A Multimodal Approach for Cultural Heritage Information Retrieval

Erasmo Purificato[1] and Antonio M. Rinaldi[1,2](✉)

[1] Department of Electrical Engineering and Information Technologies, University of Naples, Federico II Napoli, Via Claudio, 21, 80125 Naples, Italy
erasmopurif@gmail.com, antoniomaria.rinaldi@unina.it

[2] IKNOS-LAB Intelligent and Knowledge Systems - LUPT, University of Naples, Federico II Napoli, Via Toledo, 402, 80134 Naples, Italy

Abstract. The daily use of mobile devices and the expansion of the world-wide-web lead multimedia information to an uncontrolled growth. In this context, the use of smart interfaces and the combination of different features in the information retrieval process are crucial aspects. In particular, for a cultural heritage application it is important to consider that a digitized artwork is only a representation of a real object, represented under specific conditions (camera position, brightness, etc.). These issues could be causes of alterations during the features extraction task. In this paper we propose a multimodal approach for cultural heritage information retrieval combining geographic and visual data. Our approach has been implemented in a mobile system based on open source technologies. It is composed of three main parts related to image matching functionalities, Geographic Information Retrieval task, and a combination strategy for multimedia and geographic data integration. An Android application has been developed to give a user friendly interface and a case study together with some experimental results are presented to show the effectiveness of our approach for the user satisfaction.

1 Introduction

Art and culture have always represented an important part in human lives. Over the years, public and private organizations have preserved and used our cultural heritage as fundamental sources of education and learning. Due to these reasons, in recent times, there is a great interest on the archiving of ancient historical and cultural materials in digital forms for future generations [1]. Digitalization is a very important task, because computers and mobile devices have become the principal medium for learning and visiting digital art galleries from any corner of the world; furthermore, artworks tend to be preserved from deterioration and natural or human damages [2]. Such increasing digitization has lead to uncontrolled growth of data with several issues very closely the Big Data challenges [3] in every phase of the data life cycle, from storage to analytic and visualization [4–6].

O. Gervasi et al. (Eds.): ICCSA 2018, LNCS 10960, pp. 214–230, 2018.
https://doi.org/10.1007/978-3-319-95162-1_15

In such a scenario, a significant role is played by a set of technologies generally referred to as *Content-Based Image Retrieval* (CBIR). Also known as Query By Image Content (QBIC) or Content-Based Visual Information Retrieval (CBVIR), CBIR is the application of computer vision to image retrieval. Its goal is to limit the use of textual descriptions and to develop techniques for retrieving images on the basis of automatically-extracted multimedia features, such as colour, edge, texture and shape, from a user query image or user-specified image features [7]. The image content refers to measurable visual properties. As an example, for a computer scientist the content of a painting like *Mona Lisa* is a combination of textures, edges, shapes and colours, while for a tourist at the Louvre museum the content normally refers to represented the person, the author Leonardo Da Vinci, painting techniques, historical models.

In the context of CBIR, a large number of methods have been proposed and investigated but, as stated in [8], they still not provide general solutions.

On the other hand, spatial information has becoming an important feature in the information retrieval process. The concepts of *Geographic Information Retrieval* and *Spatial Query* are explained starting from the interpretation of Larson in [9]. He defines GIR as a specialization of the terms *Information Retrieval* (IR) and *Geographic Information System* (GIS) which provide access to georeferenced information sources. IR means finding information resources of an unstructured nature relevant to an information need (user query) from a document collection (e.g. web) [10]. Moreover, Geographic Information System is defined as the development and use of theories, methods, technologies and data for understanding geographic processes, relationships and patterns [11]. Eventually, Geographic Information Retrieval could be described ad as the set of technologies and processes to index, query, retrieve and browse georeferenced information. GIR is supposed to be able to better understanding the geographic knowledge in documents and user queries, and provides a more accurate answer to user needs. From this point of view, *spatial queries* could be recognized as queries about the spatial relationship of entities geometrically defined and located in the space. This kind of query requires that the space will be well-defined using a coordinate systems of the "real world". The use of different features related to objects and user behaviour could improve the quality in the whole retrieval process [12].

In this paper, an approach to integrate Image Content matching techniques, *Geographic Information Retrieval* (GIR) and spatial queries is proposed. Our approach has been implemented in a system based on a mobile interface.

The remainder of the paper is structured as follows: in Sect. 2 some of the principal recent works produced in last decades about Content-Based Image Retrieval and Geographic Information Retrieval are discussed; the architecture of the proposed system and the Android application developed as case study are described, respectively, in Sects. 3 and 4; some experimental results are presented and discussed in Sect. 5; eventually, conclusions and future works are presented in Sect. 6.

2 Related Works

Content-Based Image Retrieval is a fervent research topic where theories and technology to organize digital image archives represented by their visual contents are studied and developed. This community is composed by people from different fields, such as computer vision, data mining, human-computer interaction, information retrieval, statistics, information theory and also psychology. Smeulders published a paper presenting a review of two hundred references in CBIR [13]. According to his work, CBIR applications can be divided into three board categories: Search by association, Search at specific image and Category search. *Search by association* is a class of methods and systems aimed at search in large sets of images from unspecified sources. It implies iterative search refinement with similarity measure or examples with which the search was started. This kind of systems are typically highly interactive. The search results can be manipulated interactively by relevance feedback [14]. *Search at specific image* aims to obtain an accurate example of the same object represented in a user image. These systems are suitable for searching for stamps, artworks, industrial component, and catalogues. *Category search* goal is in the retrieval of a random image representative of a specific class of objects. Categories can be derived from labels or inferred from databases [15]. CBIR systems represents images by numeric values, called *features* or *descriptors*, that are used to describe the properties of images and allow the retrieval of visual information. There are two approaches called discrete and continuous respectively. The *discrete approach* is inspired by textual information retrieval and uses techniques like text retrieval metrics and inverted files indexing. In this approaches all features have to be mapped to binary features and an image feature is considered as a word in a text document. The main advantage of the discrete approach is that techniques from textual information retrieval can easily be transferred (e.g., storage handling and user interaction). The *continuous approach* is very similar to nearest neighbour classification. In fact, all the images are represented by a feature vector and these vectors are compared using distance-based evaluation measures. As the retrieval process result, the images ranked highest are those with lowest distances.

In the last few years, CBIR systems have been consider has an exciting application domains for several research fields in computer science. In [16] a Content-Based Image Retrieval technique based on genetic algorithms with support vector machines and user feedbacks for image retrieval is proposed using a web 3.0 architecture. Karamti *et al.* [17] present an image retrieval framework which includes a vectorization technique combined with a pseudo relevance model. *Geographic Information Retrieval* and its applications spread on different areas as urban planning, urban architecture, environmental protection, transport and logistic, engineering networks, real estate and military planning and so on. Google and Microsoft introduced commercial GIR system in 2005, respectively *Google Maps*[1] and *Live Search Maps* (since 2010 and presently *Bing Maps*[2])

[1] https://www.google.com/maps. Last seen May 7, 2018.
[2] https://www.bing.com/maps. Last seen May 7, 2018.

and these are currently the most used on-line GIR services. In 2007 a group of researchers of the University of California, Irvine, proposed a framework for GIR systems focusing on indexing strategies that can process spatial-keywords queries efficiently [18]. The paper shows through experiments that the indexing strategies lead to significant improvement in efficiency of answering spatial-keywords queries as regards to other techniques already present. A search engine called *TexSpaSearch* was presented at the University of New Brunswick, Fredericton, Canada [19]. It is a geo-textual index and search methodology that simultaneously supports text only, text with search radius, and point location with radius queries. In addition to other similar systems, TexSpaSearch supports simultaneous indexing of text having geometric figures describing their location. In [20] an automatic approach for discovering location names in WWW data selected from different domains is proposed. The described approach is based on *Apache Tika*, *Apache OpenNLP*, and *Apache Lucene* frameworks. A very interesting general-purpose geographic search engine was presented in [21]. This Geographic Information Retrieval system, called *Frankenplace*, is an interactive thematic map search engine that uses geographic context as a way to discover, organize, and interactively visualize documents related to a search query. Frankenplace is also an ad-hoc search site designed to help users in finding relevant documents that match a query[3], and visualizing the interaction between the thematic and geographic content of documents. The current version of Frankenplace (May 2015) indexes over five million articles from the English version of Wikipedia and online travel blog entries.

The increasing of computational power of microprocessors improves the research in CBIR field in terms of different directions, such as applications to art and cultural imaging. For centuries, our memory institutions (i.e. libraries, archives, museums) have spent their efforts on collecting and describing artefacts and social phenomena to preserve and give access to our cultural heritage. In this context, several questions related to digital libraries management and information storage and retrieval represent hard issues and exciting challenges.

Mobile applications represent a fast and smart solution to achieve information about cultural heritage. Smartify[4] and GetCOO Travel[5] are interesting examples. Smartify is a free app that allows to scan and identify artworks, access rich interpretation and build a personal art collections in some of best museums and galleries around the world. GetCOO Travel is a smartphone application developed for tourists. They can take pictures of monuments to receive information about it.

In this context, we propose a novel multimodal information retrieval approach used to improve the user satisfaction during his/her roaming for cultural heritage information. Our approach has been completely developed in a mobile system based on open source technologies. One of the improving of the proposed system is in the use of on board sensors as GPS device, in fact, if it is activated

[3] http://frankenplace.com/. Last seen May 7, 2018.
[4] https://smartify.org. Last seen May 7, 2018.
[5] https://www.travel.getcoo.com. Last seen May 7, 2018.

by the user during an artwork recognition, it allows to effectively exploit the combination between location of the considered artwork and the multimedia data provided from the image matching task. The location of an artwork, based on the geographic position of the user is used to filter the results obtained from a multimedia search, considering only the artworks sited in the place resulting from the geographic search. In the author opinion the use of spatial information is a basic feature to implement smart systems in different application contexts and to analyse complex scenarios [22,23].

In according to the use of mobile technologies for digital cultural heritage access, we implement an Android application and test our framework and system in a real use case scenario about museums in the Italian city of Napoli.

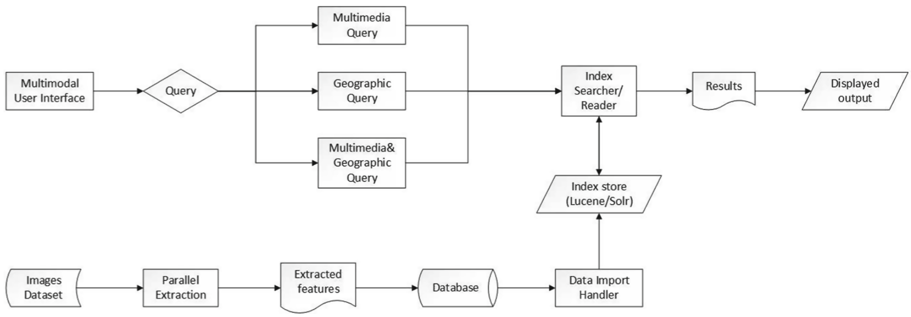


Fig. 1. System architecture

3 The Proposed Approach

The aim of our approach is to integrate Content-Based Image Retrieval and Geographic Information Retrieval to improve the accuracy of results in the retrieval process and give an easy user interface for multimodal query. The final goal is to have an enhancement of user satisfaction. Our approach has been implemented in a system and its architecture is shown in Fig. 1. In our discussion we introduce the used Content-Based Image Retrieval features starting from the description of the multimedia search implementation. In the second part the integration between multimedia and geographic data has been explained, focusing on the spatial data processing and the combination between the results obtained from CBIR and GIR. In the last part we'll present a *points of interest* (POI) *search.*

The system has been implemented using *Apache Solr* queries performed on a dataset composed of artworks located in Naples's museums. In the following sections, the three system main functionalities will be described.

3.1 Multimodal Query

The main aim of Content-Based Image Retrieval is to develop techniques for retrieving images on the basis of automatically-extracted features. For this reason the tasks of the *Multimedia Query* functionality provides to extract multimedia/visual features from an image previously loaded, and perform a search using the features vector obtained as result of the extraction. The *Geographic Query* functionality consists in points of interest search and, in our context they are represented by a museum, a church, an art gallery. In particular, for this functionality, the museums that are near to the user location are considered as points of interest. To perform this type of search, a radius has to be set to define the circle in which the museums can be retrieved. The idea of combining geographic data with multimedia data comes from observation of multimedia queries results. Most of the artworks submitted to the system was not retrieved in useful position and often the item in the top results are far from the user position. Therefore the use of a functionality based on the geolocation of the user could be useful to filter the results of a standard multimedia query. Obviously, filtering is possible only if the user geographic position is available at real-time by mobile devices.

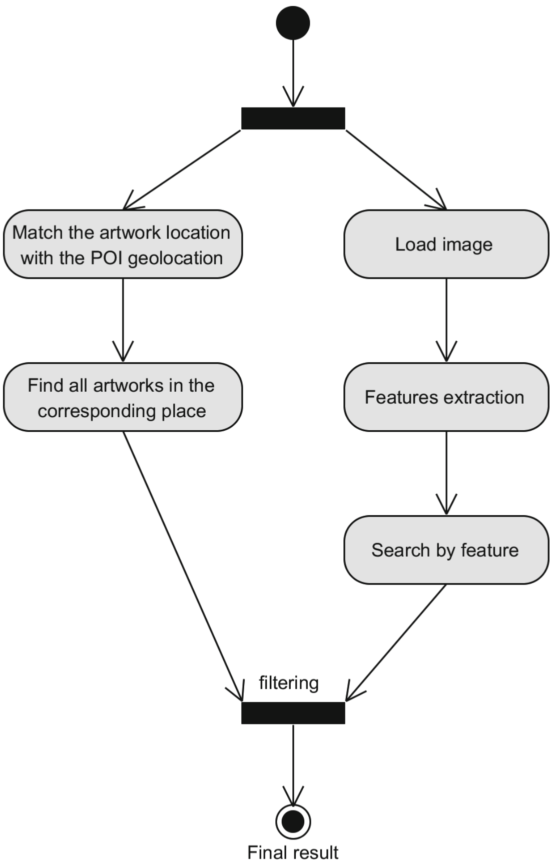


Fig. 2. Multimedia&Geographic query

The first step in the *Multimedia&Geographic Query* functionality is similar to the Multimedia Query and have the same purpose in features extraction from the loaded image and the searching by this feature vector. Afterward, a Geographic Query is performed to match the artwork location, based on the user geographic position, with the corresponding POI geolocation, to discover the place in which the user, and thus the artwork, is sited. The result of this query is used to find all the artworks in that place. Finally, the list of artworks is used in order to filter the multimedia results and obtain a more accurate result.

The Activity Diagram of this functionality is shown in Fig. 2. The artworks and their information are stored in a *Neo4J* instance. This database is used in the implementation and it is composed of the nodes drawn in Fig. 3: *Cultural Object*, *Author*, *Multimedia Representation* and *Place*.

It is important to point out that *location* is an exclusive attribute of *Place* node. Between this node and the *Cultural Object* one, there is relationship representing the current position of the artwork. This relationship is used to retrieve the POI in which the artwork is located, in the Multimedia&Geographic Query functionality, and has to be dynamically modified when an artwork changes location.

3.2 Multimodal Query Implementation

The *Multimedia Query* class is developed using the *SolrJ* library and it is composed of two queries provided by LIRe plug-in for Solr:

1. *Extract*: the URL of the selected (or captured) picture is passed to the query to extract the feature specified in the `field` parameter;
2. *Search by feature*: this query receives the results of the previous one as the `hashes` and `feature`. To manage easily the results of this query, the `doSearch` method of Lux's `LireRequestHandler` class has been modified: the type of the return value in the `SolrQueryResponse` has been changed from `LinkedList` to `SolrDocumentList`, that is, the default object type provided in SolrJ library for the queries response. In this way, the analysis of the results list from the response became immediate, being able to use the standard methods provided by the library.

The dataset used in the system includes Solr documents with the information of six Naples's museums: Museo nazionale di Capodimonte, Gallerie di palazzo Zevallos Stigliano, Museo nazionale di San Martino, Cappella Sansevero, Museo archeologico nazionale di Napoli, Museo civico di Castel Nuovo. Within these information, there is also the `location` field, expressed as a "latitude-longitude" string and configured in the `managed-schema` file as values of the `LatLonPointSpatialField` type. The *Geographic Query* method consists in a *points of interest* (POI) *search* to retrieve all the museums at a radial distance of one kilometer from the user. The Solr Spatial Query is performed using the `location` field of the Solr documents and the `geofilt` filter. As shown above, the *Multimedia&Geographic Query* functionality consists of the four steps listed below with the methods used in the implementation:

1. The first steps are equals to Multimedia Query and have the same purpose:
   (a) *Extract* query to get the feature vector from the captured image;
   (b) *Search by feature* query.
2. A Geographic Query is performed to match the artwork location, based on user's device GPS location, with the corresponding POI geolocation, to discover the place in which the user, and thus the artwork, is sited. For this query, the `geofilt` filter is used and a radial distance of 200 meters is set (`d=0.2`).
3. The result of the previous query is sent as a parameter of the join query that is performed to find all the artworks in that place.
4. The resulting list is used as the filter for the results of the *Search by feature* query performed at point 1(b).

The final result of these steps is the most similar artwork to the captured image that is located in the POI where the user is.

3.3 The Used Technologies

The technologies used to implement the functionalities previously described are now presented. These description is useful to understand the whole framework and put in evidence the complete open source vision of our work.

Apache Lucene is an open source Java-based search library providing Application Programming Interfaces (APIs) for performing common search and search related tasks like indexing, querying, highlighting, language analysis, and many others [24]. The main capabilities of Lucene are centered on the creation, maintenance and accessibility of the *Lucene inverted index*. In particular, it has implemented a modified vector space model that supports incremental index modifications. As regards querying, Lucene supports a variety of query types, including: fielded term with boost, wildcards, fuzzy (using Levensthein Distance), boolean operators (AND, NOT, OR) and proximity searches. Furthermore, the version 3.6.0 has added support for regular expressions, complex phrases, spatial distances and arbitrary scoring functions based on values in a field.

Apache Solr is an open source, enterprise-ready, and highly scalable search platform, from the Apache Lucene Project [25]. It is written in Java, runs as a stand-alone server and also provides a web based graphical administrator interface for simple monitoring. Solr is *config-based* and the main configuration files are `solrconfig.xml` and `managed-schema`. The second one of these two files primarily defines the fields of the schema and their behaviour, that is how the text must be tokenized while indexing and querying. There is also a possibility to use Solr schemaless and let to Solr itself create fields while indexing the data. Solr provides libraries in various languages for server connection, documents indexing and queries handling. The library used in the proposed system is *SolrJ*, written in Java. Furthermore, Solr supports geographic data for use in spatial/geospatial searches. The main characteristics of this kind of query are: indexing points or other shapes; filtering search results by a bounding box, circle, or other shapes; sorting or boost scoring by distance between points, or relative area between

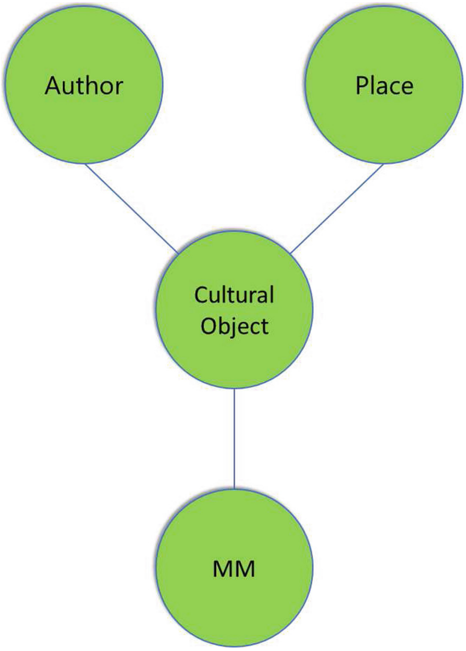


Fig. 3. Neo4J nodes

rectangles; generating a 2D grid of facet count numbers for heatmaps generation or point plotting.

`LatLonPointSpatialField` is the field type for most common use-cases for "latitude-longitude" point data, and for this reason it is the type used in our system. It replaced `LatLonType` which still exists for backwards compatibility. Two *Spatial Query Parsers* are provided for geospatial search: `geofilt` and `bbox`. The `geofilt` filter allows the user to retrieve results based on the geospatial distance, also know as "great circle distance", from a given point. This circular shape filter returns all results within a circle of the considered radius around the center point.

The `bbox` filter is very similar to the previous one. It takes the same parameters as `geofilt`, but it uses the bounding box of the calculated circle. The rectangular shape is faster to compute and for this it is often used when it is acceptable to have points outside the circle.

LIRe is a Java library for Content-Based Image Retrieval [26] used to extract image features from the dataset images and store them in a Lucene index for later retrieval. It has a very simple approach, in fact only a few lines of code are needed to integrate this library in any CBIR application. The integration of additional image features is made to further extend the functionality of LIRe.

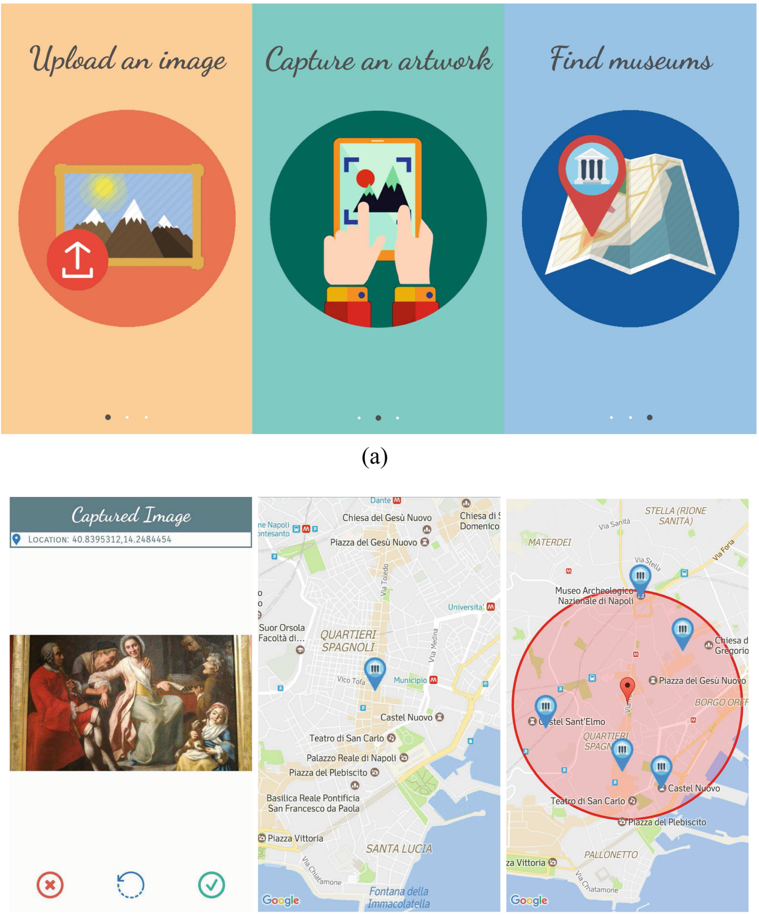
LIRe Request Handler is a Solr plug-in that provides a Content-Based Image Retrieval server, allowing the combination of a Apache Solr search server with the LIRe CBIR Java-based library [27]. The LIRe Solr plug-in includes a

`RequestHandler` which supports the different types of queries used in the system. *Extracting features* query extracts the histogram (Base64 encoded string) and the hashes from an image, selecting the feature with `field` parameter and the number of query terms should be in the queries with `accuracy` parameter. The return values are ordered by ascending document frequency values and distance from the image to the respective reference point. *Search by feature vector* query returns an image that looks like the one the given features were extracted. This method is used if the client extracts the features from the image. Mandatory parameters to set are: `field`-the feature to search for; `hashes`-string of white space separated numbers representing the hashes of the image feature; `feature`-the Base64 encoded feature histogram. This query is used to match the feature just extracted with the ones extracted from images stored in Solr indexes and retrieve the correct artwork and the information linked to it. Furthermore, to extract features from a group of images (i.e., the whole dataset) and to index it, a Java runnable class called `ParallelSolrIndexer` can be used. It creates XML documents to be sent to Solr server and indexed in Lucene, taking in input a text file containing the paths of each image. Unfortunately, this class provides the extraction of global features only. To bridge the gap, a new Java runnable class for the local feature parallel extraction has been implemented. Due to the large amount of possible data, we use a NoSQL database. For our purpose we choice to use Neo4J, an open source graph database management system developed in Java. It is based on the *property graph model*, in which a node is described with properties and labels. Neo4J is accessible from software written in other languages using the *Cypher Query Language* through a transactional HTTP endpoint. The database instance has been indexed directly into Solr with the help of Neo4J `JDBC Driver` and the Solr `Data Import Handler` (DIH) feature. The Data Import Handler provides a mechanism for importing content from a data store and indexing it. In addition to relational databases, DIH can also index content from NoSQL DB, HTTP based data sources such as RSS feeds, e-mail repositories, and structured XML with an XPath processor is used to generate fields[6]. The DIH has to be registered in `solrconfig.xml`, and the only required parameter is the `config`, which specifies the location of the handler configuration file that contains specifications for the data source, the way to fetch data, type of fetched data, and how to combine nodes and relations in the Neo4J database to create entities related to the desired structure and generate the Solr documents to be posted to the indexer.

4 A Case Study on Cultural Heritage Domain

The described system has been designed using a client-server paradigm and implemented in an Android application. We describe the whole process presenting both user interaction tasks and implementation details to better understand our application.

[6] https://cwiki.apache.org/confluence/display/solr/Uploading+Structured+Data+Store+Data+with+the+Data+Import+Handler. Last seen May 7, 2018.



(a)

(b) (c) (d)

Fig. 4. User interface example (Color figure online)

The client-side application allows users to: *upload* an image from device gallery to retrieve information about artwork represented in the given image; *take a picture* of an artwork in a POI to retrieve information about it. The user can enable his/her position given by GPS sensor to improve the accuracy of image retrieval; *discover* the location of the museums closest to him/her. The server-side system is able to: *receive* and *manage* data sent by client application to call the right method, based on data types received in the request; *establish* the connection with the `SolrClient`; *perform* the necessary `SolrQuery`; *get* results as `SolrDocument`; *send* the response to client. The connection between client and server is obtained via the HTTP protocol, and in particular by the POST request method realized by an `HttpURLConnection` Java object. JSON (JavaScript Object Notation) structures are used as *content type* of the POST requests and response; it represents the standard type of format for data interchange between a client and a server. The *media type* for JSON is `application/json`. To handle the JSON object received both on server (HTTP request) and on client (HTTP response), methods provided by *Google-Gson* library[7] are applied.

On server, the `fromJson()` method is used to convert the request into an object, having as attributes the three possible data types sent by client: `base64String`, the Base64-encoded string of the image uploaded or captured by user (`null` in case of Geographic Query); `type`-the string representing the type of uploaded image file or captured by user (`null` in case of Geographic Query); `userPosition`-the string representing the user's geographic location, in "latitude, longitude" format (`null` in case of Multimedia Query).

On client, instead, the `fromJson()` method is used to get the results from the response and store them in a Java class to be managed and displayed by the device. The results structure contains: `title`, `author`, `place` (the name of the POI in which the retrieved artwork is sited), `location` (the geographic location of the POI in which the retrieved artwork is sited, in "latitude, longitude" format), `placeUrl` (the *Wikipedia* page web address of the POI in which the retrieved artwork is sited), `base64image` (Base64 encoded string of the image of the retrieved artwork), and `imageUrl` (the *Wikipedia* page web address of the retrieved artwork, if available).

In Fig. 5 a Use Case Diagram representing a static and formal description of the application functionalities is shown.

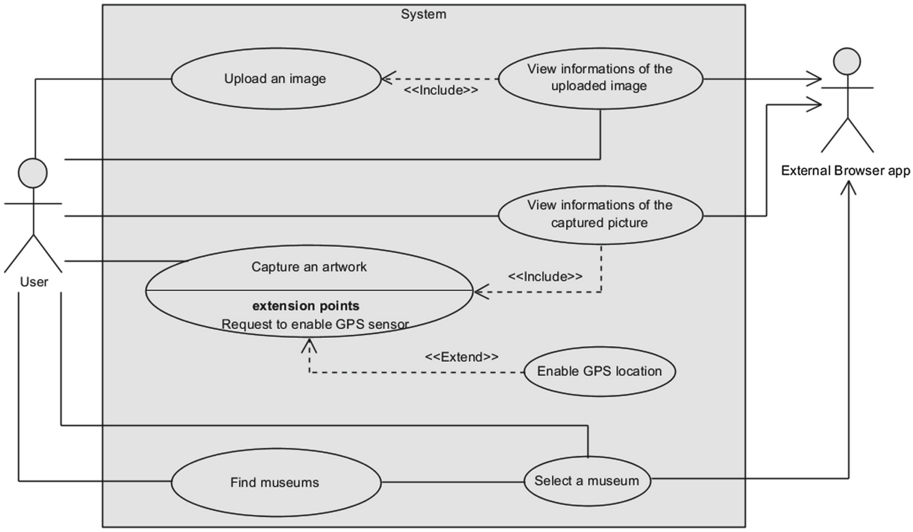


Fig. 5. Use case diagram

[7] https://github.com/google/gson. Last seen May 7, 2018.

On startup, the application checks if the internet connection is available, because this is a needful requirement for the work of the entire system. In the case that no connection is enabled, an *alert dialog* is shown warning the user to provide to turn it on. Enabled the internet connection, the user visualizes the home screen of each functionality, that allows him to click on the circular button placed at the center of the screen, Fig. 4(a). A picture representing an artwork can be submitted to the system both loading it from the device storage or taking it by device camera. In this second option, once the photo is taken, the application displays an *alert dialog* to the user asking him/her the possibility of using position, given by GPS sensor, to improve the image retrieval, allowing the filtering based on location. The image in Fig. 4(b) shows the case in which the user chooses to improve the image retrieval with position, and under the screen title a *location* field displays the geographic data, represented as "latitude, longitude". After the server side computation, a map with a marker is displayed to show the result as depicted in Fig. 4(c). The marker indicates the location of the retrieved artwork and, by clicking on it, an *info window* appears with the information about title, author, place and a web link regarding the artwork. The *link* field can contain a Wikipedia address, if the artwork has a dedicated web page, or a Google address that leads to a search query in which title and author of the artwork are used as keywords.

In Fig. 4(d) results of a points of interest search are shown. After the server computation, a map is displayed. On it a red marker indicates the user position and a red circle shows the area in which the POI search is performed. Within this circle one or more customized markers can be displayed if the retrieval produces some results. The blue markers are clickable and, doing this, an *info window* appears, one for each different result, reporting two fields in it: the place name and the Wikipedia web page link. This last fields contains the web address of the Wikipedia page of the selected museum and by clicking on it, the Browser app opens to show this page. In the case, one of the retrieved museums has not a dedicated Wikipedia web page, the link field of the info window would contain a Google address performing a search query using the place name as parameter.

5 Experimental Results

In this section, we present a qualitative evaluation of the proposed system to measure the user satisfaction.

In the experimental session a pool of 100 students with different skills without experiences with our application have been asked to perform some queries and use all functionalities. At the end of the experiments, each one has had to compile a questionnaire. The system has been evaluated using all the possible query types proposed by the system. The query and the used data set have been based on some cultural places and objects in the city of Naples. The questionnaire is structured following a methodology presented in [28] to evaluate system usability and user satisfaction. For each of the five questions selected, the user gives a score from *strongly disagree* (1) to *strongly agree* (7) (Y-axis), for the corresponding statement. The questions contain the following statements (X-axis):

1. Overall, I am satisfied with how easy it is to use this system;
2. It was simple to use this system;
3. I was able to complete the tasks quickly using this system;
4. It was easy to learn to use this system;
5. It was easy to find the information I needed.

The number of scale values (7) has been defined through a trade-off between reliability and easiness for the tester. Data from any questionnaire have been gathered and analyzed. Figure 6 show the mean of obtained results for each types of query (MQ = Multimedia Query, GQ = Geographic Query, M&GQ = Multimedia&Geographic Query).

The geographic query has slightly better values with questions 1, 2 and 4 which refer to simpleness and ease to learning. This could probably caused by the fact that the geographic query is quite similar to other widely information systems. However the strong difference is in questions 3 and 5, which are about completing the task and finding information where the combined query show an improvement. Thus, we can deduce that with a very small effort into adapting to the new way of querying, the users achieve better results.

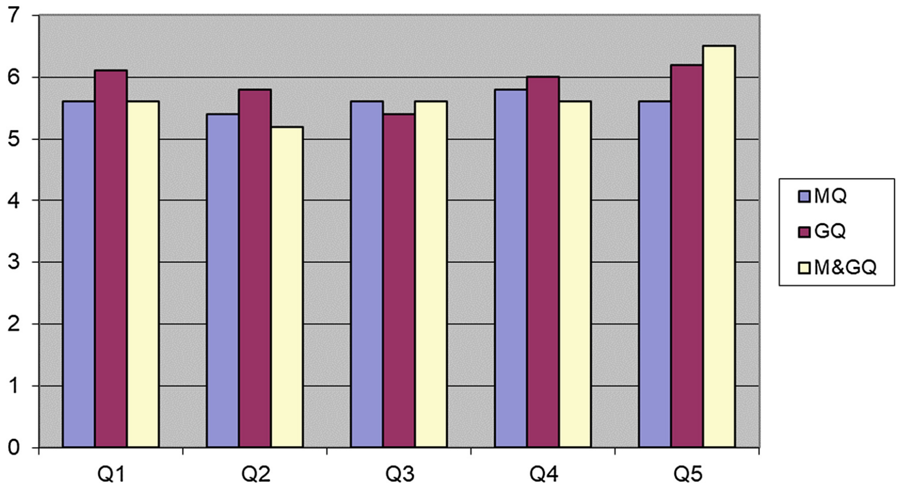


Fig. 6. Qualitative evaluation results

6 Conclusion and Future Work

In this paper a system providing a multimodal user interface for cultural heritage information retrieval has been presented. We present a complete design an implementation of our proposed approach using an Android application. It presents three functionalities: a method for the image matching based on visual descriptors, called *Multimedia Query*; a points of interest search called the

Geographic Query, and the *Multimedia&Geographic Query*, which provide the combination between geographic and multimedia data to improve the accuracy of image retrieval process. In particular, this functionality is based on the geolocation of the artwork, derived from the user geographic position who perform the query, which filter the results of a standard multimedia/visual query. The satisfaction of users has been measured using a formal methodology and our results suggest several directions for future investigations and researches. An in-depth study of the principal visual descriptors actually used in image retrieval should be analyzed in order to measure their performances and recognize the best features to use in the system implementation. Focusing on the multimodal user interface, instead, it could be expanded with other functionalities that provide, for example, the possibility to the posing query process, a vocal query could be implemented using techniques and methods for speech recognition. Furthermore, semantic web oriented techniques can be investigated and eventually exploited in order to enhance the query construction also adopting matchmaking strategies [29,30] using cultural heritage resources collected from heterogeneous databases. In addition, some improvements could be done on the extension of the image dataset to include other artworks and different cultural features. Eventually, a more specific quantitative experiments using precision and recall measures should be carried out to compare our strategy with other presented in literature.

References

1. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: ideas, influences, and trends of the new age. ACM Comput. Surv. (CSUR) **40**(2), 5 (2008)
2. Chen, C., Wactlar, H.D., Wang, J.Z., Kiernan, K.: Digital imagery for significant cultural and historical materials. Int. J. Digit. Libr. **5**(4), 275–286 (2005)
3. Caldarola, E.G., Rinaldi, A.M.: Big data: a survey - the new paradigms, methodologies and tools. In: Proceedings of 4th International Conference on Data Management Technologies and Applications, pp. 362–370 (2015)
4. Caldarola, E.G., Picariello, A., Rinaldi, A.M.: Big graph-based data visualization experiences: the WordNet case study. In: 2015 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K), vol. 1, pp. 104–115. IEEE (2015)
5. Caldarola, E.G., Rinaldi, A.M.: Improving the visualization of WordNet large lexical database through semantic tag clouds. In: 2016 IEEE International Congress on Big Data (BigData Congress), pp. 34–41. IEEE (2016)
6. Caldarola, E.G., Picariello, A., Rinaldi, A.M.: Experiences in WordNet visualization with labeled graph databases. In: Fred, A., Dietz, J.L.G., Aveiro, D., Liu, K., Filipe, J. (eds.) IC3K 2015. CCIS, vol. 631, pp. 80–99. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-52758-1_6
7. Shete, D.S., Chavan, M., Kolhapur, K.: Content based image retrieval. Int. J. Emerg. Technol. Adv. Eng. **2**(9), 85–90 (2012)
8. Deselaers, T., Keysers, D., Ney, H.: Features for image retrieval: an experimental comparison. Inf. Retr. **11**(2), 77–107 (2008)

9. Larson, R.R.: Geographic information retrieval and spatial browsing. Geographic information systems and libraries: patrons, maps, and spatial information. Papers Presented at the 1995 Clinic on Library Applications of Data Processing, 10–12 April 1995 (1996)
10. Manning, C.D., Raghavan, P., Schütze, H., et al.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
11. Mark, D.M.: Geographic information science: defining the field. Found. Geograph. Inf. Sci. **1**, 3–18 (2003)
12. Moscato, V., Picariello, A., Rinaldi, A.M.: A recommendation strategy based on user behavior in digital ecosystems. In: Proceedings of the International Conference on Management of Emergent Digital EcoSystems, pp. 25–32. ACM (2010)
13. Smeulders, A.W., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE Trans. Pattern Anal. Mach. Intell. **22**(12), 1349–1380 (2000)
14. Frederix, G., Caenen, G., Pauwels, E.J.: Panoramic, adaptive and reconfigurable interface for similarity search. In: Proceedings of the 2000 International Conference on Image Processing, vol. 3, pp. 222–225. IEEE (2000)
15. Giuca, A.M., Seitz, K.A., Furst, J., Raicu, D.: Expanding diagnostically labeled datasets using content-based image retrieval. In: 2012 19th IEEE International Conference on Image Processing (ICIP), pp. 2397–2400. IEEE (2012)
16. Irtaza, A., Jaffar, M.A., Muhammad, M.S.: Content based image retrieval in a web 3.0 environment. Multimed. Tools Appl. **74**(14), 5055–5072 (2015)
17. Karamti, H., Tmar, M., Visani, M., Urruty, T., Gargouri, F.: Vector space model adaptation and pseudo relevance feedback for content-based image retrieval. Multimed. Tools Appl. **77**, 1–27 (2017)
18. Hariharan, R., Hore, B., Li, C., Mehrotra, S.: Processing spatial-keyword (SK) queries in geographic information retrieval (GIR) systems. In: 19th International Conference on Scientific and Statistical Database Management, SSBDM 2007, p. 16. IEEE (2007)
19. Han, A., Nickerson, B.G.: Efficient combined text and spatial search. In: Gervasi, O., Murgante, B., Misra, S., Gavrilova, M.L., Rocha, A.M.A.C., Torre, C., Taniar, D., Apduhan, B.O. (eds.) ICCSA 2015. LNCS, vol. 9157, pp. 713–728. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-21470-2_52
20. Mattmann, C.A., Sharan, M.: An automatic approach for discovering and geocoding locations in domain-specific web data (application paper). In: 2016 IEEE 17th International Conference on Information Reuse and Integration (IRI), pp. 87–93. IEEE (2016)
21. Adams, B., McKenzie, G., Gahegan, M.: Frankenplace: interactive thematic mapping for ad hoc exploratory search. In: Proceedings of the 24th International Conference on World Wide Web, pp. 12–22. ACM (2015)
22. Di Pinto, V., Rinaldi, A.: A configurational approach based on geographic information systems to support decision-making process in real estate domain. Soft Comput. 1–10 (2018). https://link.springer.com/article/10.1007/s00500-018-3142-9#citeas
23. Rinaldi, A.: A GIS-based system for electromagnetic risk management in urban areas. J. Locat. Based Serv. **3**(1), 3–23 (2009)
24. Białecki, A., Muir, R., Ingersoll, G., Imagination, L.: Apache lucene 4. In: SIGIR 2012 Workshop on Open Source Information Retrieval, p. 17 (2012)
25. Shahi, D.: Apache Solr: an introduction. In: Shahi, D. (ed.) Apache Solr, pp. 1–9. Springer, Heidelberg (2015). https://doi.org/10.1007/978-1-4842-1070-3_1

26. Lux, M., Chatzichristofis, S.A.: LIRE: lucene image retrieval: an extensible Java CBIR library. In: Proceedings of the 16th ACM International Conference on Multimedia, pp. 1085–1088. ACM (2008)
27. Lux, M., Macstravic, G.: The LIRE request handler: a solr plug-in for large scale content based image retrieval. In: Gurrin, C., Hopfgartner, F., Hurst, W., Johansen, H., Lee, H., O'Connor, N. (eds.) MMM 2014. LNCS, vol. 8326, pp. 374–377. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-04117-9_39
28. Lewis, J.R.: IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. Int. J. Hum.-Comput. Interact. **7**(1), 57–78 (1995)
29. Caldarola, E.G., Rinaldi, A.M.: An approach to ontology integration for ontology reuse. In: 2016 IEEE 17th International Conference on Information Reuse and Integration (IRI), pp. 384–393. IEEE (2016)
30. Caldarola, E.G., Rinaldi, A.M.: A multi-strategy approach for ontology reuse through matching and integration techniques. In: Rubin, S.H., Bouabana-Tebibel, T. (eds.) FMI/IRI 2016 -2016. AISC, vol. 561, pp. 63–90. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-56157-8_4