# Quorum based image retrieval in large scale visual sensor networks

Stojan Milovanovic, Milos Stojmenovic

Singidunum University, Belgrade, Serbia
{smilovanovic, mstojmenovic}@singidunum.ac.rs

**Abstract.** A recent publication by [SPKK] introduces a framework and set of rules by which object recognition can work on a visual sensor network. Extracted features of the detected object are flooded (with reduced dimensionality at each hop) in the network. The Sensor will match the corresponding feature of the new object with a locally stored one, and send the query on the backward link toward the original detector for matching. Based on their framework we introduce an algorithm which attempts to minimize the number of messages passed within the network when performing an image retrieval task. Extracted features are distributed along a row, while query matching progresses along a column. We compare our results to the algorithm proposed by [SPKK] and achieve fewer transmissions in the retrieval step, and avoid flooding in the pre-processing phase. We expand our algorithm by constructing an information mesh of multiple detections of the same object, to achieve matching with the nearest copy. We also propose a novel feature reduction method, by diving the image into k2 subimages, and extracting features in each subimage. This allows replacing histogram based features with a wide range of other options.

**Keywords:** visual sensor networks, computer vision, object recognition

## 1 Introduction

Recently, visual sensor networks have received attention since they attempt to combine the seemingly non congruent research areas of image processing and ad hoc sensor networks. A philosophical gap exists between the two since they arise from different requirements which need to meet in order to form a visual sensor network. Image processing usually has real time processing requirements, which are more important than memory, storage or power consumption, whereas wireless sensor networks focus on the minimization of power consumption at the expense of a heavy computational load.

The object detection and recognition area research of Computer Vision (CV) field extracts useful information and makes sense of raster imagery. Its goal is to identify objects in images regardless of color, orientation, scale, rotation, position or lighting condition. This is a difficult problem which has only proven successful with certain classes of objects. Usually such systems are very processor, data, and memory inten-

sive which make them good candidates for parallel, powerful processing systems. However, distributed video surveillance applications are situations where each node in the grid is a visual sensor (such as a simple camera) and has limited computational capacity, but can also communicate with other nodes in the grid. Object detection and recognition tasks become distributed problems in this case, and may rely on the entire grid to form a consensus.

Due to the high volume of information, and elevated hardware requirements that are generated in CV tasks such as video surveillance, environment and traffic monitoring, communication between nodes in the network becomes a problem. The transmission and storage requirements of computer vision algorithms, would strain the network, if out of the box algorithms are directly applied to the network. Detecting and recognizing objects that have been previously seen by any of the sensors in the network would involve a great exchange of information between the candidate node and all other nodes that may have seen the same object. [SPKK] proposes one of the following two scenarios:

1. broadcasting the original video content to all nodes, so that each one can locally process queries,
2. broadcasting the unknown object query to all nodes, and wait for a response by the network.

In each case, a substantial amount of overhead would strain the network's resources, so [SPKK] proposes a hierarchical dissemination of information where each node only stores part of the feature vector of the queried object, but the network as a whole contains all of the relevant data to answer any query. They define a flooding based framework that spreads the feature vector out to n hops away from the node which first viewed an object. Later queries travel up the disseminated chain to retrieve the answer, where each node in the chain can either reject the query as a non-match, or allow it to pass one hop closer to the source node which makes the final decision. Their method involved much overhead which must be brought down to a minimum such that network communications are not strained. We incorporate the method proposed by [S1] where the node that first spots a target broadcasts the feature vector of this target to all of its horizontal neighbors. A query is performed by searching for this feature vector through all of its vertical neighbors. This way, full network broadcasting is avoided. Compared to the method proposed by [SPKK], we achieve fewer messages passed in the network for an arbitrary query. In case the sought after object is seen for the first time in the network, its query will come back empty, but will have to traverse the entire height of the network in search of an answer. Since we first send queries towards the closest edge of the network from the query node, and then away from the closest edge, our method can take more hops to result in an answer than the query method proposed by [SPKK]. In our experiments, we determine that when there is a positive outcome in 30% or more of queries, our method requires fewer search hops.

## 2 Literature Review

The main issue with VSNs is the minimization of message size and reduction in overall network traffic. Uncompressed, or otherwise unedited visual information which is to be transmitted over the network requires high bandwidth, which makes it a natural selection for optimization in grid (mesh) network computing applications such as VSNs. [LDK] propose two methods of compressing and transmitting images in wireless sensor networks that save considerable energy. [YSV] present an energy efficient JPEG-2000 image transmission system over VSNs. [LLC, WA and WA2] articulate compression schemes for visual data that is to be transmitted though VSNs. The transmission techniques presented above were classified by [CWM] into single, multi hop, and finally end to end categories. [CWM] describe a forward error correction recovery mechanism for multi-path data transmission in VSNs and outline an algorithm for the tradeoff between end to end energy cost and reliability requirements of multi-path data transmission.

An algorithm for obtaining the 'vision graph' of a VSN is described in [CDR], where two nodes in such a graph are deemed adjacent if their cameras have predominantly overlapping fields of vision. This case is preferable when the 3D structure and position of objects is a desired outcome, but it increases data traffic between nodes, and therefore overall network throughput and processing load. [DR] propose a method of auto calibrating such network based cameras based on belief propagation. Here, camera node neighbors communicate directly and match scene points in order to perform calibration.

Apart from data compression and transmission algorithms, surrounding topics include data security, embedded visual systems and P2P VSNs. [LKZ] introduce a low complexity method of providing secure data transmission over VSN, which protects against eavesdropping attacks. [ABL] propose a system of traffic monitoring where individual cars and their license plates can be isolated. Arth and Bischof [AB] progress further in this field by developing an object recognition system based on an interest point detection linked to a vocabulary tree for real time surveillance. Their system is implemented on a DSP embedded device. In [PCPGM, QKRBS], the authors applied a multi-agent framework to the management of a surveillance system using a VSN. [FBBS] propose a distributed network of smart cameras for real-time tracking. They discuss the benefits of a distributed surveillance network compared to a host centralized approach. In [GB], the authors proposed a technique for tracking objects across spatially separated, uncalibrated, non-overlapping fields of view.

[SPKK] studies the problem of determining whether any of the (distant) nodes in the network has previously seen the same or a similar object compared to the newly acquired one at one of the nodes. Thus, it deals with knowledge distribution (feature distribution) in visual-sensor networks.

They propose a novel method for the distribution of features across a network of visual sensors, the hierarchical feature-distribution scheme (HFDS). Along with the HFDS, the candidate specifies four requirements, that have to be fulfilled by any recognition method, to ensure that the results of a recognition or matching in a distributed architecture will be the same as those in a non-distributed architecture. Abstrac-

tion (requirement 1) provides a function that translates level N features into more abstract higher level N+1 features with reduced dimensionality, and reduced storage needs (requirement 2). There exists a metric which provides a measure of similarity between two feature vectors at each level N (requirement 3), which converges, meaning that the measure at level N+1 is not larger than the measure at level N. The main idea is that if there is no match at a higher level N+1 then there is no match at lower level N (requirement 4).

[SPKK] discusses how one can map four basic pattern (object) recognition methods onto the distributed visual sensor network using HFDS. Those four basic methods are: template matching, histogram matching, subspace methods, and a random projection method. For each of those methods [SPKK] proves that they fulfill the four requirements, described above. This ensures, that they will be, recognition-wise, equally successful in a distributed scenario, as in the non-distributed scenario. A few selected possibilities to map state-of-the-art methods for representation of visual samples on the distributed structure are: histogram of oriented gradients (HOG), pyramid of histograms of orientation gradients (PHOG) and covariance region descriptor (COV).

[SPKK] selected the publicly available COIL-100 image database to test the proposed hierarchical feature-distribution scheme. It contains images of 100 different objects, each one rotated by 5 degrees, 72 images per object. Simulation was performed on rectangular 4-connected grid networks.

Three feature distribution methods were simulated. In ‘flooding at match’, the captured image is stored locally, and each object search task is performed by flooding the lowest level 1 feature vector in the whole network. The node with the previously captured image will perform matching and respond to the node that detected the new object. Flooding means that each node receives a copy of the feature vector. In ‘flooding-at-learn’, the captured lowest level 1 feature vector is flooded to the whole network. Therefore the node that detected the new image already has the knowledge of previous encounters of that object and can match immediately. [SPKK] proposes a third method, M-hier, hierarchical distribution scheme, the original feature vectors are flooded as follows. The detecting node is the only one with the highest level 1 feature vector. Its horizontal and vertical neighbors receive level 2 feature vectors from it. These neighbors in turn forward level 2 feature vectors to its horizontal and vertical neighbors in an expanding direction. This process continues until reaching the highest defined level H. Afterwards flooding will continue by expanding level H features further to the remaining nodes in the network. During flooding, the coordinates of the source node can also be propagated in addition to the feature vector. When a new copy of an object is detected, it is compared with locally available feature vectors, at the level where that feature is available. For those that match, the highest level 1 feature of the tested object are sent to the original source by backward links. The source node then can decide if there is a match. Comparison is included in the communication load on the network. It can be simplified by counting each transmitted feature vector of length L as load L (this is then proportional to message size). Please see Figure 1 for a depiction of the M-hier algorithm. The red, encircled node is the source from which the feature vector is propagated throughout the network.

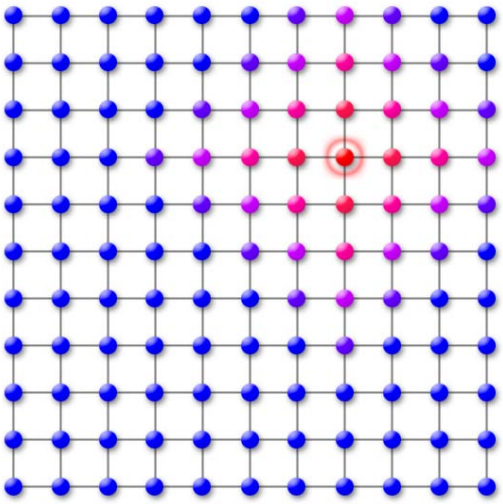

**Fig. 1.** - M-hier feature vector network dissemination

Experiments in [SPKK] are performed using histogram matching only. The number of bins is a power of two, and feature vectors at level N+1 histograms are obtained using the mapping which combines adjoining bins from level N. That is, sum of data in bin 1 and 2 at level N produces datum in bin 1 at level N+1, the sum in bins 3 and 4 produces datum in bin 2 etc. This object detection method has some limitations. First, it is a 'whole image' matching. Images contain mostly the main object and little background. Extraction of objects from larger images is not covered here, and can be done by separate image processing techniques. This limitation will be also applied in our work, which will instead concentrate on the network scalability issue.

The other limitation is that the correctness of object matching itself is not questioned here. Each judgment is assumed correct. Therefore there is no impact of threshold T on the performance, as ground truth is not established (only later in some real experiments to some limited extent). Similarly, this will also not be a focus of our investigation – we will mainly deal with the matching algorithm itself and its communication overhead.

The main remarks is that proposed M-hier algorithm is not sufficiently scalable. It is still based on flooding the whole network, which consumes bandwidth despite reducing the level of information. In the search phase, the lowest full size feature vector is still communicated between newly and previously detected locations.

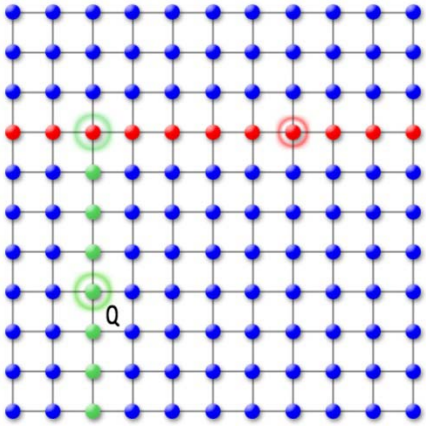


**Fig. 2.** - [SLJ] full feature vector horizontal dissemination, vertical query from node Q

[SLJ] overcomes message flooding deficiencies, by proposing a quorum-based location service. The destination node registers its location along a 'column' to form an update quorum. The source node makes a query along a 'row' to form a search quorum. The destination location is detected at the intersection between the update and search quorums. The overhead of each routing task, including location service, is $O(\sqrt{n})$, where n is the number of nodes in the network. In Figure 2, we depict the horizontal feature vector spread and vertical query method proposed by [SLJ]. The full feature vector is spread horizontally throughout the entire grid, and all queries are performed vertically.

## 3 Contribution

We address the scalability issue with the work of [SPKK], and correct it using the scheme proposed by [SLJ]. Essentially, we avoid the flooding strategy employed in SPKK to diffuse the feature vector of the target image throughout the network, and also shorten the hop count of the query message in order to get a result.

### 3.1 Quorum based image retrieval

The feature vector can be any array of features which follow the rules set out in [SPKK]. Choosing the most accurate feature vector for general object detection is a research area of its own, and not a focal point of this paper. We focus on the overall hop count, and minimizing message traffic in the network. For our purposes, we selected edge orientation histograms as the main feature vector. The feature vector of each image is transmitted to each horizontal node, and queries are done vertically as proposed by [SLJ]. We modify the query algorithm so that the query node first notes its location relative to the edges of the network, and performs the search up or down

first depending on its proximity to the border of the network. This way, fewer messages are passed in the network, at the expense of time required to get a result.

Queries performed in [SPKK] can only be answered at the source node, which means that each query must travel to it and back in order to be answered. In the worst case, there can be at most $2\sqrt{n}$ hops required to reach the node which contains the full feature vector, and another $2\sqrt{n}$ for the answer to reach the query node, where n is the number of nodes in the network.

### 3.2 Feature extraction from sub-images

We determine that images can be divided into k2 subimages by dividing rows and columns into k parts. A feature extraction method with d dimensions can be applied on each of subimages. This together gives k2d-dimensional feature representation. Feature reduction is then obtained by reducing 4 subimages into a single image (then k=1,2,4,8,…). The four properties given in [SPKK] can be proven for a wide range of specific feature extraction methods. This way, HLAC, SIFT, Viola's Haar wavelets etc. can replace the simple histogram based features.

### 3.3 Q-Hier based feature distribution

The direct improvement of M-hier [SPKK] is then Q-hier as follows. The feature vector of the detected object is distributed in its row only, instead of the whole network. Each search is then performed in the column of the query node, by transmitting the lowest level 1 feature vector. A match can be determined at the node which intersects the query column and the feature row. If there is no match, the search stops. In case of a match, the lowest level 1 feature is forwarded toward the original source, and can be tested similarly along the route, stopping with the first failure, or reaching it for final test (if it is the only node with the originally stored lowest level feature vector then only that node can make a positive decision). Compared to M-hier, row distribution may be unnecessary in case of the first mismatch. But flooding the whole network is avoided.

Note that if we have only one level of feature vectors (H=1) then Q-hier is simply a quorum based scheme. Since we will only simulate rectangular networks, it is then the basic row-column variant of it. Its superiority over M-hier for H=1 is then already demonstrated in the original papers on the quorum scheme [SLJ]. We implement this scheme in our experiments.

### 3.4 iMesh: multiple image copies

Next, [SPKK] assumes that one image is stored in only one node, throughout the process. This does not address the third, fourth etc. appearance of the same image. The node that discovered an object for the second time in the network can also serve, together with the original node, in matching for further appearances. If several copies already exist then iMesh from [LSS2] can be used. Again, for H=1 there is no difference in algorithm and performance gains compared to [LSS2].

## 4 Results and Discussion

We constructed a test set of 100 arbitrary images that were used to verify our theoretical results. The [SPKK] algorithm was compared with our own work on a 100 x 100 node grid. We chose to compare our Q-hier scheme with H=1 to their M-hier algorithm. The experiment was set up to choose a random image from the test set, and compute its feature vector based on its histogram of edge orientations. This vector was then diffused in the network via the schemes proposed by the competing algorithms. Random nodes were then chosen in the network that issued queries based on feature vectors computed from the other images in the test set. The hop count of retrieving an answer to a query was counted and compared between the two algorithms. It was observed that [SPKK] outperformed our algorithm in terms of hop count when the number of occurrences of the queried image not being in the network was very large. As the number of positively answered queries approached a threshold of 90%, our algorithm produced better results. This means that once the network becomes aware of its surroundings, our algorithm tends to outperform the scheme proposed by [SPKK]. We implemented the variant where only one row of the network contains the feature vector, which decreases message traffic, but increases the hop count of typical queries.

The input to the algorithm is a set of 100 images: (I1, I2, ... I100), and a grid network of size n x n. The expected output is the average number of hops required to determine if image I has been previously seen in the network. For each iteration of the experiment, a random node is selected as the source node in the network. The edge orientation histogram (or any variant of a feature vector) is spread horizontally to all nodes in the same row as the source node. Random nodes are then selected in the network and query images are assigned to them. Each query image is converted to its corresponding feature vector, and queries are processed vertically through the network. Feature vectors are compared using correlation to determine whether the query image is present in the network.

### 4.1 M-hier-H, Q-hier-H, M-hier-B, Q-hier-B: feature level distribution

Determining the best level distribution remains to be investigated. This is not resolved even in the original solution, because the initial node can calculate all levels and immediately flood the highest level to the whole network. This can be defined as method M-hier-H. There will be a reduction in communication cost, and no modest savings in the search phase, since tests at higher level would trigger more contacts to the source that eventually prove false. Similarly Q-hier-H can be defined, which restricts communication in rows and columns.

Further options include dividing these levels in different ways. For example, a balanced method may divide the number of rows (assume C=R for simplicity) R by the number of levels H, and reuse each level R/H times. This may define two new algorithms M-hier-B and Q-hier-B, respectively.

## 5 Acknowledgment

This work was partially supported by the following grant: "Digital signal processing, and the synthesis of an information security system", TR32054, Serbian Ministry of Science and Education.

## 6 References

[AB] C. Arth and H. Bischof. Real–time object recognition using local features on a dsp–based embedded system. Journal of Real–time Image Processing, Vol. 3, No. 4, pp. 233-253, 2008.

[ABL] C. Arth, H. Bischof, and C. Leistner, TRICam - an embedded platform for remote traffic surveillance. Conference on Computer Vision and Pattern Recognition Workshop, 2006.

[CDR] Z. Cheng, D. Devarajan, and R. J. Radke, Determining vision graph for distributed camera networks using feature digests, EURASIP Journal on Applied Signal Processing, Vol. 2007, No. 1, 2007.

[CWM] Y. Charfi, N. Wakamiya, and M. Murata, Trade-off between reliability and energy cost for content–rich data transmission in wireless sensor networks, 3rd International Conference on Broadband Communications, Networks and Systems, pp. 1-8, 2006.

[CWM] Y. Charfi, N. Wakamiya, and M. Murata, Challenging issues in visual sensor networks, IEEE Wireless Communications, Vol. 6, No. 2, pp. 44-49, 2009.

[DR] D. Devarajan and R. J. Radke. Calibrating distributed camera networks using belief propagation, EURASIP Journal on Applied Signal Processing, Vol. 2007, No. 1, pp. 221-221, 2007.

[FBBS] S. Fleck, F. Busch, P. Biber, and W. Strasser, 3d surveillance - a distributed network of smart cameras for real-time tracking and its visualization in 3d, Conference on Computer Vision and Pattern Recognition Workshop (CVPRW06), pp. 118-118, 2006.

[GB] A. Gilbert and R. Bowden, Incremental, scalable tracking of objects inter camera, Journal of Computer Vision and Image Understanding, Vol. 111, No. 1, pp. 43-58, 2008.

[LDK] V. Lecuire, C. Duran-Faundez, N. Krommenacker, Energy-efficient image transmission in sensor networks, International Journal of Sensor Networks, Vol. 4, No. 1, pp. 37-47, 2008.

[LKZ] W. Luh, D. Kundur, and T. Zourntos, A novel distributed privacy paradigm for visual sensor networks based on sharing dynamical systems, EURASIP Journal on Advances in Signal Processing, Vol. 2007, No. 1, 2007.

[LLC] Q. Lu, W. Luo, J Wang, B Chen, Low-complexity and energy efficient image compression scheme for wireless sensor networks, Computer Networks, Vol. 52, No. 13, pp. 2594-2603, 2008.

[LSS2] X. Li, N. Santoro, I. Stojmenovic, Localized Distance-Sensitive Service Discovery in Wireless Sensor and Actor Networks, IEEE Transactions on Computers, Vol. 58, No. 9, pp. 1275-1288, 2009

[PCPGM] M. Patricio, J. Carbo, O. Perez, J. Garcia, and J. M. Molina, Multi–agent framework in visual sensor networks. EURASIP Journal on Advances in Signal Processing, Vol. 2007, No. 1, 2007.

[QKRBS] M. Quaritsch, M. Kreuzthaler, B. Rinner, H. Bischof, and B. Strobl. Autonomous multicamera tracking on embedded smart cameras. EURASIP Journal on Embedded Systems, 2007.

[SLJ] I. Stojmenovic, D. Liu, and X. Jia, A scalable quorum based location service in ad hoc and sensor networks, International Journal of Communication Networks and Distributed Systems, invited paper, Vol. 1, No. 1, pp. 71-94, 2008.

[SPKK] V. Sulic, J. Pers, M. Kristan, S. Kovacic, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 21. No. 7, pp. 903 - 916, 2011.

[WA] H. Wu, A.Abouzeid, Energy efficient distributed image compression in resource-constrained multihop wireless networks, Computer Communications, Vol. 28, No. 14, pp. 1658-1668, 2005.

[WA2] H. Wu and A. A. Abouzeid. Error resilient image transport in wireless sensor networks, Computer Networks, Vol. 50, No. 15, pp. 2873–2887, 2006.

[YSV] W. Yu, Z. Sahinoglu, and A. Vetro, Energy efficient JPEG 2000 image transmission over wireless sensor networks, IEEE Global Telecommunications Conference (GLOBECOM '04), pp. 2738 - 2743 2005.