Similarity-based clustering strategy for mobile ad hoc multimedia databases

Bo Yang* and Ali R. Hurson

Department of Computer Science and Engineering, The Pennsylvania State University, University Park, PA 16802-6106, USA

Abstract. Multimedia data are becoming popular in wireless ad hoc environments. However, the traditional content-based retrieval techniques are inefficient in ad hoc networks due to the multiple limitations such as node mobility, computation capability, memory space, network bandwidth, and data heterogeneity. To provide an efficient platform for multimedia retrieval, we propose to cluster ad hoc multimedia databases based on their semantic contents, and construct a virtual hierarchical indexing infrastructure overlaid on the mobile databases. This content-aware clustering scheme uses a semantic-aware framework as the theoretical foundation for data organization. Several novel techniques are presented to facilitate the representation and manipulation of multimedia data in ad hoc networks: 1) using concise distribution expressions to represent the semantic similarity of multimedia data, 2) constructing clusters based on the semantic relationships between multimedia entities, 3) reducing the cost of content-based multimedia retrieval through the restriction of semantic distances, and 4) employing a self-adaptive mechanism that dynamically adjusts to the content and topology changes of the ad hoc networks. The proposed scheme is scalable, fault-tolerant, and efficient in performing content-based multimedia retrieval as demonstrated in our combination of theoretical analysis and extensive experimental studies.

Keywords: Wireless ad hoc network, content-aware clustering, data content distribution, multimedia data, semantic-based retrieval

1. Introduction

In recent years, mobile ad hoc networks have been increasingly popular in building temporary network connections in special areas, such as battlefields or disaster spots, where infrastructures are destroyed or too expensive to be built. An ad hoc network is a collection of cooperative mobile nodes that communicate with each other without the intervention of accessing points. These mobile nodes are capable of not only storing and processing data, but also performing complex operations through their communications, such as on-demand routing [4,5] or multimedia data streaming [2]. Within the scope of ad hoc networks, most of the previous research focuses on the routing protocols that adapt to the dynamic network topology [6]; however, information retrieval is becoming a hot issue in a variety of recent applications [3,37]. From the viewpoint of information retrieval, the organization of ad hoc networks can be classified into three categories [32].

Some earlier ad hoc network prototypes are linked to centralized data source nodes (data centers) that host constantly updated directories of data contents [4]. Queries issued from the client nodes are resolved

---

*Corresponding author: Bo Yang, 342 IST Building, Department of Computer Science and Engineering, University Park, PA 16802-6106, USA. Tel.: +1 814 863 3646; Fax: +1 814 865 3176; E-mail: byang@cse.psu.edu.

at the data centers and the results are forwarded back to the requesting nodes through unicasting. Such centralized organization does not scale well and has the single point of failure. Moreover, the data center behaves as a hotspot and its movement within the area could increase the network traffic.

The recently proposed ad hoc network frameworks are decentralized and have no data centers – the network is unstructured and mobile nodes form peer-to-peer connections to resolve queries through the cooperation with each other [7]. Flooding is the most common approach for information retrieval in such ad hoc networks, since the requesting node does not have any information of the data contents of other nodes and has to employ the blind search. The flooding approach consumes system resources, i.e., storage, bandwidth, and energy, drastically and hence offers good performance only when dealing with text information [21]. Considering the sheer size of the multimedia data, the performance deterioration is more drastic. In addition, the flooding strategy may cause duplicate queries and retrieval results, which may further increase the cost of the query processing [30].

To overcome the shortcomings of the blind search approach, alternatively, the literature has proposed the structured ad hoc networks in which the network topology does not change drastically [22,23]. In such networks, the data objects are not allocated to the network nodes randomly, but at specified locations that will make subsequent queries easier to satisfy [3]. Such design improves the efficiency of information retrieval in some cases at the expense of the flexibility and scalability of the ad hoc networks. In practical applications, the network topology and the data contents of the mobile nodes are constantly changing, which increase the difficulty of efficient multimedia data retrieval.

Motivated by the aforementioned challenges, we introduced a semantic-based, self-stabilizing scheme that uses an overlaid infrastructure to organize multimedia data sources in an ad hoc network. The fundamental idea is to organize multimedia data based on concise and abstract description of their semantic contents, and cluster data sources with similar data contents. With the help of the proposed scheme, the multimedia data retrieval in ad hoc networks becomes a clearly aimed searching process that offers reduced network traffic, energy consumption, and response time, regardless of the distribution, heterogeneity, and autonomy of the multimedia data sources.

The remaining part of this paper is organized into seven sections: Section 2 introduces the background information and related work. Section 3 outlines the preliminary concepts of the system model. Section 4 introduces the clustering of mobile nodes according to their data contents and presents the dynamic adjustment of the clusters in accordance with content and topology changes in an ad hoc network. Section 5 analyzes the performance of the proposed scheme. Section 6 evaluates the proposed scheme with experimental results. Finally, Section 7 draws the paper to conclusions.

2. Background and related work

2.1. Multimedia content extraction

Due to the sheer size of multimedia data, considerable research work has been done on extracting semantic contents that concisely describe multimedia objects [8–10]. In this work, the semantic contents can be obtained through the recognition of objects [34], which is performed in two phases:

- The multimedia object is first partitioned into several segments, which indicate the most significant visual/audio components [36], and
- The semantics of the segments are obtained through *Latent Semantic Analysis* (LSA) [8,10,11].

The segmentation method employed in this work is similar to the binary-partition-tree approach [9]. Similar pixels are merged together as homogeneous regions, and these small regions are recursively combined into segments of the multimedia data object.

The low-level feature representation obtained from segmentation cannot represent the semantic contents, and therefore does not provide an ideal basis for semantic-based retrieval. Hence, the LSA is employed to obtain the semantics of these segments. In this work, the data set of the LSA includes two sets of entities: the objects whose semantics are already known (training samples), and multimedia segments whose semantics remain unknown [8]. The LSA uses *Singular Value Decomposition* (SVD) to uncover the hidden semantic relationships (e.g. synonym and polysemy) between data objects [11]: the multimedia segments are classified into proper semantic categories after the singular value decomposition, and are assigned with proper semantics [13]. In addition, the latent semantic information obtained in the analysis process can be trained to show the personalized concepts from users [11], improving the accuracy of content representation.

2.2. Content-based retrieval

Based on the method in which multimedia data contents is extracted, the literature has recognized several content-based retrieval techniques [14–16,35,37]. The traditional centralized content-based retrieval approaches are based on feature representation of multimedia data that can be categorized as three classes:

- The partition-based retrieval approaches recursively divide the multimedia objects (or k-dimensional feature spaces) into disjoint partitions, with clustering or classifying algorithms, while generating a hierarchical indexing structure on these partitions. The earlier models in this class include Quad-tree [14], K-D-tree [15], and VP-tree [15]. The recent research has focused on models based on clusters [16,17]. Because clustering methods usually incur tremendous requirement of system resources, several schemes were proposed to reduce the dimension of feature vector space, thereby alleviating the system load [17]. The recently proposed dimension reduction methods include locality preserving projection (LPP) [38] and heavy frequency vector (HFV) [17]. The partition-based approaches normally employ very complex computations on features [15], which make these models inefficient for real-time multimedia applications. Moreover, the partition-based models do not consider feature space of different dimensions – i.e., multimedia objects from heterogeneous data sources [2].
- The region-based retrieval approaches employ small regions (either in the form of Minimum Bounding Rectangles or Spheres) to cover all the multimedia objects. Based on these bounding regions, a balanced tree is constructed as the indexing structure. This class of indexing models includes the R-tree and its variations (e.g. R*-tree, R$^+$-tree, and SR-tree) [18,35]. Relative to the partition-based approach, the region-based methods improve storage utilization by avoiding forced splits. However, due to the shapes of the bounding regions, the region-based approaches have the born weakness of overlapping [18], which deteriorates the searching performance. Furthermore, the data objects grouped in the same region may not share common semantic contents.
- The structure-based retrieval approaches use some fully-structural or semi-structural representation models to imply the semantic contents and to facilitate content-based multimedia retrieval. The literature reports research work on 3-dimensional modeling of multimedia data objects, such as B-spline-based facial expression [12] and cross-modal representation [8]. Some research work also proposes prototypes that enclose the semantic features of data objects, such as XML-based

mobile image representation and retrieval [37], while some others present logic-based or hierarchical organization of multimedia data objects [13]. Although these models achieve good performance in their defined domains, none of them have considered multimedia data manipulation in a unreliable and resource-restricted environment such as an ad hoc network.

The disadvantages of centralized strategy have motivated the research on various decentralized information retrieval models for ad hoc networks [19–21]. Some recently proposed models in the literature try to make use of data distribution to facilitate information retrieval [22–24]. *Content-Addressable Network* (CAN) model was proposed as a decentralized infrastructure that provides hash-table-like functionality on large-scale distributed networks [22]. A *Distributed Hash Table* (DHT) is employed to map the data objects into a k-dimensional logical Cartesian space, which allows the storage and retrieval of (key, object) pairs. Further improvements of CAN have led to the logical overlay networks (e.g. pSearch [23] and SSW [24]) that use dimension-reduction techniques to reduce search cost. These content-navigated retrieval models are based on the feature vector representation of object contents [23], which is inefficient in describing multimedia data: In practical multimedia database systems, the feature vectors are often of fixed sizes to facilitate the computation and representation. However, in many instances, some features may be null. The null features do not contribute to the semantic contents of multimedia objects; however, they still occupy space in the feature vectors – hence, lower storage utilization.

In an ad hoc network with heterogeneous distributed data sources, traditional centralized retrieval methods may not work efficiently. Due to the node mobility and dynamic topology, the centralized strategy may require extra network traffic to locate the data sources [5]. In many cases, the extra traffic load is formidably high to ad hoc networks, considering their limited bandwidths and computation resources. Moreover, the centralized data server behaves as the single point of failure, which deteriorates the robustness and scalability of the system.

2.3. Object-Based Indexing Infrastructure – Summary-Schemas Model (*SSM*)

The SSM was originally proposed as an object-oriented search engine that allows imprecise access to text information in a multi-database environment [25]. It is a global layer located on top of multiple pre-existing autonomous and heterogeneous local databases. The SSM comprises three major components: a thesaurus, the local nodes, and the summary-schemas nodes (Fig. 1). The thesaurus defines a set of standard access terms, the semantic categories they belong to, and their semantic relationships. It may utilize any of the off-the-shelf thesauruses (e.g. Roget's thesaurus) as its basis. A *Semantic Distance Metric* (SDM) is defined to provide quantitative measurement of "semantic similarity" between user query and the database contents [25,26]. A local node is a physical database containing physical data. A summary-schemas node is a logical data entity, a meta-data (summary schema), representing the concise and abstract contents of its children's schemas. Fewer terms are used to describe the information contents of a summary schema than the union of the terms in the input schemas while capturing the semantic contents of the input terms.

The SSM organizes data objects in a hierarchical fashion based on *synonyms, hypernyms* and *hyponyms* relationships. Synonyms are semantically similar objects in different data formats or same-format objects in different physical locations. The SSM employs synonym links to connect and cluster the "similar-content" data objects together. A hypernym is the generalized description of the common characteristics of a set of data objects. To find the proper hypernyms of data objects, the SSM checks the on-line thesaurus to obtain the mapping from data objects to hypernym terms. The higher-level hypernyms that describe the more comprehensive concepts can be obtained from abstracting the hypernym terms of data
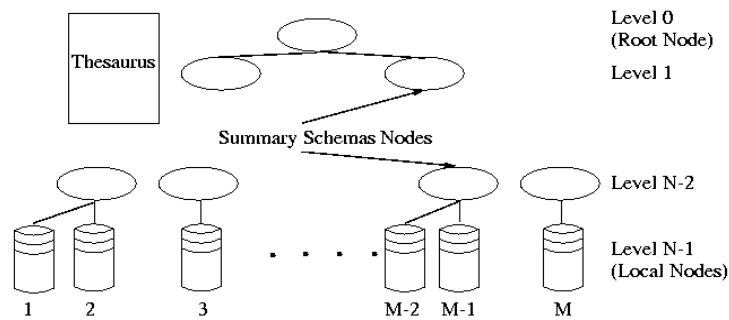
Fig. 1. The summary-schemas model.

objects. Recursive application of hypernym relation generates the hierarchical meta-data of the SSM. This in turn conceptually gives a concise semantic view of all the globally shared data objects. As the counter concept of hypernym, a hyponym is the specialized description of the characteristics of data objects. In the SSM prototype, the hyponym links compose the routes from the most abstract descriptions to the specific data objects. More details about the SSM can be found in [25–27].

3. Preliminaries

3.1. Extended Summary-Schemas Model (ESSM)

As mentioned before, the SSM was originally proposed for efficient management of textual and traditional databases in a wired multi-database environment. The original SSM prototype cannot be directly applied in accessing multimedia data in ad hoc networks due to several challenges: 1) Unlike textual and conventional databases, the multimedia data usually contain more complex semantic attributes (e.g. color, texture, and annotations) that cannot be represented by conventional data models; 2) The underlying fixed and predefined network topology of the SSM does not accommodate the infrastructure-free and dynamic nature of ad hoc networks.

Considering these limitations, we proposed an ontology-based framework – *Semantic Ad-hoc Image Retrieval* (SAIR) – that employs the linguistic characteristics of images to facilitate content-based retrieval in ad hoc networks [33]. In this paper, we strengthened the representation capability of the SSM, and proposed a clustering strategy for organizing multimedia data in ad hoc networks. The proposed scheme is called Extended Summary-Schemas Model (ESSM). The major properties of the ESSM can be summarized as follows:

– The semantic contents of multimedia data objects on each mobile node are extracted and represented using concise data-content-distribution expressions. The distribution-based representation approach we proposed in this work is capable of accommodating heterogeneity of the data sources. The details of this representation approach will be discussed in Section 3.2.
– The mobile nodes are partitioned into clusters based on the semantic similarity of their data contents. The mobile nodes with similar data contents are called *Content-Related Nodes*. And the clusters are denoted as *Content-Based Clusters*. These clusters are recursively aggregated together to form larger clusters. More detailed description will appear in Section 4.1.

3.2. Multimedia content representation

To represent the contents of multimedia objects in ad hoc networks, we present a novel semantic-based scheme, which can be briefly described as follows: Given a set of multimedia data objects distributed among a collection of mobile nodes, our goal is to find the data content distribution over the mobile nodes and set up the content-based relationships among these nodes.

The ad hoc network can be considered as an undirected connected graph $G = (N, C)$, where $N = \{n_1, n_2, \ldots, nr\}$ is the set of mobile nodes and $C = \{c_1, c_2, \ldots, c_s\}$ is the set of wireless connections between mobile nodes. $X = \{x_1, x_2, \ldots, x_m\}$ is the set of multimedia data objects. Each data object xi is represented as an n-dimensional semantic vector $\varphi_{xi} = (\omega_{xi}^1, \omega_{xi}^2, \ldots, \omega_{xi}^n)$. The elements in the semantic vector could be: manually added keywords, automatically extracted features, or relevance feedback generated by users. The distribution pattern of the data objects over the mobile nodes can be considered as a many-to-many relationship, i.e., each data object may be distributed among multiple mobile nodes, and each mobile node may contain a collection of data objects.

Figure 2 illustrates an ad hoc network and the data objects distributed among the mobile nodes. The multimedia data objects, represented as small circles, are considered as data points in the semantic space. Each mobile node ni in the network may contain a set of data objects, denoted as $\chi(n_i) = \{x_1^i, x_2^i, \ldots, x_r^i\}$, from the semantic space. The information contents of a node collectively represent an area in this semantic space (in Fig. 2, this is represented as a rectangle). As can be noted from Fig. 2, in this information space, mobile nodes may contain overlapping data (duplication) or semantically overlapping information. For example, both node A and node B contain the data objects in the semantic region of $\chi(A) \cap \chi(B)$. Generally, if an object x_j is distributed over a collection of mobile nodes, we use $\psi(x_j)$ to denote the set of mobile nodes containing the replicas of x_j.

Definition 1. The Semantic Similarity
The semantic similarity between two data objects is defined based on the Euclidean distance between their corresponding semantic vectors. Formally, the distance between two data objects x_i and x_j is defined as a cosine distance function $dist(x_i, x_j) = \cos^{-1} \frac{\psi_{xi} \cdot \psi_{xj}}{||\psi_{xi}||_2 ||\psi_{xj}||_2}$, where $||\varphi_{xi}||_2$ denotes the Euclidean vector norm of semantic vector φ_{xi} indicating the length of φ_{xi} in the Euclidean geometry $(||\varphi_{xi}||_2 = \left(\sum_{k=1}^n |\omega_{xi}^k|^2 \right)^{\frac{1}{2}}$ and $(\varphi_{xi} \cdot \varphi_{xj})$ is the scalar product of two semantic vectors φ_{xi} and φ_{xj} $(\varphi_{xi} \cdot \varphi_{xj} = \sum_{k=1}^n (\omega_{xi*}^k \omega_{xj}^k))$.

3.3. Retrieval formalization

The content-based multimedia retrieval (nearest-neighbor query) in an ad hoc network can be described as follows: Given the set of multimedia data objects $X \subset R^n$ and the ad hoc network G, for a specific integer constant k and a given query object x_q, return k data objects $X^* = \{x_1, x_2, \ldots, x_k\} \subset X$ such that for $\forall(x \subset X \wedge x \notin X^*)\ dist(x_q, x) \geqslant dist(x_q, x_i)(\forall i \in [1, k])$ is always satisfied. Formally, the nearest-neighbor retrieval is defined as:

Definition 2. The Nearest-Neighbor Retrieval
Given an object set $X = \{x_1, x_2, \ldots, x_m\}$ and a query object x_q, the nearest-neighbor retrieval of x_q within X, denoted as $\Xi^k(x_q, X)$, is the following set:

$$\Xi^k(x_q, X) = \{x_{i|}^| \forall x \notin \Xi^k(x_q, X), dist(x_q, x) \geqslant dist(x_q, xi) \wedge |\Xi^k(x_q, X)| = k\} \tag{1}$$

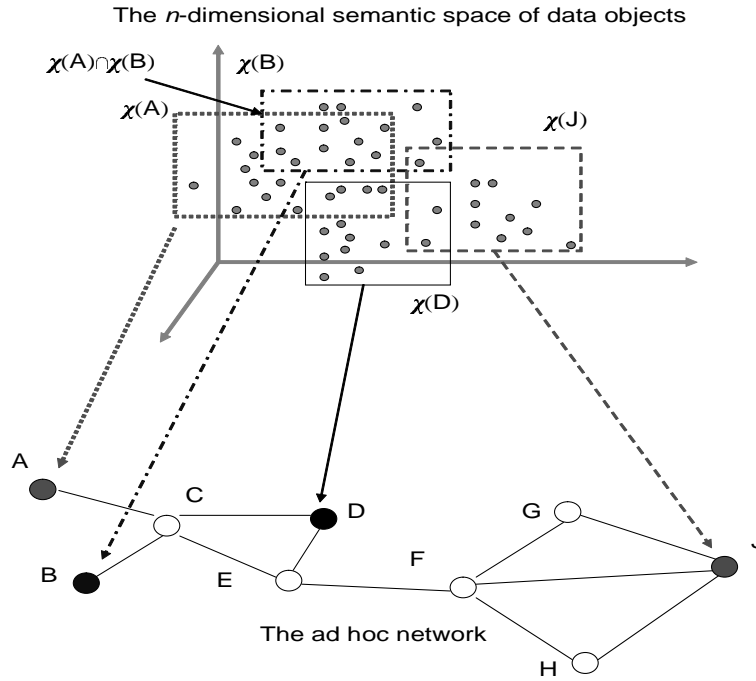The *n*-dimensional semantic space of data objects

Fig. 2. An illustrative example of the relationship between data objects and nodes.

In the context of ad hoc networks, the resolution of nearest-neighbor query may cause flooding and forcing pair-wise comparisons of data objects in each mobile node. The flooding approach is resource intensive and hence may not be applicable in real-time applications. Therefore, alternative approaches should be devised to perform the content-based multimedia retrieval with optimized search cost.

Definition 3. The Range-Constrained Nearest-Neighbor Retrieval

Given a data object set $X = \{x_1, x_2, \ldots, x_m\}$, a query object x_q, and a distance threshold d, the range-constrained nearest-neighbor retrieval returns the following data set:

$$\Xi_d^k(x_q, X, d) = \{x_i | \forall x \notin \Xi_d^k(x_q, X, d), dist(x, x_q) \geqslant dist(x_i, x_q) \land dist(x_i, x_q) \leqslant d \land$$
$$\Xi_d^k(x_q, X, d) | \leqslant k\} \tag{2}$$

Theorem 1. Given a multimedia data object set $X = \{x_1, x_2, \ldots, x_m\}$ and a query object x_q, there always exists a distance threshold d, satisfying $\Xi^k(x_q, X) = \Xi_d^k(x_q, X, d)$.

Proof: For a given distance threshold d, the multimedia data object set X is divided into two partitions – the objects whose semantic distance to x_q is less than d (i.e., within the sphere centered at x_q with a radius d) and the objects outside the sphere. By adjusting the distance threshold, a sphere that encloses exactly the same data objects in $\Xi^k(x_q, X)$ can be obtained. In other words, given the multimedia data object set X, the distance threshold can be considered as a function whose parameter is the number of nearest neighbors. Therefore, the threshold d satisfying $\Xi^k(x_q, X) = \Xi_d^k(x_q, X, d)$ always exists.

4. Content-based clustering

4.1. Clustering methodology

The main purpose of our semantic-based multimedia representation is to provide a paradigm that concisely describes the semantic contents of multimedia data objects and cluster the semantically similar ones. Based on this paradigm, the multimedia data contents of each mobile node can be automatically recognized and represented as semantic vectors with strong mathematical backbone. This representation would also allow simple and efficient analysis of similar data objects which assists classification, clustering, and integration of multimedia data. A clustering method that utilizes the aforementioned semantic-based paradigm is given in this section.

Definition 4. The Hyper-Rectangle Region

Given a set of multimedia data objects $X^* = \{x_1^*, x_2^*, \ldots, x_h^*\}$, each object x^* can be represented as a vector of attributes $\varphi_x = (\varphi_x^1, \varphi_2^x, \ldots, \varphi_x^n)$. The hyper-rectangle region $H(X^*)$ is a collection of intervals showing the n-dimensional minimum bounding region of $x_1^*, x_2^*, \ldots, x_h^*$:

$$H(X^*) = ([\min\{\omega_1^1, \ldots, \omega_h^1\}, \max\{\omega_1^1, \ldots, \omega_h^1\}], \ldots, [\min\{\omega_1^n, \ldots, \omega_h^n\}, \max\{\omega_1^n, \ldots, \omega_h^n\}]) \ (3)$$

Definition 5. The Node Content

Given a mobile node ni that contains m multimedia data objects $\chi(n_i) = \{x_1^i, x_2^i, \ldots, x_m^i\}$, the semantic content of node n_i, denoted as $S(n_i)$, is defined as a binary tuple:

$$S(n_i) = (H(\chi(n_i)), F_{dd}(.)), \tag{4}$$

where $F_{dd}(.) : H(\chi(n_i)) \to R$ (R denotes the real number set) is the data distribution density function in the n-dimensional hyper-rectangle region $H(\chi(n_i))$ satisfying $\int_{H(\chi(n_i))} F_{dd}(\chi(n_i)) = 1$.

Claim 1: Given the data object set $\chi(n_i)$, the distribution density function $F_{dd}(\chi(n_i))$ can be computed based on the density distributions of $\chi(n_i)$ in each dimension.

As noted in Definition 5, $F_{dd}(.)$ is a distribution density function in the n-dimensional semantic space R^n. The $F_{dd}(.)$ function can be defined deductively: $F_{dd}(R^n) = F_{dd}^*(R^{n-1}) * f_d(R|F_{dd}^*)$, where $F_{dd}^*(R^{n-1})$ is the distribution function in the $n-1$ dimensional sub-space and $f_d(R|F_{dd}^*)$ is the conditional distribution density in the nth dimension. Due to the independence among the dimensions, $f_d(R|F_{dd}^*)$ is equivalent to the mapping of $F_{dd}(.)$ along this dimension, which is denoted as $f_d^n(.)$. Therefore, $F_{dd}(R^n)$ can be represented as the product of the density functions along each dimension, denoted as $F_{dd}(R^n)\Pi_{j=1}^n f_d^j(R)$. For the multimedia data objects in the n-dimensional hyper-rectangle region $\chi(n_i)$, each object x is represented as a vector of attributes $\varphi_x = (\varphi_x^1, \varphi_x^2, \ldots, \varphi_x^n)$, hence, the density function of the jth dimension $f_d^j(\chi(n_i))$ is described as the probabilistic distribution over the interval $[\min\{\omega_1^j, \ldots, \omega_r^j\}, \max\{\omega_1^j, \ldots, \omega_r^j\}]$ ($r = |\chi(n_i)|$ indicates the number of objects in $\chi(n_i)$), and $F_{dd}(\chi(n_i))$ can be computed as the product of the 1-dimensional probabilistic distribution functions.

Definition 6. The Content-Related Nodes

Based on the aforementioned representation of multimedia data object, each mobile node ni can obtain a hyper-rectangle region $H(\chi(n_i))$ which encloses the data objects in $\chi(n_i)$. Given a pair of nodes n_0 and n_1, they are content-related iff:

$$H(\chi(n_0)) \wedge H(\chi(n_1)) \neq \emptyset \tag{5}$$

where $H(\chi(n_0)) \wedge H(\chi(n_1))$ is defined as the overlapping region of $H(\chi(n_0))$ and $H(\chi(n_1))$.

Definition 7. The Content-Based Cluster

Suppose an ad hoc network G comprises r mobile nodes $n_1, n_2, \ldots, n_r$. Let $n_i \approx n_j$ denote the content-related relationship between n_i and n_j, and $n_i \neq n_j$ denote that n_i and n_j are not content-related. Then a content-based cluster C_i is defined as follows:

$$C_i = \{n_i | \forall n_k \notin C_i, n_k \neq n_i; \forall n_j \in C_i, n_j \approx n_i\} \tag{6}$$

Based on the content distribution of each mobile node, the ad hoc network is divided into clusters with similar or overlapping multimedia data objects. A new cluster-level content distribution function is generated by describing the contents of mobile nodes within a cluster. A query, with high probability, will be directed and resolved within a specific cluster. This characteristic reduces the search space for a query, which should result in an improved query processing.

Definition 8. The Cluster-Level Hyper-Rectangle Region

Given a cluster $C_i = \{n_{i1}, n_{i2}, \ldots, n_{iq}\}$, the cluster-level hyper-rectangle region, denoted as $H_c(C_i)$, is the minimum bounding rectangle that encloses the hyper-rectangle regions of each node in cluster C_i:

$$\begin{aligned}
H_c(C_i) = (&[\min\{\min\{\omega^1(H(\uparrow (n_{i1})))\}, \ldots, \min\{\omega^1(H(\chi(n_{iq})))\}\}, \\
&\max\{\max\{\omega^1(H(\uparrow (n_{i1})))\}, \ldots, \max\{\omega^1(H(\chi(n_{iq})))\}\}], \ldots, \\
&[\min\{\min\{\omega^n(H(\uparrow (n_{i1})))\}, \ldots, \min\{\omega(H(\chi(n_{iq})))\}\}, \\
&\max\{\max\{\omega^n(H(\uparrow (n_{i1})))\}, \ldots, \max\{\omega^n(H(\chi(n_{iq})))\}\}])
\end{aligned} \tag{7}$$

where $\omega^j(H(\chi(n_i)))$ is the set of the jth feature values of the data objects in $H(\uparrow (n_i))$.

Definition 9. The Cluster-Level Content Distribution

Given a cluster $C_i = \{n_{i1}, n_{i2}, \ldots, n_{iq}\}$, the cluster-level content distribution, denoted as $S_c(C_i)$, is defined a binary tuple as follows:

$$S_c(C_i) = (H_c(C_i), F_{dd}^c(.)) \tag{8}$$

where $F_{dd}^c(.) : H_c(C_i) \to R$ is the data distribution density function in the cluster-level hyper-rectangle region $H_c(C_i)$ that satisfies $F_{dd}^c(H_c(C_i)) = \frac{\sum_{k=1}^{q}(|H(\chi(n_{ik}))| * F_{dd}(H(\chi(n_{ik}))))}{|H_c(C_i)|}$.

The cluster-level content distribution information is further integrated and fused together to form higher-level clustering, whose contents are represented using distribution functions in larger hyper-rectangle regions. This process is recursively performed until the whole ad hoc network is represented as one cluster. To accommodate the cluster-level content distribution information, a *centroid* node is selected for each cluster. The centroid is a node n^* who has most powerful capabilities, such as computation speed, memory space, communication bandwidth, and etc. In case there is a tie among more than one centroid candidate in a cluster, we choose the node whose data content has maximum overlapping with the cluster-level hyper-rectangle region of this cluster as the centroid node.

Definition 10. The Cluster Centroid

Given cluster $C_i = \{n_{i1}, n_{i2}, \ldots, n_{iq}\}$, let $V(n_{ij}) = \{v_1(n_{ij}), v_2(n_{ij}), \ldots, v_k(n_{ij})\}$ denote a vector of the hardware characteristics of node nij, such as computation capability and communication speed. Then the centroid of Ci can be defined as $\in(C_i)$:

$$\in(C_i) = \{n^* | \forall n' \in C_i, \sum_{s=1}^{k} c_s^* v_s(n') + |S(n') \wedge S_c(C_i)| \leqslant \sum_{s=1}^{k} c_s^* v_s(n^*) + |S(n^*) \wedge S_c(C_i)|\}. \quad (9)$$

where $c_1, c_2, \ldots, c_k$ are predefined coefficient parameters.

To reduce the searching cost, an indexing hierarchy is built as a wrapper on top of the content-based clusters. The construction of the indexing hierarchy is a two-step process:

(1) Within a cluster, the mobile nodes are grouped and one node is selected as the centroid of the cluster. This centroid node acts as the lowest indexing node in the indexing hierarchy (Definitions 11 and 12) – it contains the semantic data contents of the cluster. Connections between centroid node and other nodes provide a virtual infrastructure within the cluster.
(2) Based on the virtual infrastructures of clusters, the high-level indexing nodes are created by abstracting the data contents of several clusters. Similarly, each high-level indexing node is mapped to an ad hoc node that has most powerful capabilities among its children nodes.

4.2. Scalability and self-adaptive capability

The proposed indexing hierarchy has the self-organizing capability as the network conditions changes. The cluster centroid nodes are selected according to the hardware characteristics of mobile nodes which are likely to remain static and independent of the network topology. However, dynamic nature of the ad hoc network and the dynamic nature of the data contents of the mobile nodes require dynamic maintenance of the proposed indexing scheme to guarantee the optimized and accurate data retrieval.

Definition 11. The Capacity Entropy

Given a cluster $C_i = \{n_{i1}, n_{i2}, \ldots, n_{iq}\}$, let $V(n_{ij}) = \{v_1(n_{ij}), v_2(n_{ij}), \ldots, v_k(n_{ij})\}$ denote a vector of the hardware characteristics of node n_{ij}, and $P = \{c_1, c_2, \ldots, c_k\}$ be a vector of predefined coefficients. The capacity entropy of node n_{ij}, denoted as $E_c(n_{ij}, C_i)$, can be described as follows:

$$E_c(n_{ij}, C_i) = P \cdot V(n_{ij}) + |S(n_{ij}) \wedge S_c(C_i)|. \quad (10)$$

where $P \cdot V(n_{ij})$ denotes $c_1 \cdot v_1(n_{ij}) + c_2 \cdot v_2(n_{ij}) + \ldots + c_k \cdot v_k(n_{ij})$.

Because each cluster chooses the node with the largest capacity entropy as its centroid, the centroid nodes at top level of the indexing hierarchy should have the largest capacity entropy throughout the cluster. Suppose n' is the current centroid node. If another node n'' achieves the largest capacity entropy as a result of content changes, then n'' should be the new centroid node.

Based on the definition of capacity entropy, an adaptive scheme that dynamically adjusts the centroid node of a cluster can be proposed. The key point in this scheme is that the current centroid node of the cluster needs to dynamically keep updated information about the capacity entropies of the other nodes in its cluster. Since the hardware features of a mobile node n_i will keep unchanged for a long period, the only information that needs to be updated is the content summary of node n_i, which can be concisely described as a logic expression $S(n_i)$. The updated $S(n_i)$ can be obtained by allowing query packets to piggyback it when forwarded to the centroid node. The detailed update strategy is as follows:

– **Dynamic maintenance**: Each mobile node n_i can process a sequence of queries $Q_i^1, \ldots, Q_i^m$. For any content-based query Qit submitted to node n_i, if Q_i^t cannot be resolved locally, it is forwarded to the centroid node of n_i's cluster in an attempt to be resolved against the cluster-level content distribution information. In case of any changes to the $S(n_i)$, after the last query forwarded to the centroid node, the $S(n_i)$ will be attached with the query packet and forwarded to the centroid node. The new capacity entropy of n_i will be computed at the centroid node, and corresponding adjustments will be performed if a new largest entropy is found.
– **Node joining**: For a new node joining the ad hoc network, first an attempt is made to locate its content-related cluster and then its content is added to the indexing hierarchy.
– **Node leaving**: If a mobile node breaks down or leaves the network, its content summary will be removed from cluster centroid and ultimately from the indexing hierarchy.

5. Performance analysis

The performance of the proposed scheme is impacted by several factors as such: the initialization overhead of building the indexing hierarchy (including clustering and centroid selection), the cost of performing content-based multimedia queries, the overhead of maintaining the indexing hierarchy when network status changes, . . . etc. To facilitate content-based retrieval, these factors need to be analyzed in details.

Our analysis is based on the following notations:

– k: the number of nodes in the ad hoc network.
– m: the minimum number of children for an indexing node (minimum fan-out).
– P_J: the probability of a node joining the network.
– P_R: the probability of a node removed from the network.
– P_M: the probability of a modify operation.
– Q: the query rate.

5.1. Initialization overhead

In the initialization step, a randomly selected node, say n_i, is chosen as the coordinator to construct the indexing hierarchy. The selection of the coordinator node takes $\theta(k)$ hops. The coordinator needs to send each nodes a message to notify the coordinator's location and to collect data content descriptions, which takes $O(k).k$ hops. Hence, the initialization overhead amortized on each nodes is $\frac{\theta(k) + O(k)k}{k} = O(k)$ hops.

5.2. Searching cost

Resolution of a query requires at most $2 \log_m(k)$ steps to traverse the ESSM indexing hierarchy. Each forwarding operation takes at most k hops. Hence the average searching cost for a query in the proposed scheme is $O(k \log_m(k))$ hops. In contrast, the flooding strategy requires $\Omega(k^2)$ hops to resolve a query, since a network of k mobile nodes can have $\theta(k^2)$ connections, and a query may be transmitted on each connection at least once.

5.3. Maintenance overhead

As noted before, the proposed indexing hierarchy does not change as long as the data source contents are in tact. As a result, the indexing hierarchy changes if a new data source is inserted/deleted in/from the network or a modification is made to the contents of an existing node. Moreover, content changes on each node is piggybacked with unresolved query and forwarded to the cluster centroid to potentially trigger maintenance overhead due to the reconstruction of the indexing hierarchy.

A new node n_{k+1} joining the network requires at most $k \log_m(k)$ hops to be included in its related cluster. Hence, the average cost for processing new nodes is $2(P_J)Q \ k \ \log_m(k)$ hops. Similarly, the processing of leaving nodes requires $(P_R)Q \ k \ \log_m(k)$ hops. The modification operation can be viewed as a deletion of a node followed by an insertion of a new node. Consequently it requires $((P_M)Q \ k \ \log_m(k) + 2(P_M)Q \ k \ \log_m(k))$ hops. Finally, the average modification cost is $\theta((3P_M + 2P_J + P_R)Q \ k \ \log_m(k))$.

6. Experimental evaluation

In this section, we present the experimental analysis of the proposed content-aware clustering scheme against the flooding-based search scheme [4,5,30]. As noted earlier, in an ad hoc network with distributed data sources, flooding-based blind search strategy may cause extra computational and communication overheads due to the forwarding of the queries to some "irrelevant" nodes. In contrast with the flooding-based schemes, the ESSM organizes nodes with similar contents into clusters, and forwards query packets only to the content-related nodes.

The evaluation consists of a series of experiments conducted using both real data set and synthetic data set. Our comparative analysis is based on various performance metrics such as accuracy, search cost, scalability, and physical characteristics of the mobile nodes.

6.1. Experimental setup

The experiments were run on the basis of ad hoc network prototype with CMU extension to the ns-2 version 2.26 [29]. However, since ns-2 does not support content-based retrieval, a semantic-representation module was developed and added to facilitate multimedia data organization. In addition, the routing and data transmission processes were implemented under the framework of extended summary-schemas hierarchy.

The experiment was initialized by assuming a default number of pre-existing nodes in the network and randomly setting up the connections between the nodes. A mixture of operations, including querying, node joining, and node leaving, will then be randomly submitted to the network. The query generation time follows the exponential distribution, which is similar to the previous work [16]. The access pattern in the queries follows *Zipf-like* distribution, which is widely used to model non-uniformly distributed queries [35]. The input parameters are summarized in Table 1. Most of these parameters are self-explanatory. More details for some parameters are given below.

Node Movement Parameters: Each node in the experimental environment randomly selects waypoints within a 670 m * 670 m flat area. The node density can be adjusted by changing the number of mobile nodes in the range of 1 to 16,384 in the flat area. The node moving pattern follows the random waypoint movement model [5]; Initially, the nodes are placed randomly in the area. Each node selects

Table 1
Input parameters to the experimental system

Parameter	Default value
Number of nodes	128
Image dataset size	2,560
Texture feature type-1	Daubechies wavelet coefficients
Texture feature type-2	Tamura coarseness histogram
Texture feature type-3	Pyramid wavelet coefficients
Color feature type-1	Color histogram in HSV space
Color feature type-2	Color coherence vector in LUV space
Color feature type-3	First and second moment in Lab space
Environment size	670 m * 670 m
Transmitter range	100 m
Node mobility ($v_{\max}$)	1 m/s
Pause time	1 s
Bandwidth	1 M bps
Query packet size	512 bytes
Query rate (Qrate)	10 query/s

a random destination and moves toward the destination with a speed selected randomly from [0, $v_{\max}$]. After reaching its destination, the node pauses for a period of time and repeats this movement pattern.

Dataset Parameters: To examine both the accuracy and the scalability, we used two sets of experimental datasets as the test beds – the synthetic dataset and the real dataset as follows:

– The synthetic dataset employed in the experiments is similar as the one used in [34], which includes up to 65,536 data points in a 256-dimension feature space whose feature values are assigned by a random number generator abiding normal distribution in the interval [0, 1] on each dimension.

– The real dataset we used consists of 2,560 images of 32 semantic categories from the Corel dataset [36] (see Table 1). Each image in the dataset is represented as a vector of 435 features (color histograms and texture wavelet coefficients) and 4 annotation keywords. It is a large and heterogeneous image dataset. 2,048 images in the test dataset were used to train a semantic subspace learning module (i.e. LPP) integrated in the ESSM system that deduces the relationships among the keywords and the semantic categories. The extra 512 image were used as the candidate pool for query examples.

6.2. Retrieval process

The content-based multimedia retrieval under the framework of ESSM can be considered as a process of checking content distribution of the integrated information: the semantic content of the query (i.e. a semantic vector) is compared with a series of distribution density functions navigated in the ESSM infrastructure to find a cluster that contains the semantically most similar images. In the experiments reported in this section, the randomly generated query issued to a mobile node is resolved as follows:

1. The query object is represented as a semantic vector, which is forwarded to the ESSM infrastructure and compared against the contents of the clusters (hyper-rectangle regions and distribution density functions).

2. By comparing the overlapping of the query search sphere and the cluster-level contents, the ESSM finds a cluster that is semantically most relevant to the query. The centroid node of the selected cluster will then examine the semantic content of the query, and forward it to the mobile multimedia databases that may contain the semantically most similar objects.

Retrieved Images in 10 hops					
Retrieved Images in 20 hops					
Retrieved Images in 30 hops					
Retrieved Images in 40 hops					
Centralized Retrieval Result					

Fig. 3. The retrieval results with varied hop count.

3. For a specific mobile multimedia database, the content-based retrieval will employ a comparison metric including both semantic concepts and low-level features to find the nearest neighbors to the query object.

The ESSM shows its adaptation to the ad hoc environment in the experiments. As mentioned in Section 2.2, the centralized search strategy achieves good performance; however, it is not suitable for ad hoc networks. Therefore, the ESSM can be proved efficient if it obtains comparable results as the centralized search strategy. Consequently, we used the result from a centralized search system as the standard for content-based image retrieval. The performance of ESSM can be evaluated by matching the percentage of its search result against the centralized search scheme. Figure 3 shows the 5-nearest-neighbor retrieval results of ESSM with different hop-count constraints. The environment comprises 2048 images distributed among 128 mobile multimedia databases. The query is an image of a horse (the leftmost image in the bottom row of Fig. 3). As can be seen from Fig. 3, ESSM finds the images that are very similar to the centralized search. In addition, limiting the number of hops (10–30 hops) reduces the accuracy of the search results – i.e., results are not identical to the centralized search, however, a close observation shows that the returned results are somehow semantically related to the query image – i.e., they are mainly mammals. This shows that unlike the conventional feature-based CBR systems ESSM retrieves multimedia objects based on the gradually increasing granularities of semantic categories.

6.3. Retrieval performance evaluation

As mentioned earlier, the main focus in the development of the simulator is to evaluate the performance improvement of the proposed clustering scheme based on performance metrics such as search cost and accuracy:

– **Accuracy** is the percentage of the results generated by the search schemes (either the proposed clustering model or flooding) match with the results of centralized strategy. The higher matching percentage implies higher retrieval accuracy.
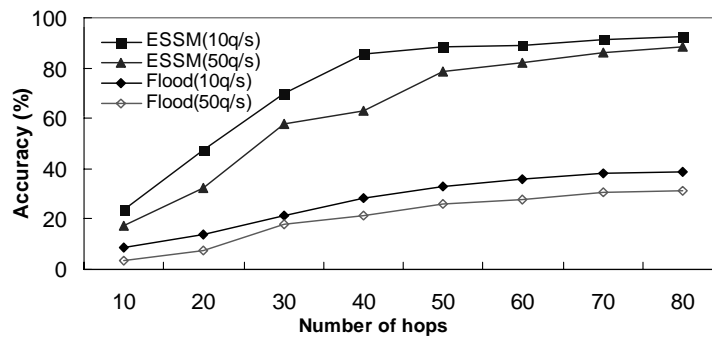
Fig. 4. Impact of hop count (query rate = 10 q/s).

- **Search path length** is the average number of hops spent on locating the data source node that contains the semantically nearest neighbors of the query image.
- **Search cost** is the average number of messages incurred during the process of query resolution. Flooding-based schemes may have short search path length, but their search cost is high due to the duplicate query replies.
- **Maintenance cost** is the number of messages spent on adjusting the indexing hierarchy according to the topology and content changes in the network.

6.3.1. Retrieval accuracy evaluation

The retrieval accuracy is evaluated for different simulation settings, varying the number of mobile nodes, the speed of node movement, the rate of query submissions, and the number of hops, during the query resolution process. In these experimental runs, we assumed 2,048 real images randomly scattered on mobile nodes. The maximum fan-out of the indexing hierarchy was set to 10, and the number of nearest neighbors in the content-based search was set to 10.

Figure 4 illustrates the impact of the number of hops on the retrieval accuracy. For a network of 128 mobile nodes moving at maximum speed of 1 m/s we limited the number of hops during the query resolution. Given a fixed setting of network scale (128 mobile nodes), the number of hops was varied from 10 to 80. As can be observed the retrieval results generated by the ESSM are more semantically related with the query image. Also note that both schemes (ESSM and flooding) achieves better accuracy as queries are generated at lower rate. This can be explained by the fact that higher query rate increases the work load of each node in storing and forwarding the queries to its neighboring nodes. The superior performance of the ESSM in contrast with flooding stems from its content-based clustering capability, since the search domain is restricted to a few clusters that are semantically most relevant to the query.

In another simulation run, we examined the impact of data density on the search accuracy. Using the same real image dataset, we varied the number of mobile nodes from 256 down to 32, increasing the average number of images per node from 8 to 64. With the fixed mobility and query rate (moving speed limit at 1 m/s and averagely 10 queries per second), the number of hops was limited to 64. Figure 5 depicts the experimental results. Since the probability of finding semantically related results increases as the data density increases, both schemes offer better accuracy as data density increases. However, the accuracy of flooding-based schemes is still less than that of the ESSM. Also note that the ESSM achieves over 90% search accuracy at a relatively low data density (16 images per node). This implies that in a relatively large-scale network (2048/16 = 128 nodes), the ESSM almost achieves its peak performance with a small search cost (less than 64 hops). In the scope of wireless ad hoc networks, this is an interesting observation since the mobile nodes usually have small storage and cannot support large data density.
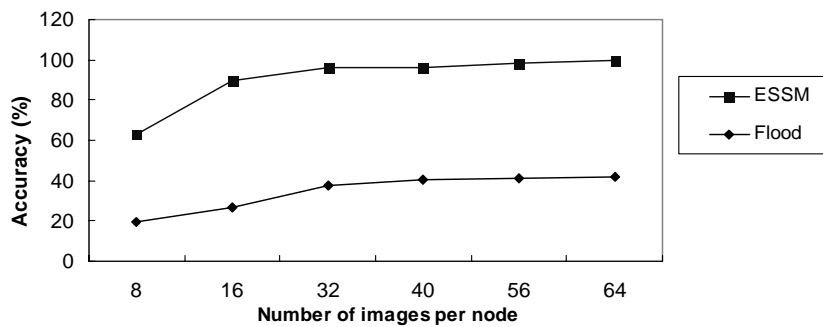
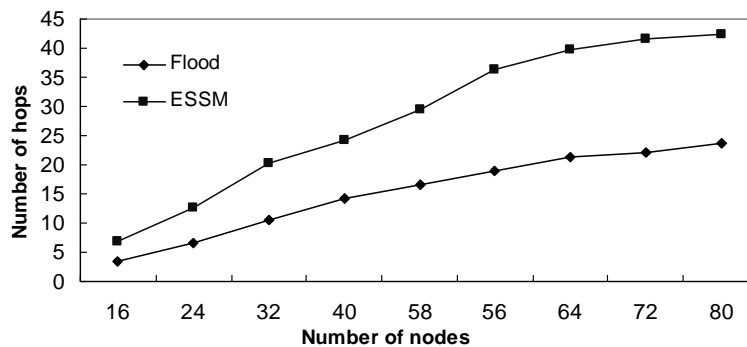Fig. 5. Impact of data density (query rate = 10 q/s).

Fig. 6. Comparison on search path length (query rate = 10 q/s).

6.3.2. Search cost evaluation

In an ad-hoc network, the search path length is usually calculated as the number of hops needed to deliver the query to proper mobile nodes that contain requested data. The real image dataset was used in this simulation run. Several factors, such as mobility, node number, and query rate, have direct impacts on the search cost; hence, we ran our simulator with different combinations of these parameters.

Figure 6 illustrates the number of hops needed during the query resolution for various network densities. As can be seen from Fig. 6, the flooding scheme finds the data source node using less number of hops. This is due to the fact that flooding broadcasts the query to all its neighboring nodes and reaches the data source node using the shortest path. However, due to the broadcasting and duplicate replies, the search cost of flooding may be formidably high.

Figure 7 shows the comparison of the search cost of both schemes in the same environment. As anticipated the ESSM retrieves the images using much less search cost than flooding-based schemes due to its content-based clustering characteristic. In addition, from Fig. 7 it can also be concluded that as node density increase, the ESSM offers even better performance than the traditional flooding schemes – steady performance improvement as the network scales up.

We also evaluated the impact of node mobility on search cost. For an ad-hoc network of 64 nodes the maximum speed of mobile nodes varied from 10 to 50. Figure 8 shows the simulation results. As can be concluded, the search cost of both schemes increases as the node mobility increases; however, the ESSM resolves queries at comparatively much less cost. This is due to the very nature of the flooding scheme – higher network traffic and higher workload at mobile nodes. In contrast, the ESSM resolves the queries within the scope of content-related nodes, and avoids forwarding queries to irrelevant nodes.
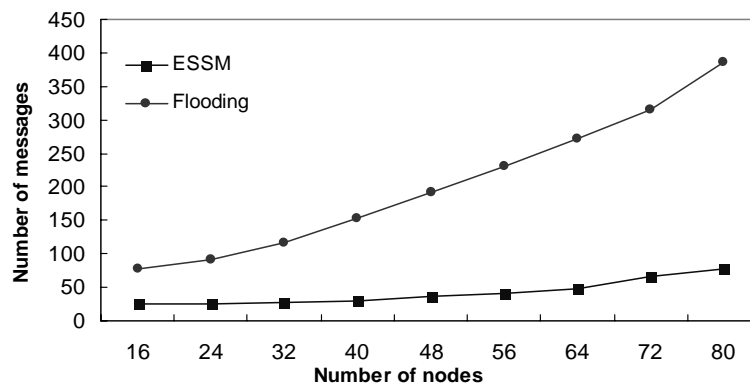
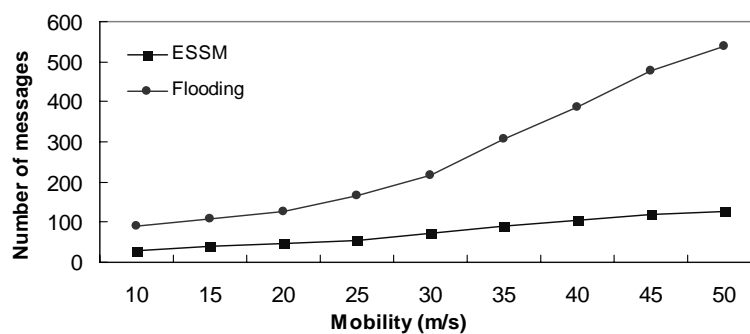Fig. 7. Comparison on search cost (query rate = 10 q/s).

Fig. 8. Impact of mobility (query rate = 10 q/s).

6.3.3. Scalability evaluation

In a separate simulation run, the search cost of both schemes was evaluated as the network scales up. In this simulation run, we varied the number of nodes from 128 to 512 and randomly distributed 65,536 synthetic data points on the mobile nodes. Figure 9 illustrates the result. Similar to our earlier observation (Fig. 7) one can conclude that the ESSM is scalable to large network sizes and large number of data objects.

6.3.4. Maintenance overhead

As mentioned in Section 5, self-adjustment capability of the proposed model incurs maintenance overhead that needs to be evaluated. In contrast to the ESSM, the maintenance cost of flooding scheme is very limited to the messages needed to update the neighborhood relationships between mobile nodes. Figure 10 shows the average search cost and the maintenance cost of both strategies as the network becomes denser. As one can conclude, even taking the maintenance overhead into account, the ESSM still offers a better performance than flooding.

7. Conclusions

We proposed a novel dynamic content-aware clustering scheme that facilitates content-based multimedia retrieval in ad hoc networks. This scheme is based on a concise descriptive framework –
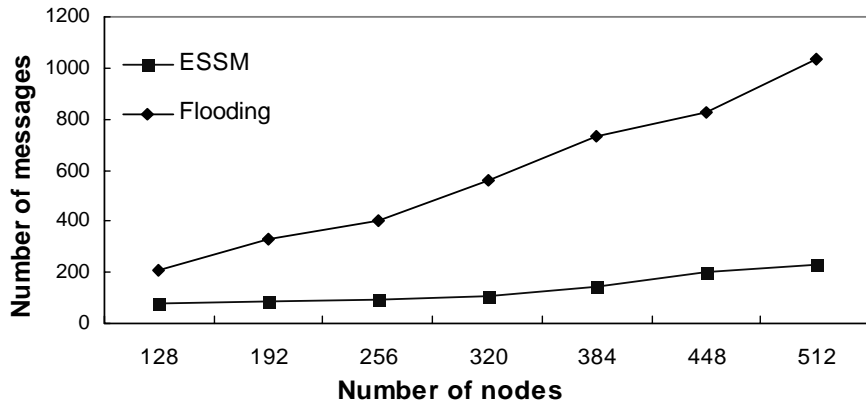
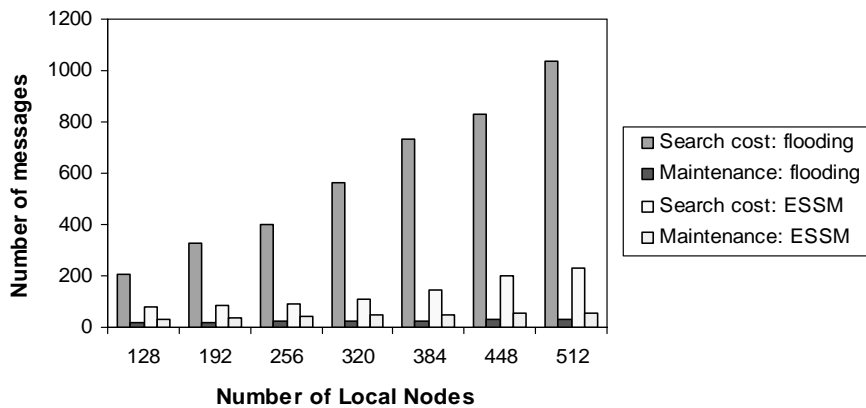Fig. 9. Search cost in large-scale networks (query rate = 10 q/s).

Fig. 10. Search cost and maintenance cost.

Extended Summary-Schemas Model (ESSM) – to represent the semantic contents of multimedia objects. It has several innovative characteristics such as logic-based content representation and semantic-based clustering of mobile nodes.

The proposed scheme makes use of the data content distribution in ad hoc networks to reduce the search cost without incurring high maintenance overhead. We have quantified the efficiency and effectiveness of our scheme with respect to various performance metrics – retrieval accuracy, search cost, and maintenance overhead. Through extensive theoretical and experimental analysis, we found that our content-based indexing methodology has the following features:

- The ESSM is a decentralized non-flooding search strategy performing content-based multimedia retrieval in ad hoc networks. As shown in our simulation results, it can achieve high accuracy while visiting only a small portion of mobile nodes.
- We employed semantic-based clustering to organize multimedia data – The content-related mobile nodes are grouped into clusters. As witnessed by simulation results, this approach reduces the search cost dramatically relative to the traditional flooding strategy.
- Our model is dynamic and capable of self-organizing itself as the network status changes. This further offers scalability and robustness in large-scale networks.

We are interested in integrating the proposed method with other mobile environments such as WLAN and cellular networks to allow global sharing among mobile nodes in larger distributed environments.

Acknowledgements

References

[1] G.N. Aggelou and R. Tafazolli, *QoS support in 4th generation mobile multimedia ad hoc networks*, Proceedings of Second International Conference on 3G Mobile Communication Technologies, 2001, 412–416.

[2] G. Auffret, J. Foote, C. Li, B. Shahraray, T. Syeda-Mahmood and H. Zhang, *Multimedia access and retrieval: The state of the art and future directions*, Proceedings of ACM Conference on Multimedia, 1999, 443–445.

[3] B. Li and K.H. Wang, NonStop: continuous multimedia streaming in wireless ad hoc networks with node mobility, *Proceedings of IEEE Journal on Selected Areas in Communications* **21**(10) (2003), 1627–1641.

[4] J. Broch, D. Maltz, D. Johnson, Y. Hu and J. Jetcheva, *A performance comparison of multi-hop wireless ad hoc network routing protocols*, Proceedings of ACM Mobicom, 1998, 85–97.

[5] T. Guven, C. Kommareddy, R. La and M. Shayman, *Measurement based optimal multi-path routing*, Proceedings of IEEE Infocom, 2004, 187–196.

[6] J. Sucec and I. Marsic, *Clustering overhead for hierarchical routing in mobile ad hoc networks*, Proceedings of IEEE Infocom, 2002, 1698–1706.

[7] C. Borcea, C. Intanagonwiwat, A. Saxena and L. Iftode, *Self-routing in pervasive computing environments using smart messages*, Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003, 87–96.

[8] M. Li, D. Li, N. Dimitrova and I.K. Sethi, *Audio-visual talking face detection*, Proceedings of IEEE International Conference on Multimedia and Expo, 2003, 473–476.

[9] P. Salembier and L. Garrido, Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval, *IEEE Transaction on Image Processing* **9**(4) (2000), 561–576.

[10] D. Li, N. Dimitrova, M. Li and I.K. Sethi, *Multimedia content processing through cross-modal association*, Proceedings of ACM Multimedia, 2003, 604–611.

[11] S.Y. Chen and X. Liu,, Data mining from 1994 to 2004: an application-orientated review, *International Journal of Business Intelligence and Data Mining* **1**(1) (2005), 4–21.

[12] J.S. Tang, A.W. Liew and H. Yan, Human face animation based on video analysis, with applications to mobile entertainment, *Journal of Mobile Multimedia* **1**(2) (2005), 133–148.

[13] B. Yang and A.R. Hurson, Ad hoc image retrieval using hierarchical semantic-based index, *Proceedings of IEEE AINA* (1) (2005), 629–634.

[14] H. Samet, The Quadtree and related hierarchical data structures, *ACM computing surveys* **16**(2) (1984), 187–260.

[15] A.W. Fu, P.M. Chan, Y. Cheung and Y.S. Moon, Dynamic VP-tree indexing for n-nearest neighbor search given pair-wise distances, *VLDB Journal* **9**(2) (2000), 154–173.

[16] H. Schneiderman and T. Kanade, Object detection using the statistics of parts, *International Journal of Computer Vision* (2002).

[17] J. Bao, J. Shen, X. Liu and H. Liu, The heavy frequency vector-based text clustering, *International Journal of Business Intelligence and Data Mining* **1**(1) (2005), 42–53.

[18] N. Katayama and S. Satoh, The SR-tree: An index structure for high-dimensional nearest neighbor queries, *Proceedings of ACM SIGMOD* **26**(2) (1997), 369–380.

[19] M. Castro, M. Costa and A. Rowstron, Should we build Gnutella on a structured overlay? *ACM SIGCOMM Computer Communication Review* (1) (2004), 131–136.

[20] V. Dheap, M. Munawar and S. Ward, *Parameterized neighborhood based flooding for ad hoc wireless networks*, Proceedings of IEEE Milcom, 2003, 1048–1053.

[21] B. Yang and A.R. Hurson, *Adaptive content-aware multimedia clustering in wireless ad hoc networks*, Proceedings of MoMM 2004, 35–44.

[22] S. Ratnasamy, P. Francis, M. Handley and R. Karp, *A scalable content-addressable network*, Proceedings of ACM Sigcomm, 2001, 161–172.

[23] C. Tang, Z. Xu and S. Dwarkadas, *Peer-to-peer information retrieval using self-organizing semantic overlay networks*, Proceedings of ACM Sigcomm, 2003, 175–186.
[24] M. Li, W. Lee and A. Sivasubramaniam, *Semantic small world: an overlay network for peer-to-peer search*, Proceedings of IEEE ICNP, 2004, 228–238.
[25] M.W. Bright, A.R. Hurson and S. Pakzad, A taxonomy and current issues in multidatabase systems, *IEEE Computer* **25**(3) (1992), 50–60.
[26] M.W. Bright, A.R. Hurson and S. Pakzad, Automated resolution of semantic heterogeneity in multidatabases, *ACM Transactions on Database Systems* **19**(2) (1994), 212–253.
[27] A.R. Hurson and M.W. Bright, Multidatabase Systems: An advanced concept in handling distributed data, *Advances in Computers* (32) (1991), 149–200.
[28] J.B. Kim and H.J. Kim, Unsupervised moving object segmentation and recognition using clustering and a neural network, *Proceedings of the 2002 International Joint Conference on neural networks* (*IJCNN '02*) (2) (2002), 1240–1245.
[29] NS Notes and Documentation, http://www.isi.edu/nsnam/ns/, 2004.
[30] C.E. Perkins, E.M. Royer, S.R. Das and M.K. Marina, Performance comparison of two on-demand routing protocols for ad hoc networks, *IEEE Personal Communications* **8**(1) (2001), 16–28.
[31] Computer Vision Test Images, http://www.iro.umontreal.ca/~mignotte/TestImages.html.
[32] B.F. Cooper and H. Garcia-Molina, Ad hoc self-supervising peer-to-peer search networks, *ACM Transaction on Information Systems* **32**(2) (2005), 169–200.
[33] B. Yang and A.R. Hurson, Hierarchical semantic-based index for ad hoc image retrieval, *Journal of Mobile Multimedia, Special Issue on Mobile Multimedia Computing and Communications* (2005), in press.
[34] W. Hsu, T.S. Chua and H.K. Pung, Approximating content-based object-level image retrieval, *Multimedia Tools and Applications* **12**(1) (2000), 59–79.
[35] N. Beckmann, H. Kriegel, R. Schneider and B. eeger, The R*-tree: an efficient and robust access method for points and rectangles, *Proceedings of Conference on Management of data* **19**(2) (1990), 322–331.
[36] B. Yang and A.R. Hurson, *An extendible semantic-based content representation method for distributed image databases*, Proceedings of IEEE MSE, 2004, 222–226.
[37] S. So, Efficient image indexing and retrieval over mobile devices, *Journal of Mobile Multimedia* **1**(2) (2005), 149–160.
[38] X. He, O. King, W.Y. Ma, M.J. Li and H.J. Zhang, *Learning a locality preserving subspace for visual recognition*, Proceedings of IEEE International Conference on Computer Vision, 2003, 385–393.

Bo Yang received the BS and MS degrees in computer science from Shandong University China in 1997 and 2000, respectively. He is currently a Ph.D. candidate of the Department of Computer Science and Engineering at The Pennsylvania State University. His research interest include semantic-based multimedia content management and mobile information retrieval. He is a member of IEEE society. 360E Information Science and Technology Building, Department of Computer Science and Engineering, The Pennsylvania State University, PA 16802, USA

Dr. Ali Hurson received his Ph.D. in computer science in 1980 from the University of Central Florida. He is a professor of the Department of Computer Science and Engineering at the Pennsylvania State University. His research for the past 20 years has been directed toward the design and analysis of computer architectures; multidatabases; and global information sharing environments. This research is supported by NSF, DARPA, ONR, NCR, Lockheed Martin, IBM, and Penn State. He has published over 230 technical papers in areas including computer architecture, parallel processing, hybrid dataflow architecture, database systems and database machines, object oriented databases, mobile computing, mobile data access systems, software agents, and VLSI algorithms.

Dr. Hurson served as the guest co-editor of special issues of the IEEE Proceedings on Supercomputing Technology, the Journal of Parallel and Distributed Computing on Load Balancing and Scheduling, the Journal of Integrated Computer-aided Engineering on Multidatabase and Interoperable Systems, IEEE Transactions on Computers on Parallel Architectures and Compilation Techniques, Journal of Multimedia Tools and Applications, and Journal of Pervasive and Mobile Computing. He is the coauthor of the IEEE Tutorials on Parallel Architectures for Database Systems, Multidatabase Systems: An Advanced Solution for Global Information Sharing, Parallel Architectures for Data/Knowledge Base Systems, Scheduling and Load Balancing in Parallel and Distributed Systems. He is also the guest editor of Advances in Computers for Parallel, Distributed, and Pervasive Computing. Professor Hurson is the co-founder of the IEEE Symposium on Parallel and Distributed Processing (currently IPDPS) and IEEE conference on Pervasive Computing and Communications. Professor Hurson has been active in various IEEE/ACM conferences and has given tutorials for various conferences on global information sharing, database management systems, supercomputer technology, data/knowledge-based systems, dataflow processing, scheduling and load balancing, and parallel computing. He served as a member of the IEEE Computer Society Press Editorial Board, and IEEE Distinguished speaker, editor of IEEE Transactions on Computers, and IEEE/ACM Computer Sciences Accreditation Board. Currently, he is serving as an ACM lecturer and editor of the Journal of Pervasive and Mobile Computing. 360F Information

Science and Technology Building, Department of Computer Science and Engineering, The Pennsylvania State University, PA 16802, USA