A Distributed Framework for Content Search Using Small World Communities

Seyyed-Mohammad Javadi-Moghaddam
Image, Video, and Multimedia Systems Laboratory
National Technical University
Athens, Greece

Stefanos Kollias
Image, Video, and Multimedia Systems Laboratory
National Technical University
Athens,Greece

***Abstract*—The continuous growth of multimedia content available all over the web is raising the importance of a distributed framework for searching it. One of the important parameters in a distributed environment is system response time. This parameter specially plays an important role in search and retrieval. A novel two-tier structure is introduced in this paper, which focuses on the community concept to facilitate creation of ontological small worlds that can effectively assist the search task. As a result, user queries are forwarded to nodes that are likely to contain the relevant resources. Evaluation of the framework proves that the small world character of the proposed structure provides queries with better route selection and searching efficiency.**

Keywords—Small-world networks; distributed multimedia model; ontology; community; fuzzy similarity

I. INTRODUCTION

Over the last decades, a media availability explosion has emerged with respect to multimedia content. Consumers can simultaneously access widely available digital processing electronic devices. On the other hand, the resulting large amount of generated multimedia content and content descriptive metadata all over the web has greatly increased the required content storage. This growth has encouraged researchers to utilize distributed models for content search and content management. In distributed systems, fault tolerance is advantageous, because failure of one component does not result in a complete system failure. In such systems, if a component fails, in most cases only part of the system will be disabled, and most of it will continue to be active. In fact, this failure will only reduce system performance. As a consequence, a distributed approach is preferable, due to increase of actual data storage and to avoidance of complete loss of system function in the presence of system component failures.

To design a distributed content search framework, this study focuses on some factors which play an important role in order to achieve improved performances. The first one is decentralization, in both system function and data storage. Secondly, an important factor is to achieve admissible performance reduction in cases that the system remains at least partially functional. Another essential factor is the availability of a flexible indexing technique that can be applied to different indexing scenarios. Moreover, an effective technique is needed to insert new elements into the index of the database, as the latter continuously grows. Finally, to retrieve data, a search method with good generalization ability which can be applied to all types of content such as images, audio, video was used.

So far, a number of distributed frameworks have been proposed for multimedia storage, but all of them use a centralized knowledge part, which is used to determine the location of multimedia data when a query is run or to find a location when the data needs to be stored. This results in a reduction in system efficiency as far as search, insertion and updating time are concerned. Moreover, to retrieve content, most models must send a request to a list of servers that have been selected based on specific criteria and then choose the best answer. So when there is a central infrastructure environment for content retrieval, a large number of messages is needed. Even in cases when the models use summarized requests, it is still needed sending the message to a lot of servers. There are some models that use small world networks to decrease number of messages but due to lack of attention to the community of the small world, they have not significantly reduced this number. Since, increasing number of messages in a distributed system would provide a reduced performance, *decreasing the number of these messages* is very effective in improving the situation.

One of the important parameters in a distributed environment is system response time. Clearly, smaller times are targeted, since they are of great importance in search and retrieval. As search or retrieval time gets smaller, the system performance becomes better. So *a more appropriate framework requests a small retrieval time*.

Lately, some researchers have proposed to use small-world networks for distributed multimedia information. They have imitated social networks. As humans keep track of descriptions of their friends and acquaintances, every media object can store descriptions of the objects that have maximum similarity with itself. Community plays an important role in these small world networks. Whenever, the members of community are many, the query execution time will decrease.

In this paper, appropriate enhancements in schemes which target community creation has been proposed. To achieve efficiency, the present study was focused on designing distributed multimedia frameworks which reduce the complexity of information sharing and time searching on a large-scale network. The proposed system relies on ontologies to describe the structure and semantics of multimedia properties. By gathering nodes with similar fuzzy ontological interests together, search gets more focused and efficient.

The proposed distributed multimedia framework focuses on decreasing centralization in small-world networks and consequently on increasing efficiency. In this context, each society operates as an independent server, permitting insertion or retrieval requests be sent to each community and *increasing decentralization*. In this framework, the members of each community are examined locally. Then, if nothing is found, communities related to the current one are examined, thus *reducing average response time*, as well as *average number of messages*.

In the following, Section 2 makes reference to existing related work. Section 3 presents the components which form the basis of the proposed system. Section 4 describes the proposed system architecture. Section 5 presents an evaluation study in which the proposed system provides improved performances, while section 6 gives conclusions and suggestions for further research.

II. Related Works

There are a lot of methods which have been proposed for creating distributed frameworks and platforms for multimedia systems [1, 2]. In [3] crisp clustering is addressed, using a clustering model with control constraints that is formulated as a linear optimization problem with Boolean variables. This model allows the control of the quality of clustering through adjustment of the related parameters. Amoretti[4] has provided an ontology-based Grid Service for searching multimedia content. In this work, too many local search services are connected to a global search service. Brut [5] has designed a framework that uses a central server and many remote ones. In this approach, a query is executed only on a set of relevant servers based on user queries, using semantic processing and available knowledge about the distributed servers, the multimedia content and the indexing algorithms. In [6] two components are used to perform distributed query processing: a multimedia application interface, that is a global query processing interface, and a distributed query content-retrieval engine. Laborie [7] has designed a framework for distributed multimedia that also uses a central server and many remote servers. It transfers only a concise version of the distributed metadata to the central server. When a set of servers has been selected by the central server as one that includes data likely to match a particular user query, the query can be processed locally on these servers. In [8] the application of Distributed Information Retrieval (DIR) is described, based on three main phases: resource description, resource selection and merging of results. Resource selection is executed using a centralized sample index of documents and a centralized ranking of sources based on the number and the position of their documents. Then, the user's query is forwarded to the selected sources and the retrieved source-specific results are merged into a single list using score normalization methods. In this framework, gathering information about the sources of a distributed system is the cause for the resulting computational overload.

In all these approaches, a centralized knowledge repository is used to feed the algorithms and perform content filtering. As was above mentioned, decentralization as well as efficiency are important and desired factors. Subsequently, some researchers have proposed approaches based on the small world network principle [9, 10] due to which a network has a short distance between nodes and a high decentralization ability. In general, small world networks mimic social networks. In these networks, community plays an important role. Androutsos [11] has applied the small network concept to his framework, but he has not paid the necessary attention to the community. Community structure is a common property that can be seen in many networks. Newman has shown that the community has a high density within edges and a lower density elsewhere (M. Newman and Girvan 2004). Therefore, affecting the identity of the members and their degree of cohesiveness can be considered.

One of the key issues for the management and retrieval of relevant information within a community is efficient content indexing. Indexing can be done according to a set of algorithms, which generate diverse and heterogeneous multimedia metadata, and in which resource consuming is high. To design a distributed multimedia systems, there are some choices that must be determined for indexing (Özsu and Valduriez 2011). Examples include using a fixed or variable set of algorithms, using algorithms executed over the entire multimedia collection, or only over a filtered sub-collection, indexing in a distributed manner - executed on the same location as the content, or in centralized mode by transferring the content to an indexation server, the latter being a decision which defines the distributed, or centralized, placement of multimedia metadata.

In the following,, a distributed framework has been proposed for content management and search using small-world networks. It is based on the community concept, while using "fuzzy similarity" to build cliques of data nodes, thus, locally performing most of the tasks. Multimedia data comprises two parts. The first includes multimedia components like text, graphics, animation, sounds, or video. The second part contains multimedia metadata and semantic annotation of multimedia content which can improve the services based on multimedia content. In this work, similarity is measured with regards to the above-mentioned second part of the data.

III. The Basic Components

A. Small-world networks

In late 1960, Stanley Milgram performed quantitative studies on the structure of social networks [9]. Milgram found that an average of six steps is needed for a letter to get from Nebraska to Boston. This finding has been further extended and called "six degrees of separation" [10].

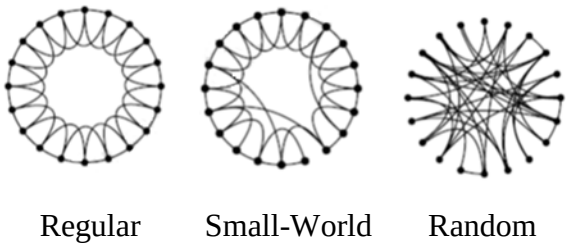

Fig. 1. Three types of networks [12]

In 1998, Watts *et al*. [12] proposed a model for small-worlds ranging between regular network and random distributed network (see Figure 1). It has been proved that some parameters such as size and average distance in a regular network are bigger than in a random network because of their topology [13]. Watts assumed a small probability p for a regular network's edges, and reconnected edges according to the value of p. This was called a rewiring action, with selection of a new node made absolutely at random. In this framework, each edge in a graph can be disconnected from its end point and be connected to another random node. By this action, the graph diameter can be reduced, because the farther nodes are reconnected to shorter ones. In this model, each node in the graph is considered equally likely to be rewired.

Newman and Watts [14] clarified that considering an equal probability for each node in the Watts-Strogatz model has two problems. First, shortcuts do not distribute totally uniformly and all positions are not equally appropriate for rewiring. Secondly, this model poorly defines the average distance between pairs of vertices on the graph because there is a probability of detaching some parts of the graph through the process of rewiring. Consequently, they kept the original links without variation, while adding shortcuts between pairs of vertices that were chosen at random. They also considered more than one link between any vertices.

SWIM [11] is a model resembling Newman and Watts's one, but has some different fundamental assumptions. At first, in contrast to the models in [12] and [14], which exclusively use undirected edges, all connections within the SWIM network are directed, similarly to the structure of the WWW.

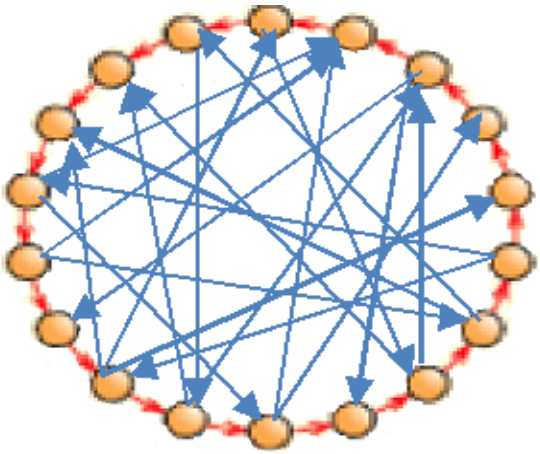

Fig. 2. The SWIM graph made by combining a similar graph and K-lattice

Specifically, in the SWIM system, an underlying directed *k*-lattice is embedded together with a directed pseudorandomgraph in an *N*-node network (see Figure 2). It should be noted that the SWIM network does not establish random connections between nodes according to a probability value (as in the Newman-Watts model). Instead, connections are built in accordance to a particular external distance that is computed using data stored locally at each node. There is one descriptor in all nodes that is used to calculate distances. The similarity graph is related to a specific descriptor and distance measure (see similarity section). Therefore, using a different descriptor for describing the data will lead to a different similarity graph. Similarly, the use of a different distance measure also results in a different graph. It is important to note that the k-lattice (called weak network) in SWIM is not dependent on the distance measure. Instead, it is only dependent on the order in which nodes are introduced into the network.

B. *Small world communities*

A common property that can be considered crucial for many networks, is the community structure, i.e., the division of network nodes into groups with the following property: network connections are dense within each of these groups, but sparse, between different groups. By analyzing such groups, it is easy to understand and visualize the structure of networks[15]. Small world graphs contain inherent community structure. Community structure is similar to clustering, with clusters being defined through similarity terms; distance is in general proportional to similarity between any two nodes.

A wide variety of similarity measures can be used for community generation. For example, some networks have built in natural similarity metrics. On the opposite, other networks have no natural metric, but suitable ones can be devised using correlation coefficients, path lengths, or matrix methods. In this paper, a number of communities are constructed according to fuzzy ontology-based similarity, measured using both ontological and fuzzy metrics.

C. *Ontologies*

From a structural point of view, an ontology is composed of disjoint sets of concepts, relations, attributes and data types [16]. Concepts are sets of real world entities with common features. Relations are binary associations between concepts. There exist inter-concept relations, which are common to any domain and domain-dependent associations. Attributes represent quantitative and qualitative features of particular concepts, which take values in a given scale defined by the data type.

An ontology is usually shown by graph G= (V, E) where the nodes of the graph represent concepts. Each edge expresses a kind of semantic relation between two nodes that represent these two concepts. V is a set of nodes and E is a set of edges. E includes two sub-sets, which are a set of hierarchical components and a set of non-hierarchical components respectively [17].

Ontologies have been developed for many purposes [18] targeting reusability and information sharing. To enable content to be discovered and exploited by services, agents and applications, the latter needs to be semantically described. Ontologies provide a formal description of the abstractions that represent content. Generation of ontological representations for a specific semantic area is of high importance and is continuously expanding in most content analysis cases today. Using ontologies is very useful for semantic processing of multimedia data. Multimedia metadata and semantic annotation of multimedia content are very important for retrieval and management of multimedia content.

Several metadata models and metadata standards have been proposed in which the scope and level of detail is different.

Saathoff (Simou, Saathoff, and Dasiopoulou 2006) has suggested the Multimedia Metadata Ontology (M3O) that is a framework in which both semantic and low-level metadata are integrated. This can be used for a variety of tasks, such as (Sjekavica, Obradović, and Gledec 2013) tagging or labeling of multimedia content, ontology-driven semantic analysis and retrieval of multimedia content, recommendation and filtering of multimedia content based on user preferences, personalization and retrieval.

D. Similarity

Similarity plays a fundamental role in the theories of knowledge and behavior. It serves as an organizing principle by which individuals classify object concepts and make generalizations. Similar, or dissimilar data appear in different forms: rating of pairs, sorting of objects, communality between associations, errors of substitution, and correlation between occurrences. Analysis of such data attempts to explain the observed similarity relations and to capture the underlying structure of the objects under study [19].

The assessment of similarity (from a domain independent point of view) has many direct applications, such as word-sense disambiguation [20], information retrieval [21], ontology learning, and biomedical applications [22].

E. Hierarchy and Semantic Similarity

To compute similarity in hierarchy ontologies, it is necessary to determine the way in which ancestral concepts, that are relative closely to a concept, can be identified. Moreover, it is needed how shared information of a pair concept within ontologies is specified. For example, Wu and Palmer [23] proposed a method that computes the number of is-a links (N1 and N2) from each term to their Least Common Subsumer (LCS) (*i.e.*, the most concrete taxonomical ancestor who subsumes both terms) and also the number of is-a links from the LCS to the root (N3) of the ontology.

$$Sim(a,b) = \frac{2*N_3}{N_1+N_2+2*N_3} \quad (1)$$

In (1) *a* and *b* represent two hierarchy ontologies.

Semantic similarity assessment is generally based on the estimation of semantic evidence observed in one or several knowledge or information sources. On the other hand, semantic similarity is non- hierarchical. In [17] the non-hierarchical components of an ontology are represented by an additional adjacency matrix. Different semantic relations are linked to each other based on their semantic similarity. Assuming there are *k* different semantic relations $R_1, R_2, \dots R_k$. Then, let β_1, β_2, … β_k represent the corresponding similarity factors (weights). A non-hierarchical semantic relation adjacency matrix *S* is defined as follows:

$$S_{ij} = \begin{cases} 1 & if\ i = j \\ \beta_i & if\ i \neq j\ and\ (i,j) \in R_i \\ o & otherwise \end{cases} \quad (2)$$

The semantic similarity measure may be generalized to a fuzzy semantic similarity measure, if the weights of the relation link are replaced by membership degrees indicating the strength of the relationships between parent and child concepts.

F. Ontology-based Fuzzy Similarity with respect to Imprecise Knowledge

Some models use imprecise knowledge to compute similarity [24]. Due to the fact that imprecise knowledge is acquired by experts/users, this knowledge should be rated, or the best part of it is chosen. In a previous work [25], similarity has been calculated by also giving credit to the experts/users. For example, in the area of European Fashion digitization and service provision, different Fashion stakeholders contribute in related knowledge generation, including experts working for fashion content providers, who annotate images of fashion objects or fashion events, or users who generate new fashion content and images, e.g. in social media. In fact, if the knowledge acquired by experts/users is going to be used in the semantic similarity approach, one should consider validation of experts to be a very important factor. One may give degrees of confidence to the actual expert skills, or to the actual confidence of the expert with respect to the specific topic under consideration. In this framework, using the validity factor, taking into account fuzzy properties of the actual validation process has been proposed. In all cases, when imprecise knowledge is present, similarity is computed based on exploitation of fuzzy properties, since imprecision is of rather fuzzy nature. In ontology-based cased-based reasoning (CBR), one typically utilizes two sets of case attributes: one set represents the database (DC), whereas the other one forms the query (QC). To compute the similarity, all members of these sets are considered. At the same time, the effect of validation in all imprecision situations was depicted, as was explained before. The validation degree can be considered as a weight for all imprecise values, because it depicts the amount of the acceptability. At first, let us assume that d_i and q_j are literals for DC and QC respectively (such as value of property attribute that is expressed in an imprecise way). In order to compute the overall fuzzy semantic similarity, including validation in the process, the following equation is proposed:

$$Fsim(QC,DC) = Agg(sim(q_i,d_j) * \max(\max\left(W_{vk} * FDC(d_{jk})\right), 1 - FIK) \quad (3)$$

where:

Agg is an aggregation function;

sim (q_i,d_j) is the similarity between the i_{th} literal value of DC and the j_{th} literal value of QC;

FDC is the fuzzy membership degree of the j_{th} literal value of DC, for the k_{th} expert;

FIK value may be either 0 or 1, as it denotes whether imprecise knowledge is considered in similarity evaluation or not;

W_{vk} is the degree of validation related to k_{th} expert.

In (3), the *max* function of Zadeh [26] has been adopted, because when a DC is used, two parameters are important: the fuzzy membership degree which shows the strength of the association between DC and QC literals, as well as W_v, which expresses the validation of this degree.

In addition and as already mentioned, it is also needed to take care of imprecision knowledge. Thus, when *FIK* equals 1, its impact within the inner *max* function of Eq. (3) is considered in order to compute the overall similarity. On the contrary, when *FIK* equals 0, the imprecision knowledge should not be utilized in the process. Consequently, in this case 1-*FIK* equals 1 and the output of *max* function becomes 1, thus neutralizing its effect. Finally, to combine both situations, a fuzzy union operator, which is depicted by the outer *max* function in Eq. (3) is utilized.

IV. The Proposed Architecture for Content Search

In the following,, an architecture that fulfills the requirements of improved performance of the search and retrieval process and of the indexing framework is described. This architecture has a two-layered structure. The lower layer provides the data resources, as well as specific services related to the storage system and the metadata repository. The upper layer includes high-level components, such as communities and small network links. Small-words are constructed by grouping nodes according to their fuzzy ontological similarity. In particular, the small-world layer organizes multimedia nodes in such communities, allowing nodes to efficiently locate areas of ontology similarity. The communities facilitate creation of the ontological small worlds. Accordingly, queries can be executed on the nodes that are likely to contain the relevant resource; as a consequence, a lower network load and a better search performance is achieved.

To make a community, firstly, the community coefficient (CC) is computed, defined as follows:

$$CC(Head, New\ Node) = \begin{cases} 1 & Sim(Head, New\ node) \geq th \\ 0 & Otherwise \end{cases} \quad (4)$$

The threshold (**th**) is determined according to the amount of similarity between nodes of the network. The amount of threshold plays an important role in framework structure because a small value of it causes creation of a big community. In the opposite, a big value of it results in a small community. After CC coefficient computing, if the CC value is equal to one, the node is inserted to the current community. Otherwise, it is moved to the next community head.

Figure 3 shows the proposed two-layer structure, including the physical layer and the small-world layer.

A. The Physical Layer

The physical layer comprises multimedia, as well as metadata collections. These collections can be organized in many ways. In the large scale distributed indexation of multimedia objects (LINDO) system [5], there are some remote servers that store and index all acquired multimedia information. Some modules are used to manage all needed tasks such as the *Storage Manager, t*he *Access Manager, t*he *Feature Extractor Manager, the Metadata Engine.* Laborie [7] proposed a scheme in which parts of every server store multimedia information. Moreover, a multimedia collection is stored on a server dedicated to acquire remote site information. A set of extractors is applied to a given piece of multimedia content returning a set of content metadata. The latter contain information about the media characteristics, while a metadata collection contains all content metadata describing the objects of the multimedia collection.

Chatterjee [6] proposed an architecture that includes a set of data nodes. Each data node has a multimedia database management system embedded in it. It also has a GridFTP server that takes care of the physical transfer of multimedia objects from one node to another. The data is basically stored in a data server. The multimedia database framework is divided into four major components: a multimedia interface, a core DBMS engine, a content-retrieval engine and a high-level relationship manager. These four components interact with each another to achieve the major functionalities, including queries, and updates.

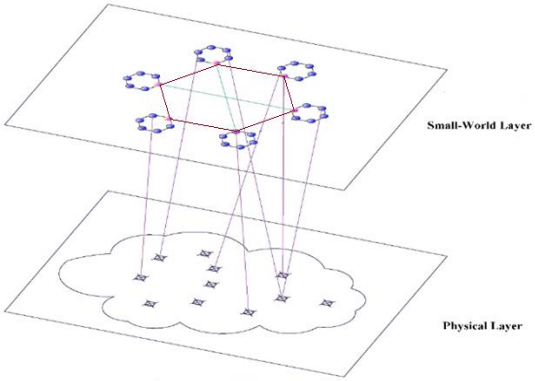

Fig. 3. The two-tier structure

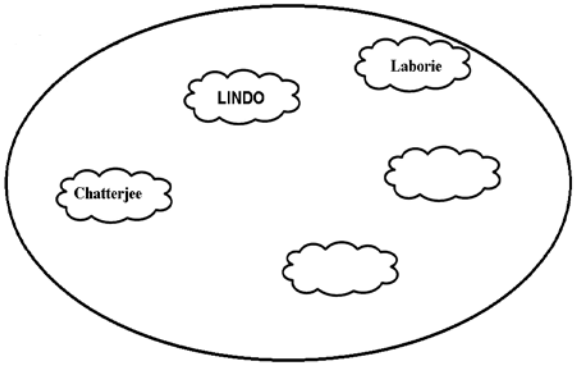

Fig. 4. Physical Layer

In the framework, the physical layer can include different storing policies for each node, as shown in Figure 4.

It is, however, necessary to generate a mapping which will help to meet the requirements of a small-network layer.

B. The Small Network Layer

With the help of this layer, nodes are connected according to their ontological similarity, forming small-world network communities. The topology of each community overlay can be flexible: if the community is large, nodes will be further broken up into small subcommunities to distribute multimedia nodes efficiently into the community, according to their fuzzy ontology similarity. Each community has a node, as

community index (shown, with red circle, in Figure 3), that is used to insert and retrieve nodes.

The layer's distributed nature stems from the fact that it attempts to exploit the small world phenomenon; i.e. all members of a social network retain information about respective small groups of peers and apply it to perform media indexing. As can be seen in Figure 3, the small-world network properties of the layer has been used. The nodes are located in this layer according to the amount of similarity they have with the head of each community. Every community has a head that is used to enter and search content in the community. The properties of small worlds are set by the heads. The latter locally store descriptions of themselves and their peers. The peers can be selected using a distance measure function, based on multiple criteria of similarity. As a consequence, clustering based on this formulation can result in improving system performance.

In addition, this layer attempts to make each community behave as an independent entity in the network in which it participates. Since a small-world network allows only few step searching between nodes (6 according to Milgram), search time is significantly reduced. To understand better the structure, the 'enter' and 'search' methods have been explained in the following.

C. Small World Layer Construction

The first node is considered as head of each community. When adding a new node, it is compared to the head. The following situations may be happen:

Similarity between head and new node is greater than a threshold:

It will be added to the community. Based on the similarity degree of nodes within the community, nodes can be sorted in descending order.

Similarity between head and new node is less than a threshold:

Relevant heads are compared to the new node. The following situations may occur:

There are communities, in which similarity degrees are greater than a threshold. In this case, the new node is added to the community with the highest similarity.

There is no community, in which a similarity degree is greater than a threshold. In this case, a new community, with this node as head, is created. Then, all other communities are sorted based on their degree of similarity. According to the number of peers, the unique number associated with the new community can compute its peers. These steps are shown in detail in Figure 5.

To evaluate the proposed framework, the image database of Lotus Hill Institute (LHI) has been used. The database provides large scale annotated ground truth data including extraction of edges, contours, contour attributes, segmentation, grouping, occluded contour completion, text, and object category recognition, 3D frames, UAV images, Google Earth images, video and cartoons. Part of the ground truth data are packed for various vision tasks and released one-by-one in xml format, together with respective Matlab code for reading and visualizing the annotation.

V. Evaluation Study

A. Experiment 1

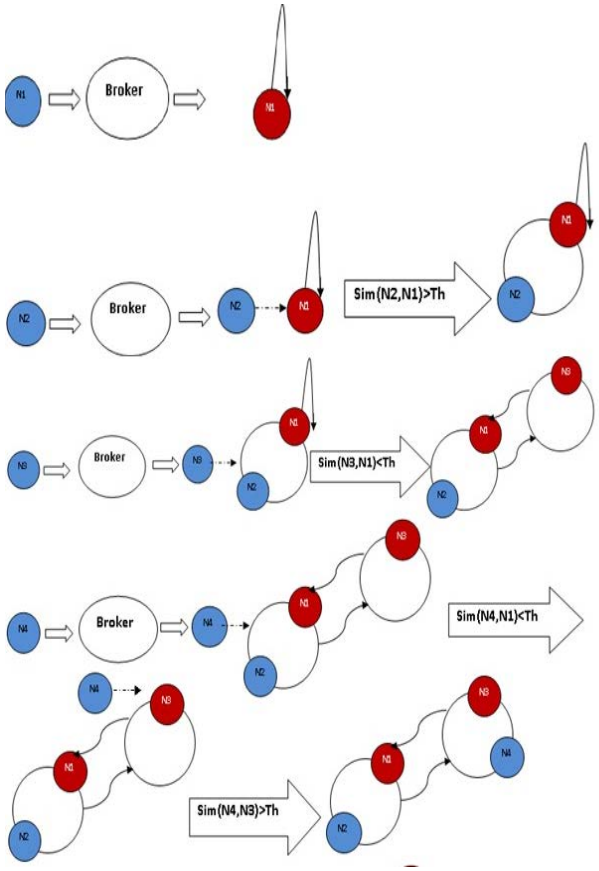

Fig. 5. The Small World Layer Construction

Then, the images of the database is used to test the proposed method. To build the community, the fuzzy XML similarity algorithm has been used as it is presented in [27].

One of the most important factors in evaluating the performance of the distributed system is the number of hops to locate an object; a hop corresponds to a move from one community or cluster to another. Reducing the number of hops is equivalent to improving the system's performance.

As was mentioned before, each node in the framework has some peers that are selected based on specific criteria. First, the number of hops is compared when the peers of maximum, or, of minimum similarity is selected.

First, the effect of an increased number of peers to the required number of hops has been measured. The result is shown in Figures 6 and 7. It can be seen that performance is improved when peers are added.

Finally, the required number of hops have been computed when using a normal clustering framework compared to the proposed framework. It can be seen that the proposed method achieves a better performance; performance gets higher, as the number of images increases (see Figure 8).

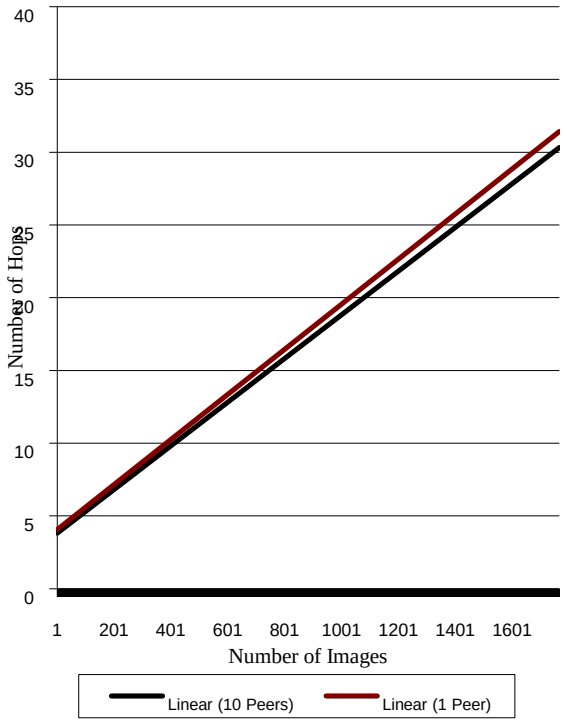

Fig. 6. The effect of increasing number of peers on the required number of hops

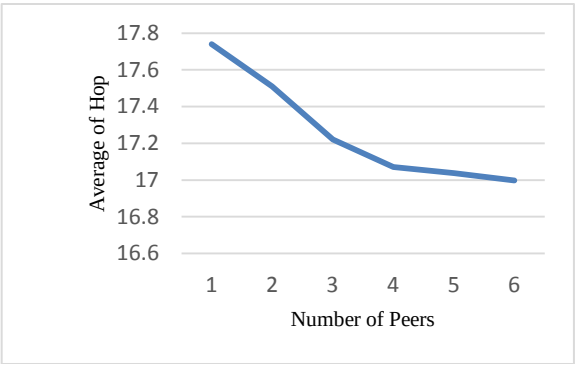

Fig. 7. The average number of hops when increasing the number of peers

B. Experiment 2

The dataset of experiment 2 was a set of 10000 homogeneous XML documents taken from the dblp DTD. The DBLP computer science bibliography contains the metadata of over 1.8 million publications, written by over 1 million authors in several thousands of journals or conference proceedings series. For computer science researchers the DBLP web site is a popular tool to trace the work of colleagues and to retrieve bibliographic details when composing the lists of references for new papers. Ranking and profiling of persons, institutions, journals, or conferences are the other controversial usage of DBLP. The bibliographic records are contained in a huge XML file. Many researchers simply need non-toy files to test and evaluate their algorithms. It is easy to derive several graphs like the bipartite person publication graph, the person-journal or person-conference graphs, or the coauthor graph, which are examples of a social network. Methods for analysis and visualization of these medium sized graphs are reported in many papers. To evaluate the proposed approach, the protocol develop by [3] was used.

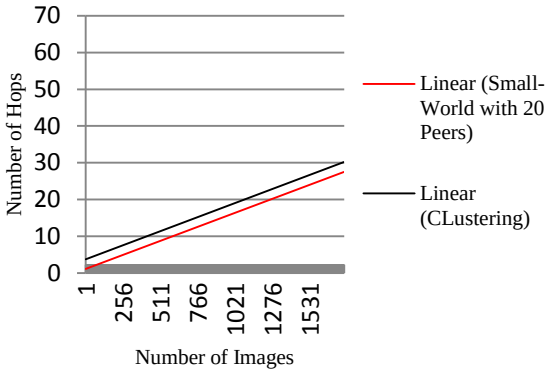

Fig. 8. The required number of hops when using a normal clustering framework and the proposed framework

The DBLP data set is available in http://dblp.uni-trier.de/xml/. The file dblp.xml contains all bibliographic records which make DBLP. It is accompanied by the data type definition file dblp.dtd. dblp.xml has a simple layout:

<?xml version="1.0" encoding="ISO-8859-1"?>

<!DOCTYPE dblp SYSTEM "dblp.dtd">

<dblp>

record 1

...

record n

</dblp>

The header line specifies ISO-8859-1 (\Latin-1") as the encoding, but in fact the file only contains characters <128, i.e. pure ASCII. All non-ASCII characters are represented by symbolic or numeric entities. The symbolic entities like é for the character '_e' are declared in the DTD. Numeric entities like é should be understood by any XML parser without declaration. In practice there are some obstacles in parsing the large XML file which cost us a lot of time: The SAX parser contained in the Java standard distribution has a limit for handling symbolic entities. When starting the Java virtual machine the option 'DentityExpansionLimit' has to be set to a large number. The Xerces-J from the Apache XML project reads dblp.xml without any problem. The DBLP FAQ3 reports more details. The XML root element <dblp> contains a long sequence of bibliographic records. The DTD lists several elements to be used as a bibliographic record:

<!ELEMENT dblp (articlejinproceedingsj

proceedingsjbookjincollectionj

phdthesisjmasterthesisjwww)>*

<article key="journals/cacm/Szalay08"

mdate="2008-11-03">

<author>Alexander S. Szalay</author>

<title>Jim Gray, astronomer.</title>

<pages>58-65</pages>

<year>2008</year>

<volume>51</volume>

<journal>Commun. ACM</journal>

<number>11</number>

<ee>http://doi.acm.org/10.1145/

1400214.1400231</ee>

<url>db/journals/cacm/

cacm51.html#Szalay08</url>

</article>

The experiment aims at providing an efficient way to search for xml files in the dblp dataset. The performance evaluation is done on a computer with Intel 3Ghz CPU and 4GB RAM.

Clustering of the files has been achieved using the concept of fuzzy similarity which was above described. In the presented results, shortcuts were built after clustering execution. Three parameters have been considered to evaluate the proposed system: the number of hops (similarly to Experiment 1), the number of shortcuts, and the number of files. In this experiment, at first the data size was changed from 500 to 10,000 xml files. All files were introduced in the clusters and then there was a search for them. Figure 9 shows the evolution of the number of total hops with respect to the data set size at different shortcuts (peers).

It is clear from Figure 9 that an increase of the data set size leads to an increase in the number of hops. This finding is in agreement with previous researches [3, 6, 8]. Moreover, increasing the number of peers from 5 to 10 decreases the respective number of hops when dealing with a small dataset (pink region in Figure 9); if, however, the dataset size increases, this effect is reduced.

On the other hand, the effect when changing the number of peers from 10 to 50 is not significant when the dataset size is small. On the contrary, it gets significant in larger datasets, e.g., when 10000 files are considered (blue region in Figure 9). A regression equation that describes the chart can be computed as follows:

$$Z = 0.0043 + 2.152X - 197.122Y \qquad R^2 = .7 \qquad (5)$$

Where X, Y, and Z are number of files, number of peers, and sum of hops respectively.

Finally, the required number of hops when using a normal clustering framework is compared to the proposed framework. It can be seen that the proposed method shows a much better performance, which requires less hops as the number of xml files increases (see Figure 10).

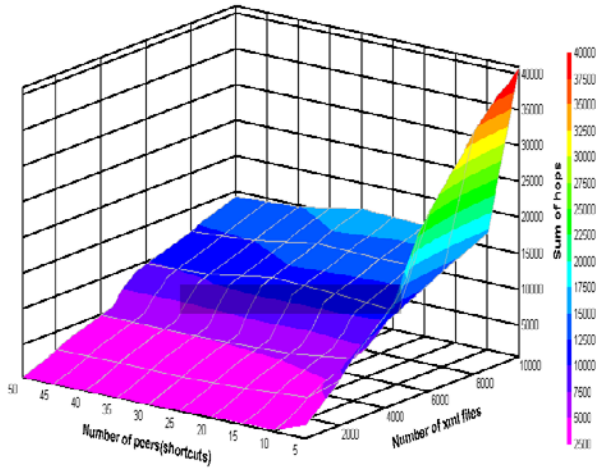

Fig. 9. Dependence of the hops on number of peers and dataset size

VI. Conclusions

This study has presented a distributed content search framework based on a two-tier architecture, consisting of a physical and a small-world layer. It was shown that this scheme provides efficient content search. In the presented experiments, the performance of the proposed small-world framework has been compared to a clustering system. The results showed that the small-world network requires a much smaller number of hops and subsequently results in lower content search time.

The proposed method can be of great importance for a variety of image and video retrieval and indexing. A great diversity of methodologies can gain from adopting the presented frameworks. In particular, the focus of [28] has been on reducing the data space in the considered clusters. However, in this approach, no relation between the clusters is taken into account, while the advantage of the framework is the small-world network which is built between the clusters Indexing approaches for image or video files have been presented in [29, 30]. However, in this approach one step goes forward, using an ontology-based fuzzy similarity, based on both semantic and structural issues. The centralization used in these techniques [28-31] for dealing with a distributed framework results in a reduction of the search efficiency. Obviously, gathering information within the distributed system causes a computational overload. This does not occur in the approach, since, in each node of the distributed framework, resource selection can be calculated without gathering information from other nodes – it was built it with regard to the shortcut list locally.

Other works (e.g., [32]) adopt (probably semantic based) user modeling techniques to capture users' evolving information needs. Such techniques can be considered as

particular cases of the approach, if it is focused on semantic similarity based on the interests of users.

One of the features of the proposed system is its ability to take into account semantic similarity between resources, possibly expressed in ontological form. There is a great number of publications related to knowledge technologies and multimedia content service provision [33]. Recent advances include creation of the “Billion Triple Challenge” [34] aiming at investigating the scalability of applications as well as the capability to deal with the specifics of data that has been crawled from the public web. Extending the proposed approach in this framework constitutes a topic of future work.

Application of the proposed method in a real cultural heritage environment [35] is currently under development.

REFERENCES

[1] F. Siqueira, "A Framework for Distributed Multimedia Applications based on CORBA and Integrated Services Networks," DSG, Department of Computer Science, Trinity College, Dublin, 1998.

[2] H. Muller, W. Muller, D. Squire, Z. Pecenovic, S. Marchand-Maillet, and T. Pun, "An open framework for distributed multimedia retrieval," RIAO, 2000.

[3] R. M. Alguliev, R. M. Aliguliyev, and I. Y. Alekperova, "Cluster approach to the efficient use of multimedia resources in information warfare in wikimedia," Automatic Control and Computer Sciences, vol. 48, pp. 97-108, 2014.

[4] M. Amoretti, D. Bianchi, G. Conte, M. Reggiani, and F. Zanichelli, "An ontology-based Grid service for multimedia search," in AICA 2004 XLII Congresso Annuale, Sessione: Tecnologie software GRID-oriented legate a Internet, 2004.

[5] M. Brut, D. Codreanu, S. Dumitrescu, A.-M. Manzat, and F. Sedes, "A distributed architecture for flexible multimedia management and retrieval," in DEXA 2011, pp. 249-263.

[6] K. Chatterjee, S. M. Sadjadi, and S.-C. Chen, "A distributed multimedia data management over the grid," in Multimedia Services in Intelligent Environments, ed: Springer, 2010, pp. 27-48.

[7] S. Laborie, A.-M. Manzat, and F. Sedes, "Managing and querying efficiently distributed semantic multimedia metadata collections,", IEEE MultiMedia, no. 1, 2009.

[8] F. Crestani and I. Markov, "Distributed information retrieval and applications," in Advances in IR, ed: Springer, 2013, pp. 865-868.

[9] S. Milgram, "The small world problem," Psychology today, vol. 2, pp. 60-67, 1967.

[10] J. Guare, Six degrees of separation: A play: Vintage, 1990.

[11] A. Venetsanopoulos, P. Androutsos, and D. Androutsos, "Small world distributed access of multimedia data," IEEE Signal Processing Magazine, vol.23, no.2, pp.142-153, 2006.

[12] D. J. Watts and S. H. Strogatz, "Collective dynamics of ‘small-world’networks," nature, vol. 393, pp. 440-442, 1998.

[13] H. Weichun, C. Min, and L. Jian, "A Small World Network Based Grid Resource Discovery Mechanism," IITSI, 2010 Third International Symposium on, 2010, pp. 8-11.

[14] M. E. Newman and D. J. Watts, "Scaling and percolation in the small-world network model," Physical Review E, vol. 60, p. 7332, 1999.

[15] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," Physical review E, vol. 69, p. 026113, 2004.

[16] M. Batet, "Ontology-based semantic clustering," AI Communications, vol. 24, pp. 291-292, 2011.

[17] L. Song, J. Ma, H. Liu, L. Lian, and D. Zhang, "Fuzzy semantic similarity between ontological concepts," in Advances and Innovations in systems, computing sciences and software engineering, ed: Springer, 2007, pp. 275-280.

[18] V. Cross, "Fuzzy semantic distance measures between ontological concepts," in Fuzzy Information, Processing NAFIPS. IEEE Annual Meeting of the, 2004, pp. 635-640.

[19] A. Tversky, Preference, Belief, and Similarity, 2004.

[20] G. Hirst and D. St-Onge, "Lexical chains as representations of context for the detection and correction of malapropisms," WordNet: An electronic lexical database, vol. 305, pp. 305-332, 1998.

[21] M. A. Rodríguez and M. J. Egenhofer, "Determining semantic similarity among entity classes from different ontologies," IEEE TKDE, vol. 15, pp. 442-456, 2003.

[22] D. Sánchez, M. Batet, and A. Valls, "Web-based semantic similarity: an evaluation in the biomedical domain," IJSI, vol. 4, pp. 39-52, 2010.

[23] Z. Wu and M. Palmer, "Verbs semantics and lexical selection," in Proceedings of the 32nd annual meeting on ACL, 1994, pp. 133-138.

[24] P. Alexopoulos, M. Wallace, K. Kafentzis, and D. Askounis, "Utilizing imprecise knowledge in ontology-based CBR systems by means of fuzzy algebra," IJFS, vol. 12, p. 1, 2010.

[25] S. M. Javadi-Moghaddam and S. Kollias, "The Important Role of Validation in Knowledge Intensive," in UKCBR, Cambridge, 2012.

[26] L. A. Zadeh, "Fuzzy sets," Information and control, vol. 8, pp. 338-353, 1965.

[27] S.-M. Javadi-Moghaddam and S. Kollias, "A Fuzzy Similarity Measure for XML Documents," IJITCS , vol. 13, pp. 9–17, 2014.

[28] T. Urruty, F. Belkouch, and C. Djeraba, "Kpyr: an efficient indexing method," in ICME, 2005, pp. 1448-1451.

[29] F. Idris and S. Panchanathan, "Review of image and video indexing techniques," Journal of visual communication and image representation, vol. 8, pp. 146-166, 1997.

[30] M. Worring and C. Snoek, "Semantic indexing and retrieval of video," in Pattern Recognition, 2008. ICPR 2008. 19th International Conference on, 2008, pp. 1-1.

[31] A. Kulkarni and J. Callan, "Document allocation policies for selective searching of distributed indexes," in Proceedings of the 19th ACM international conference on Information and knowledge management, 2010, pp. 449-458.

[32] F. Hopfgartner and J. M. Jose, "Semantic user modelling for personal news video retrieval," in Advances in Multimedia Modeling, ed: Springer, 2010, pp. 336-346.

[33] G. Stamou and S. Kollias, "Multimedia content and the semantic Web," Joh n Wiley & Sons, 2005.

[34] “Billion Triple Challenge 2012”, ISWC, Boston, USA, http://challenge.semanticweb.org.

[35] “Europeana Fashion project CIP-ICT-PSP,” 2012-2015, http://www.europeanafashion.eu.