2

Images captured by VSNs may contain objects that are partially occluded. Therefore, it is necessary to evaluate the performance of retrieval system in such cases. This experiment is designed to evaluate how the proposed retrieval scheme perform with occluded images from the COIL dataset. Various degrees of occlusions were manually introduced into images, ranging from light (10%) to high (50%) occlusions. The retrieval results with various degrees of occlusions are provided in Figure 5.

Table 1. Retrieval results of proposed scheme for distributed image search in a VSN having 10, 20, 30, and 50 nodes

Nodes	Retrieved Images	Precision	Recall	F-Measure
10	5	1.00	0.25	0.40
	10	0.96	0.48	0.64
	15	0.95	0.72	0.82
	20	0.93	0.93	0.93
20	5	1.00	0.25	0.40
	10	0.96	0.48	0.64
	15	0.95	0.71	0.81
	20	0.92	0.92	0.92
30	5	1.00	0.25	0.40
	10	0.95	0.48	0.63
	15	0.97	0.73	0.83
	20	0.92	0.92	0.92
50	5	1.00	0.25	0.40
	10	0.91	0.46	0.61
	15	0.90	0.68	0.77
	20	0.77	0.77	0.77
	Average	**0.94**	**0.58**	**0.68**

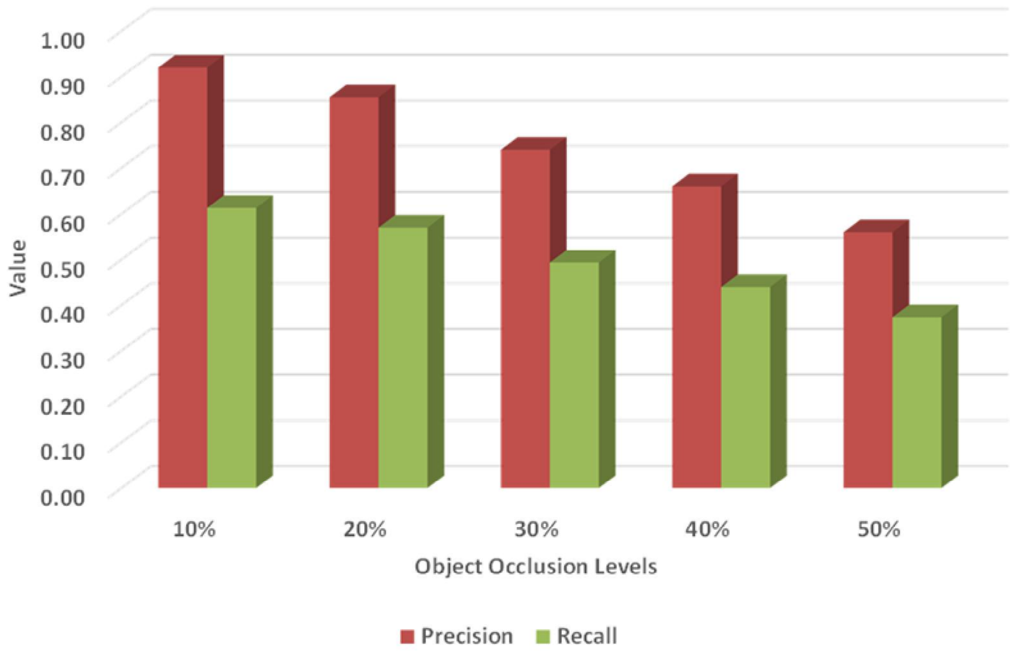


Figure 5. Retrieval results with various degrees of occlusions in occluded COIL-100 dataset

A gradual degradation in performance is noticed in both precision and recall, as given in Figure 5. However, it is important to mention here that even with a high occlusion of 50%, the precision is almost 0.60 for the degraded COIL-100 dataset. For lower degradations (10%-30%), only a slight decrease of 0.08 is noticed in precision. These results show the robustness of the proposed scheme for retrieval of images in a distributed VSN.

4.6. Experiment 3

This test was implemented to check the real surveillance images performance, which is much more challenging than the previous dataset. All objects were automatically segmented, however, some portion of the background could still be found around the salient object. Encouraging retrieval results for this dataset using the proposed framework are given in Table 2. The high precision and recall values for the dataset shows the robustness of proposed descriptor in challenging scenarios.

Table 2. Retrieval results of distributed image search for surveillance images dataset

Query #	Precision	Recall	F-Measure
1	0.90	0.72	0.80
2	0.95	0.61	0.75
3	0.85	0.50	0.63
4	0.90	0.41	0.56
5	0.95	0.68	0.79
6	1.00	0.74	0.85
7	0.95	0.43	0.59
8	0.90	0.72	0.80
9	0.75	0.63	0.68
10	0.95	0.68	0.79

11	0.75	0.47	0.58
12	0.95	0.51	0.67
13	0.80	0.76	0.78
14	0.65	0.65	0.65
15	0.55	0.37	0.44
Average of 50 Images	**0.834**	**0.569**	**0.671**

4.7. Framework Efficiency Analysis

It is essential to save processing time and transmission data sizes for efficient utilization of the available resources in a VSN. The proposed scheme tends to reduce data sizes wherever possible to conserve energy. The proposed background elimination scheme helps reduce the data size by almost 65% which decreases the energy required to upload it to the VPH. Details of reduction in data sizes is given in Table 3.

Table 3. Reduction in image size as a result of background removal

Image Resolution	Full Image Size (KB)	Reduced Image Size (KB)	Percent Reduction in Data Size
2048 x 1536	1340	175	86.940
1920 x 1080	648	110	83.025
640 x 480	333	126	62.162
320 x 240	110	40	63.636
180 x 120	7	5	28.571

As a result of this significant decrease in the amount of data, the processing times also gets reduced. The time required for feature computation has reduced by a significant factor of 16% as shown in Table 4.

Table 4. Reduction in feature computation time as a result of reduced image sizes

Image Resolution	Full Image Size (s)	Reduced Image Size (s)	Percent Reduction in Data Size
2048 x 1536	1.400	1.200	14.286
1920 x 1080	0.700	0.530	24.286
640 x 480	0.200	0.170	15.000
320 x 240	0.160	0.130	18.750
180 x 120	0.100	0.090	10.000
		Average	**16.464**

4.8. Comparison with state-of-the-art methods

The proposed scheme is compared with other state-of-the-art methods to prove its efficiency. The most relevant work in distributed image search is carried out by Yan et al. [12], who employed SIFT based visual vocabulary scheme to search images. Their method used visual words “visterms”, whose computation require more time and energy than our method. The detailed comparison is provided in Table 5. The improvement in query processing times and performance is attributed to the selection of powerful discriminative features, reduced data sizes for transmission during query processing, and efficient resource management.

Table 5. Energy consumption and retrieval performance comparison of proposed framework with comparative method on COIL-100 dataset

Method	Feature Computation(J)	Query Processing (J)	Total Energy (J)	Precision (Top 5 Images)
Visterm [12]	0.52	0.41	0.93	0.92
Proposed	0.27	0.40	0.67	0.97

From Table 5, it can be observed that the proposed distributed image search improves energy consumption by almost 28% compared to the other method. In addition to this, our framework improves retrieval performance by 5% for top-5 retrieved images. From these results, it is evident that the proposed image search process brings significant improvements in terms of both performance and efficiency.

5. Conclusion and Feature Directions

Efficient resource utilization in a VSN is the key to their successful use. In this paper, we presented a framework for performing image search in a distributed visual sensor network by efficiently utilizing the limited network resources. During each phase of this framework, we attempted to minimize data transmissions, reduced computation times, and storage space requirements for query processing. The use of lightweight and intelligent prioritization scheme allowed a huge reduction in the captured visual data. The removal of background from high priority images, prior to transmission and feature extraction helped in reduction of both transmission energy consumption and feature computation times. The extraction of discriminative and compact features from the segmented images helped us achieve high retrieval performance with large collections of images.

Distributing features of the query image from the base station and the transmission of potentially relevant image lists from the sensor nodes to the base station resulted in low energy consumption during distributed query processing. The efficient utilization of resources and the high performance of the proposed framework make it a better choice for performing distributed image search on a VSN compared to other such methods.

6. References

[1] J. Ahmad, I. Mehmood, and S. W. Baik, "Efficient object-based surveillance image search using spatial pooling of convolutional features," Journal of Visual Communication and Image Representation, vol. 45, pp. 62-76, 2017.

[2] W. Yu, Z. Sahinoglu, and A. Vetro, "Energy efficient JPEG 2000 image transmission over wireless sensor networks," in Global Telecommunications Conference, 2004. GLOBECOM'04. IEEE, 2004, pp. 2738-2743.

[3] D.-U. Lee, H. Kim, M. Rahimi, D. Estrin, and J. D. Villasenor, "Energy-efficient image compression for resource-constrained platforms," Image Processing, IEEE Transactions on, vol. 18, pp. 2100-2113, 2009.

[4] V. Lecuire, C. Duran-Faundez, and N. Krommenacker, "Energy-efficient image transmission in sensor networks," International Journal of Sensor Networks, vol. 4, pp. 37-47, 2008.

[5] V. Sulić, J. Perš, M. Kristan, and S. Kovačič, "Efficient feature distribution for object matching in visual-sensor networks," Circuits and Systems for Video Technology, IEEE Transactions on, vol. 21, pp. 903-916, 2011.

[6] E. Simmons, E. Ljung, and R. Kleihorst, "Distributed vision with multiple uncalibrated smart cameras," in International Workshop on Distributed Smart Cameras (DSC 06), 2006.

[7] D. Devarajan and R. J. Radke, "Calibrating distributed camera networks using belief propagation," EURASIP Journal on Applied Signal Processing, vol. 2007, pp. 221-221, 2007.

[8] T. Kirishima, Y. Manabe, K. Sato, and K. Chihara, "Real-time multiview recognition of human gestures by distributed image processing," Journal on Image and Video Processing, vol. 2010, p. 1, 2010.

[9] H. Aghajan and A. Cavallaro, Multi-camera networks: principles and applications: Academic press, 2009.
[10] A. Y. Yang, S. Maji, C. M. Christoudias, T. Darrell, J. Malik, and S. S. Sastry, "Multiple-view object recognition in band-limited distributed camera networks," in Distributed Smart Cameras, 2009. ICDSC 2009. Third ACM/IEEE International Conference on, 2009, pp. 1-8.
[11] M. J. Mirza and N. Anjum, "Association of moving objects across visual sensor networks," Journal of Multimedia, vol. 7, pp. 2-8, 2012.
[12] Si-Yong Park, Sun-Myung Hwang, "A Congestion Avoidance Policy to extend lifetime of Sensor Networks ", Journal of Security Engineering, Vol.12, No.2 (2015), pp.169-180, http://dx.doi.org/10.14257/jse.2015.04.06
[13] S. Milovanovic and M. Stojmenovic, "Quorum based image retrieval in large scale visual sensor networks," in Ad-hoc, Mobile, and Wireless Networks, ed: Springer, 2012, pp. 449-458.
[14] Sangho Lee, Haengrae Cho, Aekyung Moon, "Data Suppression Scheme for Corona based Wireless Sensor Networks", Journal of Security Engineering, Vol.11, No.4 (2014), pp.339-354, http://dx.doi.org/10.14257/jse.2014.08.06
[15] I. Mehmood, M. Sajjad, W. Ejaz, and S. W. Baik, "Saliency-directed prioritization of visual data in wireless surveillance networks," Information Fusion, vol. 24, pp. 16-30, 2015.
[16] S. A. Chatzichristofis and Y. S. Boutalis, "CEDD: color and edge directivity descriptor: a compact descriptor for image indexing and retrieval," in Computer vision systems, ed: Springer, 2008, pp. 312-322.
[17] K. Konstantinidis, A. Gasteratos, and I. Andreadis, "Image retrieval based on fuzzy color histogram processing," Optics Communications, vol. 248, pp. 375-386, 2005.
[18] H.-J. Zimmermann, Fuzzy sets, decision making, and expert systems vol. 10: Springer Science & Business Media, 2012.
[19] M. Rahimi, R. Baer, O. I. Iroezi, J. C. Garcia, J. Warrior, D. Estrin, et al., "Cyclops: in situ image sensing and interpretation in wireless sensor networks," in Proceedings of the 3rd international conference on Embedded networked sensor systems, 2005, pp. 192-204.
[20] K.-Y. Chow, K.-S. Lui, and E. Y. Lam, "Efficient on-demand image transmission in visual sensor networks," EURASIP Journal on Applied Signal Processing, vol. 2007, pp. 225-225, 2007.
[21] S. F. Tahir and A. Cavallaro, "Cost-effective features for reidentification in camera networks," Circuits and Systems for Video Technology, IEEE Transactions on, vol. 24, pp. 1362-1374, 2014.