Multimed Tools Appl
DOI 10.1007/s11042-011-0745-x

PhotoGeo: a photo digital library with spatial-temporal support and self-annotation

Hugo Feitosa de Figueirêdo · Yuri Almeida Lacerda · Anselmo Cardoso de Paiva · Marco Antonio Casanova · Cláudio de Souza Baptista

© Springer Science+Business Media, LLC 2011

Abstract The recent popularity of digital cameras has posed a new problem: how to efficiently store and retrieve the very large number of digital photos captured and chaotically stored in multiple locations without any annotation. This paper proposes an infrastructure, called PhotoGeo, which aims at helping users with the people photo annotation, event photo annotation, storage and retrieval of personal digital photos. To achieve the desired objective, PhotoGeo uses new algorithms that make it possible to annotate photos with the key metadata to facilitate their retrieval, such as: the people who were shown in the photo (*who*); where it was captured (*where*); the date and time of capture (*when*); and the event that was captured. The paper concludes with a detailed evaluation of these algorithms.

Keywords Personal photo collections · Geo-referenced photos · Automatic photo organization · Metadata · Geo-tags

H. F. de Figueirêdo (✉) · Y. A. Lacerda · C. de Souza Baptista
Computer Science Department, University of Campina Grande,
Av. Aprígio Veloso, 882, Bodocongó - Campina Grande, Paraíba 58109-900, Brazil
e-mail: hugoff@dsc.ufcg.edu.br

Y. A. Lacerda
e-mail: yuri@dsc.ufcg.edu.br

C. de Souza Baptista
e-mail: baptista@dsc.ufcg.edu.br

A. C. de Paiva
Applied Computing Group NCA, Federal University of Maranhão,
Av. dos Portugueses, SN - São Luís, Maranhão 58109-900, Brazil
e-mail: paiva@deinf.ufma.br

M. A. Casanova
Pontifical Catholic University of Rio de Janeiro,
Av. Marques de São Vicente, 225 RDC CEP 22451-900 - Gávea - Rio de Janeiro, Rio de Janeiro, Brazil
e-mail: casanova@inf.puc-rio.br

Published online: 01 February 2011

1 Introduction

Digital photos are no longer limited to digital cameras. Indeed, most smartphones, PDAs and cell phones can also capture photos. These ubiquitous devices enable new opportunities to capture media, because they are always on and available. Besides, they are easily programmable, are connected to the Internet and are aware of context [1]. As a result, the number of personal digital photos is growing fast.

This growth has given rise to a costly activity for the holders of large collections of digital photos as regards the life cycle of a photo [7]—its capture, upload, storage, organization, sharing, annotation and retrieval. In fact, solutions have been proposed to help users in one or more of these activities. Several tools have been devised both in the academia and in the industry. Examples from the industry include Picasa (Google), Flickr (Yahoo!) and iPhoto (Apple), and from the academia, PhotoCompas [13], PhotoMap [18], MMM [4], and WWMX [17].

Research works have shown that the main photo attributes that help people remember photos are: *who*, *where* and *when*, in that order [12]. These three attributes do in fact define an event, such as “FIFA World Cup in 2010 in South Africa” or, more abstractly, “a group of people gathered at a given place and time interval with the same goal”. Throughout the paper, we will refer to the process of detecting the event in which a photo was captured as *event photo annotation*, and to the process of annotating a photo with a list of people who are shown in it as *people photo annotation*.

Metadata play a crucial role in photo digital libraries. Examples of such metadata include: author, date, place, subject, people involved, camera make and model, resolution, format, focus, exposition, and aperture. They can be used at different levels: semantic, syntactic and structural, in order to annotate the underlying photos [2].

Furthermore, photo annotation may be automatic, semi-automatic, and manual. Since users would obviously like to have photos annotated with less effort, the more automatic annotation is carried out, the more satisfied users will be. In the automatic method, feature matching techniques are used to gather the desired information. If the data are set in standard format (e.g. MIME, SGML, XML) then it will be possible to implement automatic extraction. Conversely, in the manual method, a domain-expert user may introduce all the semantic information on to the system. For instance, when users look at an image, they can describe the semantic information they can recognize. Lastly, the semi-automatic method consists of a combination of automatic and manual methods.

This paper proposes a photo digital library, called PhotoGeo, which aims at helping users with the people photo annotation, event photo annotation, storage and retrieval of personal digital photos. The main contribution of Photogeo is the use of new algorithms which will enable the user to annotate photos with key metadata attributes so as to facilitate photo retrieval. These attributes are the following: the people shown in the photo (*who*), the place where it was captured (*where*), the date and time of capture (*when*), and the event that was captured.

The remainder of this paper is organized as follows. Section 2 discusses related work. Section 3 presents the system architecture proposed. Section 4 focuses on an algorithm for people photo annotation. Section 5 addresses an algorithm for event photo annotation. Section 6 highlights the PhotoGeo retrieval and storage components. Section 7 presents the results obtained in experiments. Finally, Section 8 concludes the paper and suggests future work.

2 Related work

Rodden and Wood [16] have analyzed the way people manage their digital photos and argued that the main features of a photo organizer include self-organization and the use of thumbnails. Thus, the development of algorithms for photo collection self organization, and the development of tools for helping the interactive implementation of this task in a semi automatic way are highly desirable facilities.

One of the problems that have been addressed in this direction is the automatic detection of events. The most often proposed approaches make use of visual, temporal, and spatial information. The visual approach is based on the photo content i.e., the visual features extracted from it by using content-based information retrieval techniques. Photos captured at the same event usually contain similar visual features, such as: color, texture, shape and structure. For instance, photos taken during a walk on the beach on a sunny day may have significant visual similarities.

The approach based on temporal metadata uses the date and time extracted from the photo metadata file format (such as the EXIF [10] used in most digital cameras). This approach is based on the evidence that photos are taken sequentially during a given event. For instance, during a birthday party, people will take photos at different times of the party. This approach can be found in a number of research works, such as those by Cooper et al. [3]. Moreover, there are research works which combine visual and temporal techniques, such as that by O'Hare et al. [14].

Another approach to automatic detection of events employs both spatial and temporal metadata, and are based on the fact that a given event usually happens *somewhere* at a *certain time*. Thus, an event may contain photos of nearby locations. An example of this approach is PhotoCompas [12].

WWMX (World Wide Media eXchange) [17] is a pioneering photo digital library system that uses spatio-temporal information to index the photos. WWMX has a multi-modal interface which allows the user to navigate the photos not only through a list of photos, but also by means of a digital map. Moreover, this system has a search engine that allows the retrieval of photos based on their geographic locations. Photos are displayed on maps from media dots, which are groups of photos which are spatially close to one another. The search engine of WWMX is very simple and does not allow the user to formulate spatio-temporal queries, or search for people in photos. Furthermore, WWMX does not have a scheme for facilitating photo annotation; besides, it does not perform automatic event detection.

PhotoCompas [12] detects events by means of an algorithm that deals with spatio-temporal metadata. Moreover, you can automatically name events from place names and temporal information. Another feature of this system is that it provides a list of people with high probability of being in a given photo so as to render person annotation of photos easier [13]. The PhotoCompas also allows hierarchical navigation on photos through text. From this hierarchical navigation it is possible to perform the filtering of photos.

PhotoMap [18] performs automatic annotation by using the spatial, temporal and social context of a photo. Its metadata attributes include weather conditions and nearby user's friends and important objects. PhotoMap provides a mechanism to organize, share and retrieve photos. PhotoMap uses bluetooth technology to identify through their mobile phones which people are getting spatially closer at the moment of taking a particular photo. This system also has a spatio-temporal interface for viewing the photos. The context is expanded by connecting it to the Web services that capture context information (weather, temperature, light, etc.). PhotoMap does not automatically organize the photos into events. It does not have a mechanism to query the metadata of photos.

iPhoto[1] automatically detects events by using only temporal information. Picasa[2] does not support the automatic detection of events. Flickr[3] enables the user to organize her photos based on spatial metadata and tags. However, there is no automatic organization of photos. iPhoto, Picasa and Flickr enable the annotation and retrieval of photos, but their solutions are still very limited.

Based on related work presented in this section, the following features have been considered applicable to a photo digital library:

- Automatic detection of the event on which the photo was taken—It is essential that the system provides a mechanism to automatically annotate the event on which a particular photo was taken, by using information from metadata, such as temporal and spatial information, or information on photo content (derived from content based image retrieval techniques). A photo digital library is expected to have algorithms to separate the photo from the user at events automatically, semi-automatically or manually. On the automatic mode, the system separates the photos into events without the intervention of the user. On the semiautomatic mode, the user supplies some parameters to help detect the events. In the manual mode, however, the user separates the photos into events manually. The following research works have such features: O'Hare et al., Cooper et al., PhotoCompas, Picasa and Flickr;
- Multimodal navigation—Another essential feature of a photo digital library is the one that offers users a number of ways to navigate through the photos. For example, the user can navigate by means of a digital map, or through time hierarchy. Cooper et al., WWMX, PhotoCompas, Picasa and Flickr have this aspect;
- Mechanism to assist photo annotation—The annotation is essential to make the retrieval of photos easy. However, the user is not inclined to make any unnecessary effort to annotate the photos; so it is important that the system has mechanisms that can assist or automate the photo annotation process. A photo digital library must provide the user with various annotation facilities. There is information which people would like to remember through photos, such as the people shown in the photos (who), geographic locations (where), and dates (when). O'Hare et al., PhotoMap, PhotoCompas, Picasa and Flickr cater for this;
- Textual Queries—One essential aspect of a photo digital library is to allow the user to carry out text queries in order to retrieve images. There is much photo metadata information that can be used as search parameters. This aspect can be found in the following research works: The O'Hare et al., WWMX and Flickr;
- Temporal Queries—Temporal information constitutes a very important metadata of the photo. For this reason, a photo digital library should provide a mechanism by means of which the user can put queries on temporal information of the photos therein. A temporal query interface must provide users with the basic topological operators: before, after, equals, meets, during, overlaps, starts and ends. For example, the system should make possible to tackle questions like: "Retrieve all the photographs taken between 2000 and 2001." The following research works do have these features: O'Hare et al. and Flickr;
- Spatial Queries—A photo digital library may possess a spatial search engine, this is, capable of working out spatial operations when filtering photos. O'Hare et al. contemplate such feature; and

[1] http://www.apple.com/br/ilife/iphoto
[2] http://picasa.google.com.br
[3] http://www.flickr.com

- Web Repository—It is important that the system makes available a web repository to enable the user to store her photos. Two factors prompt the system to develop such a repository: data connectivity and data security. Connectivity has been motivated by the ever increasing development and availability of the Internet. We can also add to that a number of Web connecting devices. This enables users to access their photos wherever they are. It is very common for Web servers to have software and hardware which are less prone to failure, to prevent data loss. These following devices do have such mechanisms: WWMX, Picasa and Flickr.

PhotoGeo includes all the above mentioned features. These features are considered important for a photo digital library. To the best of our knowledge, PhotoGeo is the only recommended tool that includes all these features.

3 The PhotoGeo architecture

The PhotoGeo architecture is presented in this section (see Fig. 1). PhotoGeo was designed on a distributed multi-tier architecture compliant with the Model-View-Controller (MVC) design pattern so that presentation, control and business model are in different tiers. There

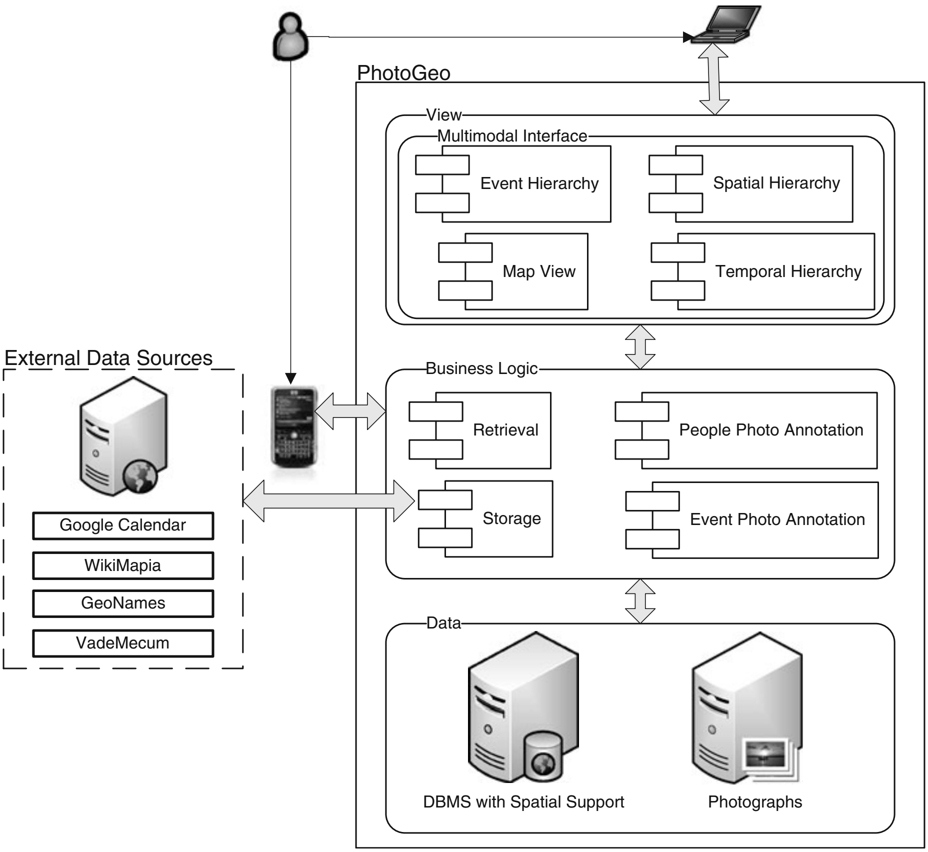


Fig. 1 The PhotoGeo architecture

are two types of PhotoGeo clients: the mobile client and the Web client. The mobile client is responsible for capturing pictures on mobile devices, and annotate them by including the people shown in the photo and the geographic location where the photo was taken, after which the photo will be sent to the server. The Web client is responsible for management functionalities and visualization.

The view tier uses a Web browser to generate content from JSP and HTML. The view tier has a multimodal interface to enable the user to navigate along the system by means of a map view, an event, or spatial or temporal hierarchies.

The PhotoGeo was designed to help the user to retrieve photos by means of the information about *who*, *when* and *where* of the photos; and annotate photos with these information. Thus, the modules of PhotoGeo aim to fulfill these features.

The PhotoGeo business logic tier is divided into four modules (Fig. 1), namely: event photo annotation module, people photo annotation module, retrieval module and storage module. The event photo annotation module uses an algorithm to identify events associated with the photos. The event photo annotation module is responsible for annotating the information about *where* and *when* of the photo. The people photo annotation module creates a list of the people most likely to be in a particular photo from which the user selects those shown in the photo. The people photo annotation module is responsible for annotating the information about *who* were shown in the photo. The retrieval module allows the user to retrieve photos by using spatial and temporal dimensions, the people shown in photo and the conventional metadata attributes. Finally, the storage module is responsible for obtaining data from external sources and from photo metadata to store in a data tier. The data from external sources are used in the PhotoGeo algorithms.

The storage module is divided into two components: spatial and temporal data collectors. The temporal data collector is responsible for acquiring information related to user appointments. Online Web calendars are largely used nowadays. There are very complex calendars with a rich set of functions which can be accessed from anywhere and shared with other people. The iCalendar standard is implemented by several of such calendars, as for example the Google Calendar. PhotoGeo accesses the Google Calendar[4] through a URL that the user provides to generate a file using the iCalendar standard. PhotoGeo extracts information of the user schedule at the time the photo was taken and stored in DBMS. The spatial data collector obtains geographic information from a collaborative mapping tool—Wikimapia[5]—, from a gazetter—GeoNames[6]—and from a provider of user's friends geographic location—VadeMecum system [6].

Wikimapia is a collaborative mapping tool where users share points of interest anywhere in the world. PhotoGeo extracts information from the places where the user's photos were taken in the Wikimapia. PhotoGeo obtains such metadata from a Web service,[7] which generates a KML file with the spatial objects that can be found within a given area of interest.

In order to identify place names and types more precisely, the utilization of a gazetteer is indispensable. Photogeo uses the Geonames gazetteer. GeoNames is a gazetteer that links a place name to both location and type [8]. With the help of this gazetteer, PhotoGeo may annotate the place name with more details, such as the name of a building, a mountain, a road, a street, a city, a state or a country. In fact, Geonames is like a dictionary for place

[4] http://www.google.com/calendar
[5] http://wikimapia.org/
[6] http://www.geonames.org
[7] http://wikimapia.org/api/

names, so users do not need to be concerned about latitude and longitude coordinates. PhotoGeo extracts GeoNames information from a Web service which receives a photo location and returns place names associated to that location.

The spatial data collector obtains user's friends geographic location from VadeMecum system [6]. This information is requested through a Web service using a SPARQL query. VadeMecum is a context aware system for mobile devices that provides contextual information about a particular person. The PhotoGeo inquires about the user's friends geographic location at the time the photo was taken. The obtained information is used in the algorithm that lists the people most probable of been on the photo. There are other applications that can retrieve the user's friends geographic location, such as Google Latitude[8]; however, the VadeMecum system was chosen for its based-protocol communication HTTP Get, and SPARQL to retrieve the user's friends geographic location.

The data tier is made up of a database server with spatial support and user's photo files. In the Section 6, the data tier operation and implementation is described in detail.

PhotoGeo also allows users to manually annotate photos. For example, the user can edit a metadata attribute of a particular photo. If the photos are not georeferenced, PhotoGeo also has a tool that allows users to georeference them. People photo annotation, event photo annotation and retrieval modules will be discussed further in the following sections.

4 People photo annotation

The process of people photo annotation can become a very tedious process if carried out manually. Several mechanisms have been proposed to help the user to annotate photos with lists of people. The most common techniques are the ones that employ facial recognition algorithms and context information, or a combination of both.

The facial recognition methods are imprecise and require a huge processing capacity when applied to photo albums [15]. Therefore, it is still not viable to deploy them in mobile devices with restricted processing capacity. Nevertheless, the techniques based on context information are quite promising.

4.1 Formalization of the problem

Let F be a set of n photos, which contains several photo albums of a certain user, and P be a set of people (identified by their names) who appear in the photos. For each photo f, let $p_f \subseteq P$ be the set of people who indeed occur in f.

Given a photo f, a people photo annotation process creates a set $H_f \subseteq P$ of people suggested to be in a photo f. A *perfect system* is such that $H_f = p_f$. Let $i \in P$. We say that a *hit occurs for i* iff $i \in H_f \cap p_f$, and that a *miss occurs for i* iff $i \in p_f - H_f$. We say that the system has a *hit* for photo f iff $H_f \cap p_f \neq \emptyset$, that is, at least one person from H_f is correctly associated with f; otherwise, we say that the system has a *miss*, that is, the system did not suggest anyone who actually is in the photo.

Let K be the set of people already identified by the system. In order to generate the set H_f, the people photo annotation system may adopt various estimators, so it can evaluate the

[8] www.google.com.br/latitude

probability that a person in K is in f. The candidates with higher probability are included in the set H_f, which then filtered by spatial proximity and presented to the user.

4.2 Estimators

A people photo annotation system may explore certain patterns that the photos in a given set follow. For example, if we consider a photo album, probably some people will appear in the photos more frequently than others, such as the photo album owner, her family and her closest friends. This is an example of a recurrence pattern, which implies that some people will have higher probability of appearing on photos than others. Other patterns can be identified by examining the photos, as described below.

In general, a people photo annotation system tries to generate an estimative for the probability that a certain person will appears in a photo. For this purpose, we extend the work on estimators proposed by Naaman et al. [12]:

- **[GR] Global Recurrence:** Percentage of how many times a person was annotated in the photos. If a person has a high recurrence value, it means that she has a higher probability of being in another photo. The global recurrence of a person $i \in K$ can be defined by using the Eq. 1. To calculate the global recurrence, X is the set with n_x photos from the set F.;

$$GR_i = \frac{\sum_{f \in X} g_f(i)}{n_x}, \text{ with } g_f(i) = \begin{cases} 1 \; if \; i \in \; p_f \\ 0 \; otherwise \end{cases} \tag{1}$$

- **[SR] Spatial Recurrence:** Percentage of how many times a person appears in the photos that belong to a certain spatial region. This estimator can be computed, for a person $i \in K$, through the Eq. 1, where X is the set with n_x photos from the set $C \subseteq F$, such that C represents the photos which were taken in a certain city;
- **[TR] Temporal Recurrence:** Percentage of how many times a person appears in the photos that belong to a time interval t, from the moment of the photo capture. To calculate the value for a person $i \in K$, the Eq. 1 is also employed, where X is the set with n_x photos from $T \subseteq F$, such that, T are those that belong to the time interval t of the photo f;
- **[ER] Event Recurrence:** Percentage of how many times a person appeared in the photos of a certain event. The Eq. 1 is also used to calculate the value for a person $i \in K$, where X is the set with n_x photos from the set $E \subseteq F$, such that, E are those that belong to the same event as the photo f;
- **[CO] Co-Occurrence:** Percentage of how many times two people appear together in the same photo. For instance, John is dating Mary, and they, frequently, appear together in the photos; thus, if John appears in another photo, there is a high probability of Mary appearing in that photo as well. The co-occurrence of two persons in a photo can be defined as the number of times that both of them appear in the same photo divided by the number of photos that only the first person appears. The Eq. 2 is used to calculate the CO value for a person $i \in K$;

$$CO_{i_1} = \max_{i_2 \in p_f} \left(\frac{\sum_{f \in F} h_f(i_1, i_2)}{\sum_{f \in F} g_f(i_1)} \right), \text{ with } h_f(i_1, i_2) = \begin{cases} 1 \; if \; i_1, i_2 \in \; p_f \\ 0 \; otherwise \end{cases} \text{ and } g_f(i) = \begin{cases} 1 \; if \; i \in \; p_f \\ 0 \; otherwise \end{cases} \tag{2}$$

- **[SP] Spatial Proximity:** This verifies, at the shot, how spatially close the digital camera is to the people registered in the system. For instance, two friends were near the digital camera when a photo was captured and the others were a hundred meters away. Thus, there is a higher chance that those friends near the camera are in the photo, while the others certainly are not; and
- **[PE] Presence in Event:** It verifies if a person was in the same event as the user at the moment of the photo capture. For example, the user might have inserted in her calendar information about an event, including who will attend the event. When the user takes the photos, the system matches the date of the photos with the time intervals of the events annotated in the user's calendar to estimate who appeared in the photos.

4.3 People photo annotation algorithms

The people photo annotation module of PhotoGeo adopts two methods based on context information captured in mobile devices which have embedded digital cameras. The first method, called *connected*, is based on a client-server model, where the mobile device sends context information of the captured photo to a server in order to get a list with suggested people. The other solution, called *disconnected*, does not need a connection with the server, so it solves the problem of suggesting people for photo annotation by using only the resources available in the mobile device itself. There are no differences between the interfaces of the approaches to the user.

Figure 2 shows the PhotoGeo mobile client interface for the people photo annotation. After capturing a photo, the application generates a list of suggestions, and displays the menu from which the user can select the people to be annotated on the photo metadata.

4.3.1 Connected method

In a scenario that uses the connected method, as soon as the user captures a photo by using her mobile device, she will be asked if she wants to annotate the taken photo. If she accepts it, the system will send the context information (date/time, geographical position) of the photo to a PhotoGeo server that will process the context information and will return a list of people ranked by decreasing probability of appearing in the photo.

The people photo annotation module is responsible for assigning weights to each person who has been already annotated in the system. The suggestion algorithm is composed of a series of estimators that assign weights. After the execution of the people photo annotation

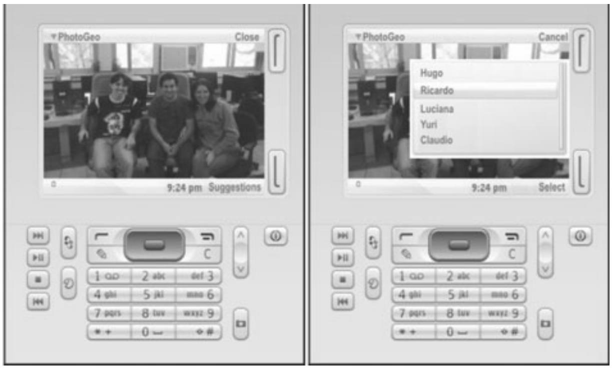


Fig. 2 PhotoGeo mobile client interface for people photo annotation

suggestion algorithm, the suggestion list will be sent to the user. Hence, she can decide which suggestions will be accepted. If none of the suggested annotations are useful, the user can manually annotate the photo.

The main advantage of the connected method is the possibility of employing a more complex algorithm to generate the suggestion list, since the server processing capacity is higher than that of a mobile device. A disadvantage of this solution is that it depends on an Internet connection established with the server, which is not always available.

4.3.2 Disconnected method

In the disconnected method, the people photo annotation suggestion algorithm is executed in the mobile device itself. However, this kind of equipment, despite its technological advancements, has limited storage and processing capability. Thus, a suggestion algorithm, suitable to mobile devices, was implemented, employing simpler versions of the GR, SR, TR, ER and SP estimators. The CO and PE estimators are not employed in the disconnected method. Indeed, the CO estimator demands a huge storage capacity, given that, for each person existing in the contact list, it would be necessary to generate a list containing her co-occurrence with all the other people. The PE estimator must access the Internet to retrieve information about the user's Web calendar.

The following paragraphs will detail the implementation of each estimator, including the differences between the connected and disconnected methods. The notation used to indicate the estimators will implement the pattern "disconnected + estimator name", where the abbreviations for the disconnected estimators start with the letter "D".

The disconnected global recurrence estimator (DGR) was implemented as follow: there is a counter A that stores the number of annotations already performed in the system and another counter $B(i)$, which is responsible for counting the number of times each user i has already annotated. These counters are stored internally in the device. For each user i, the global recurrence value is computed as $A/B(i)$. The difference between the DGR and the GR estimators lies in the way the counters are stored: GR computes the estimator value in real time, using annotations stored in the server; whereas DGR stores a counter that contains information about the number of annotations for each user and another counter with all annotations performed. Thus, there is no difference between the computations of both approaches.

The disconnected spatial recurrence estimator (DSR) employs information about cell phone towers (cell-ids). The ideal solution would be to make use of latitude and longitude, which is used in SR estimator. However, spatial operations demand more sophisticated computations. Therefore, the approach applied was simpler: it was accomplished just by mapping the cell-ids to the user's friends which had already been annotated in the photos captured inside the area covered by each cell-id. The solution stores a counter $C(i,c)$ that relates the number of each user i annotations with a cell-id c and another counter D, which stores the number of all cell-id annotations. For each user i, the spatial recurrence of a cell-id c is computed as $C(i,c)/D$.

The disconnected temporal recurrence (DTR) was implemented through a set Z that contains the last N distinct annotations, as well as their insertion orders. The following situations can occur when a new annotation is inserted into the system: (i) if the annotation belongs to the set, then its position will be updated to the most recent; (ii) if the annotation does not belong to the set and there is a number of annotations less or equal to N, then this new annotation will be inserted with its insertion order; and (iii) if the annotation does not belong to the set, and there is more than N annotations in the system, the earliest annotation will be removed, and the new annotation will be inserted in the set with its insertion order. For each contact, the DTR value will be 1 if she belongs to *Z*, and 0 if otherwise. The TR estimator uses all system annotations, which are performed in photos captured during a time interval *t* before the time of the actual photo capture.

The disconnected event recurrence estimator (DER) uses a counter *E* that stores the total number of event annotation and a counter *F(i,e)* that stores the number of times that each contact *i* is annotated during the event *e*. Hence, it is possible to calculate the event recurrence as *F(i,e)/E*. The difference between the DER and the ER estimators lies in the event information source: in the DER estimator, the user manually points out the beginning and the end of the event, whereas in the ER estimator, this information is retrieved automatically from the user calendar.

The disconnected spatial proximity estimator (DSP) uses bluetooth to identify the location of the closest user's friends at the moment of the photo capture. For each contact detected via bluetooth, the estimator value will be 1, if she is in the photo, and 0 if otherwise. The difference between the DSP and the SP estimators lies in the way the probability is assigned: DSP uses bluetooth and SP uses GPS data.

In order to generate the list of people suggested as possible annotations for a certain photo, the algorithm calculates, for each name contained in the mobile device friends list, the disconnected estimators' values, and performs a sum operation between their results. In both approaches, to generate the list of suggestions, the estimators had the same weight.

5 Event photo annotation

We begin by noticing the existence of several sources of information that can be explored to generate user's context. For example, collaborative networks create online communities which enable information sharing among subscribed users. Some of such networks provide geographical information sharing. Basically, some points of interest (POIs) such as restaurants, hotels, theaters, museum, and so on, are annotated and shared through digital maps among users. This information, available on the Internet, may help users to identify location context.

Concerning place names, a gazetteer which is available on the Web, such as the Getty Thesaurus of Geographic Names (TGN) and the Geonames, provides important information about geographic locations. A gazetteer links a place name to both location and type. With the help of gazetteers, users may annotate the place name with more details, such as the name of a building, a mountain, a road, a street, a city, a state or a country. The gazetteer in fact works as a dictionary for place names, so that users do not need to worry about latitude and longitude coordinates. Another important source of information for user context is the Web-based calendars, on which users annotate their appointments and share them with a user's community.

The business logic layer of the PhotoGeo architecture (Fig. 1) includes an event photo annotation module that explores user context thereby reducing users' efforts during the organization and annotation of their photo collections. The event photo annotation has the following requirements:

- The photos should be from a personal collection, so that there will be no parallel events;
- The photos should contain, among their metadata, information on latitude, longitude, date and time; and
- Users must utilize a Web-based calendar, which implements the iCalendar standard.

The algorithm for event photo annotation is divided into four steps: *spatial classification*, *temporal classification*, *improvement* and *naming*. After executing these four steps, there will be a set of photos separated into events. In the two first steps of the algorithm—*spatial* and *temporal classification*—photos are grouped according to similar properties.

In the first step, clusters of photos are formed based on geographic proximity. In the second step, each cluster is divided into subgroups that are close in time. After the separation of photos into clusters, inconsistencies may arise, e.g., photos from the same event may appear in different clusters, or photos from different events may be allocated to the same cluster. Thus, it is necessary to execute a further step to correct these mistakes. This step called *improvement* joins or divides clusters, by moving photos from one cluster to another, based on comparisons involving photo metadata. For instance, if two photos belong to different groups due to the large spatial distance between them no matter how close they are to each other as far as time is concerned, then probably they should be moved to the same group. The last step involves the *naming* of the extracted groups. In this step, the photo metadata is matched to the spatial and temporal information from the database so as to generate event names.

The event photo annotation is obtained when the user adds a photo collection to the library. This operation can also be carried out later onto another specific album. Since the event photo annotation module is based on the photo classification algorithm, it is possible to classify photos in the wrong events, or even to create events that should not exist. When such inconsistencies arise, the system enables the user to correct them manually. In other words, the system enables the manual annotation of an event in a photo. In this section, we discuss further each of the four steps of the event detector algorithm.

5.1 First step: spatial classification

The first step has to do with clustering photos based on geographic nearness, using unsupervised methods. The location of a photo is represented as a 2D point (latitude, longitude), extracted from the geographic metadata of the photo. This step receives, as input, a collection of geo-referenced photos and then returns a collection of clusters, each one with a collection of photos.

A possible approach to spatially classify the photos is to use the political division of countries. For instance, the photos would be grouped according to the cities where they were shot. The problem with this procedure is that some photos may be on the same event, but shot in different cities. Another possible solution is to use an unsupervised algorithm. In this case, the photos taken of nearby places, even if they are from different cities, may be put into the same cluster. We adopted this approach in our algorithm.

We tested the K-means algorithm [11] and the DBSCAN algorithm [5], a non-supervised classification algorithm based on density. However, as our event photo annotation algorithm implements the strategy design pattern, it is possible to change the spatial clustering algorithm on-the-fly, which enables the user to select her preferable algorithm.

Some parameters must be informed when an unsupervised classification algorithm is used, such as: the *seed* used and the number k of clusters in the K-means algorithm; the value of *epsilon*—the minimum distance among members of the same cluster—and the minimum number of elements in a cluster in the DBSCAN algorithm.

Depending on the input parameters for the clustering algorithms, we can obtain different results, using distinct granularities, which will result in clusters with different number of photos. Hence, these parameters may be used to customize the photo collections, according to the user's preferences. We will address the issue of granularity at the end of this section.

At the end of this first step, there will be a set of clusters which contain photos that are geographically near. It is interesting to notice that, by using only spatial clustering, it is not possible to detect events appropriately, as a given cluster may map to more than one event. For instance, consider two events that occur in the City of Campina Grande, at the State of Paraíba, in Brazil, during the Brazilian Database Symposium and during the summer vacation. These events happen at different times, but by using only spatial clustering, they will be put in the same cluster. Thus, it is necessary to refine the algorithm in order to better segment the photos according to events. The next step introduces temporal clustering as a way of improving the event photo annotation.

5.2 Second step: temporal classification

According to Cooper et al. [3] and Graham et al. [9] personal photos have an explosive behavior. For instance, people usually take several photos at special events, such as: birthday parties, wedding parties, and so on. Thus, the time interval between two consecutive photos taken at two different events tends to be greater than the time interval between two consecutive photos taken at the same event.

Temporal clustering first orders photos based on temporal metadata. Then, it calculates the time interval between consecutive photos as well as the average and standard deviation of the time intervals. Lastly, it separates consecutive photos into different clusters, if the time interval is greater than a given threshold t. The value of t is given by the sum of the average, standard deviation, and a parameter i. The parameter i is used to adjust the cases in which the average and standard deviation have values which are too small.

This step receives as input a set of cluster with spatially close elements and the parameter i, which is used to modify the level of granularity of the photo collections generated. As in the spatial clustering, according to the parameter value it is possible to obtain different event combinations. At the end of this step, each cluster will correspond to a set of events. This step may lead to at least two kinds of inconsistency: first, several events were joined in a unique event; and second, a unique event was segmented into several events.

5.3 Third step: improvement

This step attempts to reduce the inconsistencies created in the previous step. In this step, the algorithm extracts external information in order to improve the quality of segmentation. There are at least two types of information that can be extracted externally: spatial and temporal information.

To obtain spatial data externally, the algorithm requests them from Wikimapia. This step calls for places closer to the cluster generated in the first step. The input parameter is the bounding box of each cluster and a buffer of N meters. In our test we used N=500. Concerning the temporal clustering, the external information extracted comes from the user's calendar. We take two hypotheses to solve temporal inconsistencies:

- If a given event has photos which have been taken at different user appointments, then this event will be divided; and
- If there are two or more consecutive events, and they are in the same appointment in the user's calendar, then these events should be unified into a single event.

5.4 Fourth step: naming

The fourth and last step of the algorithm is responsible for naming the events obtained from the previous steps. We use spatial and temporal information to name events. The following format is adopted to name an event: "Event name + Place name + City + State + Country + Begin data + Event duration". An example of an event is: "Symposium on Multimedia—Dallas Hotel—New York—NY-USA—12/19/2009—3 days". From the user calendar, we obtain the event name that matches the date the photo was taken. If there are photos from more than one place in a single event, they will be annotated with spatial information related to the city, state and country. If there is more than one city, the spatial information will be restricted to state and country, and so on.

The place names, cities, states and countries are retrieved by the spatial data extractor. The places are initially looked up in the personal database of places; if there is no personal data, a query is then submitted to Wikimapia. The information regarding geopolitical division is obtained from the Geonames gazetteer. Information on date and event duration is extracted from the PhotoGeo database.

5.5 Granularity

PhotoGeo uses input parameters for customizing the event photo annotation algorithm. As mentioned in Section 5.2, the K-means algorithm needs the values of k and *seed* as input parameters. On the other hand, the DBSCAN algorithm needs the values of *epsilon* and the minimum number of elements in a given cluster as input parameters. Lastly, the temporal segmentation algorithm needs the value of i, which represents the time interval used to calculate the value of t, as described in Section 5.3.

By changing these input parameters separately, users may customize the resulting collections, so that there will be several events at different levels of granularity. In order to facilitate the use of PhotoGeo, these input parameters have default values, and it is a user's choice to change them to obtain a better organization according to her necessity.

Table 1 presents these input parameters, using as an example five predefined levels of granularity. The level 1 has the default values of each parameter. As we can see, the temporal granularity—represented by the input parameter i—is *hour*. The spatial granularity of the *epsilon* input parameter is *Km,* and the value of k is percentage (%). These values are internally converted to fulfill the requirements of the respective algorithms. For instance, the parameter k is converted into the number of directories that will be used to store the photos into different clusters by using the K-means approach. The *epsilon* parameter in DBSCAN represents the maximum distance that an element will be from the next group element. As the spatial data is represented by latitude and longitude coordinates, then the value informed in kilometers (Km) must be converted to degrees. The user can change the granularity level of each parameter separately, for instance, she can select DBSCAN in level 1 and temporal parameter in level 3.

6 PhotoGeo—retrieval and storage components

The use of a user friendly interface is an important requirement for the success of a multimedia digital library. However, most photo digital libraries offer a rather limited query mechanism, which means that the user cannot perform more advanced searches. As previously mentioned, the information where a photo was captured is most important for

Table 1 Predefined levels of granularity for the automatic organizer

Granularity level	Temporal	DBSCAN		K-means	
	i (hours)	*epsilon* (Km)	min number of elements	*k* (%)	seed
1	6	1	1	5	7
2	12	50	1	8	7
3	24	100	1	12	7
4	48	250	1	15	7
5	72	500	1	18	7

the user to remember. However, spatial operators have been little considered in such digital libraries. Thus, a more complex query, such as: “retrieve the photos such that Mike and Mary are shown and that were captured near New York, between the years 2003 and 2004”, has not been efficiently implemented.

The retrieval interface offers three types of constraints: conventional, spatial and temporal. Figure 3 shows the retrieval interface. On the left, are the fields that can be displayed in the query result, such as: description of the photo, date and time of capture, file name, and size. On the right, one can observe the possibility of creating a new filter for the query, and also to display the filters that have been created for the query. In Fig. 3 example, a filter was created for all types. The photos filtered were the ones captured between 2003 and 2005 within a distance of 50 km from Sao Paulo and that had the words “My Car” on their descriptions. No other photo digital library analyzed lets you do this type of query.

The query result is displayed in Fig. 4. On the left, a digital map of the photos is displayed. On the right, the list of results is shown. This list includes the fields which have been selected to come out before executing the query, the projection operator of the relational algebra. There is a filter that can be used to retrieve conventional metadata attributes such as: photo description, file name, flash status, camera manufacturer, camera model, directory, width and height. The spatial filter is meant to be applied to the geographic location of the photos. Among the various spatial operators, we chose *contains* and *far*. The temporal filter is applied to the Date / Time metadata, using the temporal

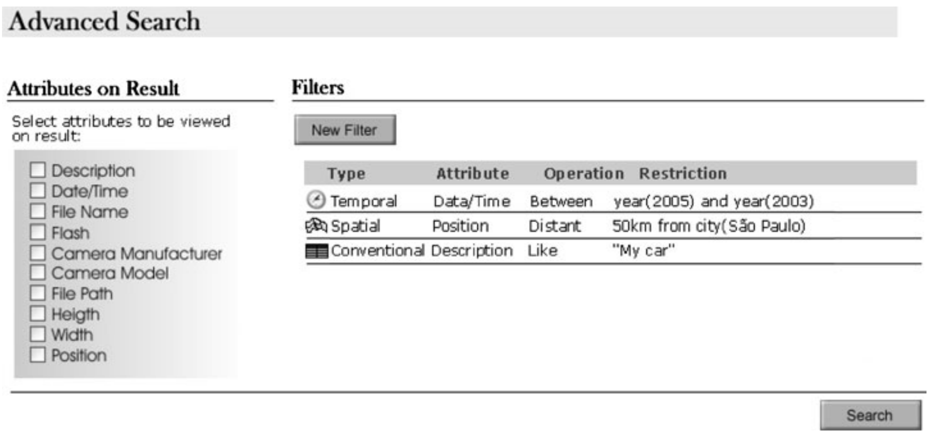


Fig. 3 PhotoGeo retrieval interface

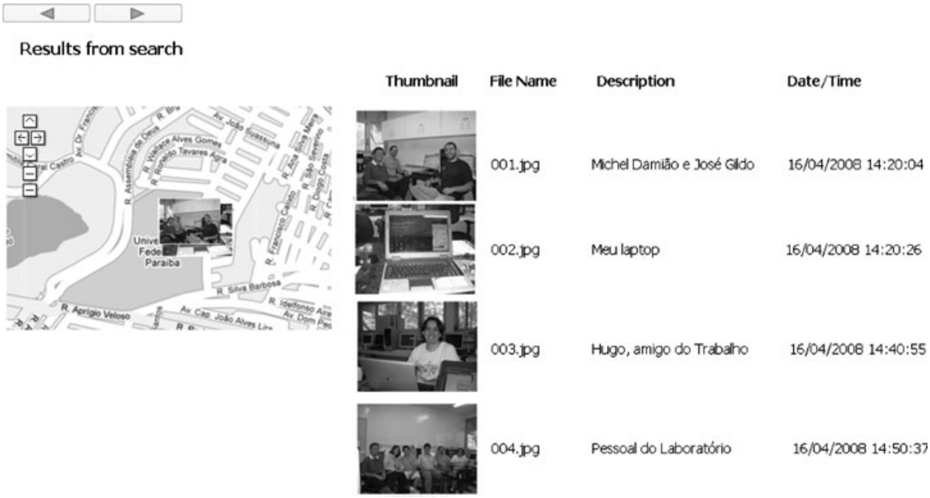


Fig. 4 A query result

operators—before, after, between, and during—to express the time interval the user is interested in.

The photographs and their metadata are stored into the data tier (see Fig. 1). This tier is made up of a database management server and a file server. Postgresql is used as a database server, and the PostGIS spatial library is employed to foster spatial capabilities. GIST indexing is used to improve performance on spatial queries.

Figure 5 shows an Entity-Relationship diagram for the PhotoGeo database schema. Such schema contains the following entities: user, album, calendar, appointment, location, event, photo, country, state, city and place.

As you can see in the proposed schema (Fig. 5), there is an entity to represent users. Each user may have access to calendars and albums. A calendar stores information extracted from the temporal data collector. An album may contain several photos. Each photo, on the other hand, may draw reference to an event, a place or a city. The geo-referenced information extracted from the spatial data collector is stored in entities, including city, state, country, and place.

The EXIF standard stores photo metadata and has been incorporated into a large number of digital cameras from different makers. As to the PhotoGeo schema, the photo table stores the EXIF metadata into the database server for future retrieval.

7 Experiments

In this section, we present the experiments executed to measure the effectiveness and efficiency of the algorithms discussed in this article.

7.1 Event photo annotation

As discussed in Section 5, PhotoGeo outputs a set of photos clustered into events. In our experiments, we used a total of 2,412 geo-referenced photos, divided into five collections. Each collection is from a different user. Also, we used the calendars of the owners of the collections.

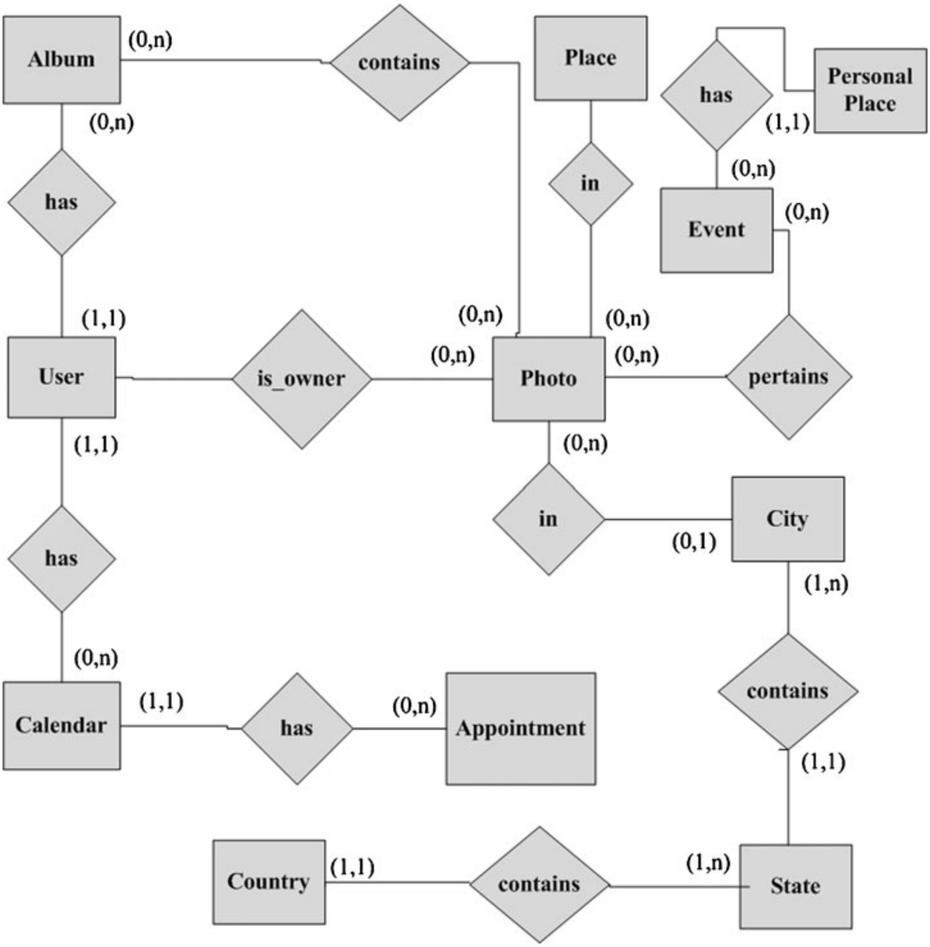


Fig. 5 PhotoGeo schema

Collection A has 596 photos, which are distributed in approximately 20 months. During this time span, there are months with 0–3 events. Collection B contains 162 photos and has a time interval of 11 months. Collection C contains 238 photos and has a time interval of approximately 11 months. Collection D contains 541 photos and has a time interval of 12 months. Lastly, collection E contains 812 photos taken during a 30-month period. All collections contain photos shot in Brazilian cities, except collection E, which contains photos from Brazil and Canada. In collections A and B, almost all photos have events registered in the user's calendar. In collection D, only 25% of the photos have appointments registered in the calendar. Collections C and E have no events described in user's calendars.

We have chosen two approaches to validate the algorithms. The first approach was based on a questionnaire answered by the users who have their photo collections automatically organized by PhotoGeo and who took part in its evaluation. In the second approach, we made comparisons between manual and automatic photo organization.

7.1.1 User analysis

After running the tests, the users evaluated the effectiveness of the proposed algorithm by answering a questionnaire with the following questions:

- Question 1: How many events were correctly detected?
- Question 2: How many names have represented the events well?
- Question 3: Try to find five photos without considering their events. How has the proposed storage structure enabled you to find these events?
  ( ) slowly ( ) normal ( ) quickly.
- Question 4: In general, how do you define the self-naming given to the events?
  ( ) very bad ( ) bad ( ) normal ( ) good ( ) excellent.

One of the main contributions of the proposed algorithm is the use of external sources found in step 3 to better separate the photos into events. Hence, we have done two kinds of analysis: *full* and *partial* analysis. The *full analysis* executed all steps of the proposed algorithm. For the partial analysis we took into account only steps 1, 2 and 4. Thus, the influence of step 3 in the results was evaluated. We have used the following metrics:

- The quality of classification;
- The quality of the self-naming of events; and
- The importance of the calendar.

For the users' evaluation, the photo collections' main features were analyzed. It was also verified that most photos were taken in a small geographic area. Thus, the DBSCAN algorithm was defined as the spatial clustering algorithm, and the value of epsilon parameter was set to 1 Km. As for the temporal part, it was found that most users rarely participate in several events on a single day. Consequently, the value of i was defined as 24 h.

According to the results obtained from the execution of the entire algorithm, users from collections A and B have considered as excellent the classification and naming of the events, as shown in Fig. 6. This was obtained for almost all events annotated in the user's calendar.

For collection C, PhotoGeo results were evaluated as good, resulting in 83% of user acceptance. The user mentioned that there was an event that was divided into three others. In this particular event, the photos were captured along three consecutive days. The user

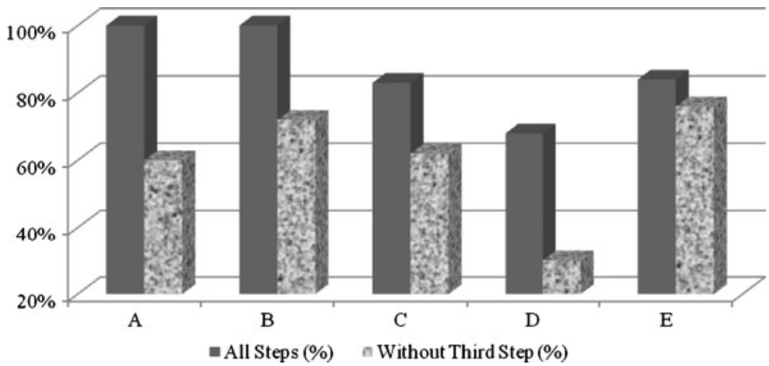


Fig. 6 User acceptance of the event photo annotation results

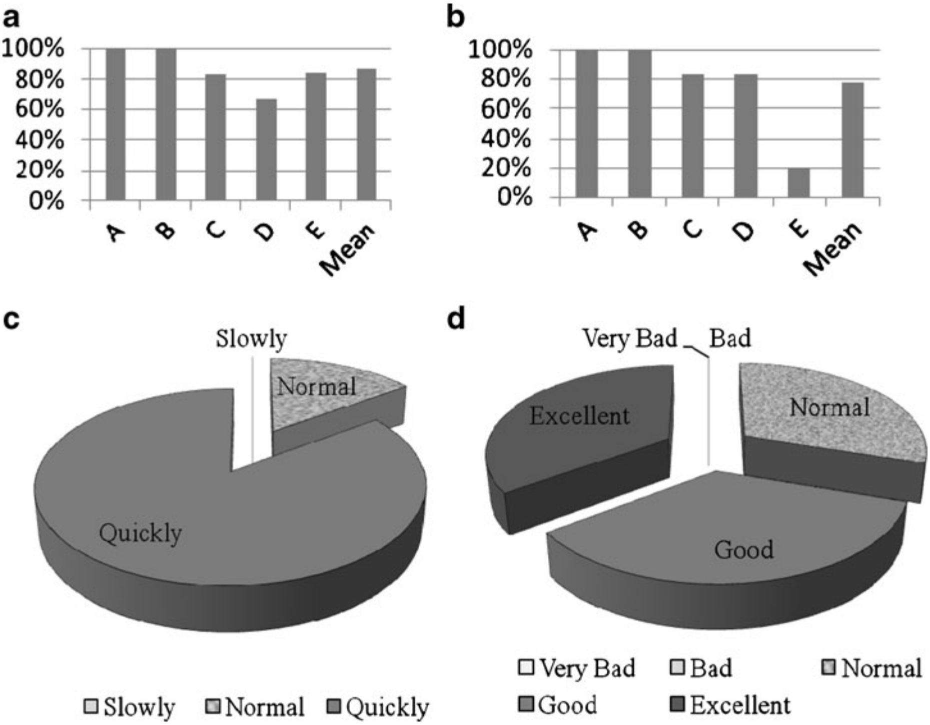


Fig. 7 Results from the questionnaire: **a** question 1; **b** question 2; **c** question 3; **d** question 4

found the use of a calendar most appropriate, and mentioned that the PhotoGeo system was very useful to automatically generate a first approximation of the photo classification, which should then be manually refined.

Collection D had one inconsistency regarding event separation. The inconsistency was related to a user trip which took two days and should be classified as a single event, but it was considered two events. However, this trip was not annotated in the user calendar.

Finally, in collection E, 84% of the events were sorted out correctly. Only 20% of the events had a correct naming. This happened due to the fact that the photos taken in Canada could not have the spatial dimension properly set, but only the temporal dimension. Figure 7 presents the result of the applied questionnaire.

The first metric which we are going to analyze is the correct separation of the events. This is a subjective metric, because a given collection of photographs can be separated into different ways, and all of these ways are correct. But all users have their own preferences, so an organization may be good to one user and bad to another. Based on the results of question 1 (Fig. 7a) there was an average of 87% of approval of users. Another important result is related to Fig. 7c, which presents that the majority of users had rapid access to their photos.

With regard to the quality of the self-naming of events, we used answers to questions 2 and 4. Figure 7b presents the result for question 2. The results for question 4 (Fig. 7d) varied between average and excellent.

The results of the event photo annotation without the step 3 of the algorithm may be visualized in Fig. 6. In this analysis it was checked if the event separation was in accordance to user requirements.

One should notice that, in all collections, the results using the full algorithm were better than the others. By not applying step 3, the user acceptance average dropped from 87% to 60%. These results have demonstrated the importance of the inconsistency correction done in this step.

7.1.2 Automatic event detection versus manual event detection

In order to evaluate the effectiveness of the levels of granularity, comparison has been drawn between the events manually annotated by the user—this has been rated the best—and the events automatically annotated by event photo annotation module. In order to measure how similar these events were, we used the classical information retrieval metrics: *precision*, *recall* and *f-measure*.

To facilitate the description of the experiments performed, we will use the following convention: automatic photo classification will be called *event*, and the manual photo classification will be called *directory.*

In our evaluation, *precision* is the percentage of photos of a given event which are correctly classified. *Recall* is the percent of photos in a directory which are classified in the right event. To calculate *precision* and *recall*, it is necessary to associate each directory to an event. In order to do that, the following procedures were carried out: (i) the event that contains the largest photo recall from one directory will be related to it; (ii) if there is more than one event with the same precision related to a directory, then the event with higher precision will be chosen. Thus, it is possible for an event to be associated with more than one directory; however, a directory must be associated with only one event. The *precision*, *recall*, and *f-measure* are calculated according to Eqs. 3, 4 and 5, respectively:

$$\Pr ecision(d,e) = \frac{p(d,e)}{p(d,e) + fp(d,e)} \tag{3}$$

$$\operatorname{Re} call(d,e) = \frac{p(d,e)}{p(d,e) + fn(d,e)} \tag{4}$$

$$F - Measure(d,e) = 2 \times \frac{\Pr ecision(d,e) \times \operatorname{Re} call(d,e)}{\Pr ecision(d,e) + \operatorname{Re} call(d,e)} \tag{5}$$

Where:

- p(d,e) is the number of photos which were correctly classified from directory *d* in event *e;*
- fp(d,e) is the number of photos classified in event *e* that do not belong to directory *d; and*
- fn(d,e) is the number of photos from directory *d* which were not classified into event *e.*

Figure 8 shows a set of graphics representing precision versus several predefined levels of granularity related to the five photo collections studied in our experiments using all steps of the event photo annotation algorithm. Note that precision decreases as granularity increases for both spatial clustering algorithms. This is due to the fact that, as the level of granularity increases, the number of photos of a given event increases as well; as a result, precision decreases. Furthermore, the first level of granularity led to events closer to the directories. For the first level of granularity with DBSCAN, an average precision of 92.86% was obtained; whereas the precision for the fifth level was 74.75%. By using K-means, an

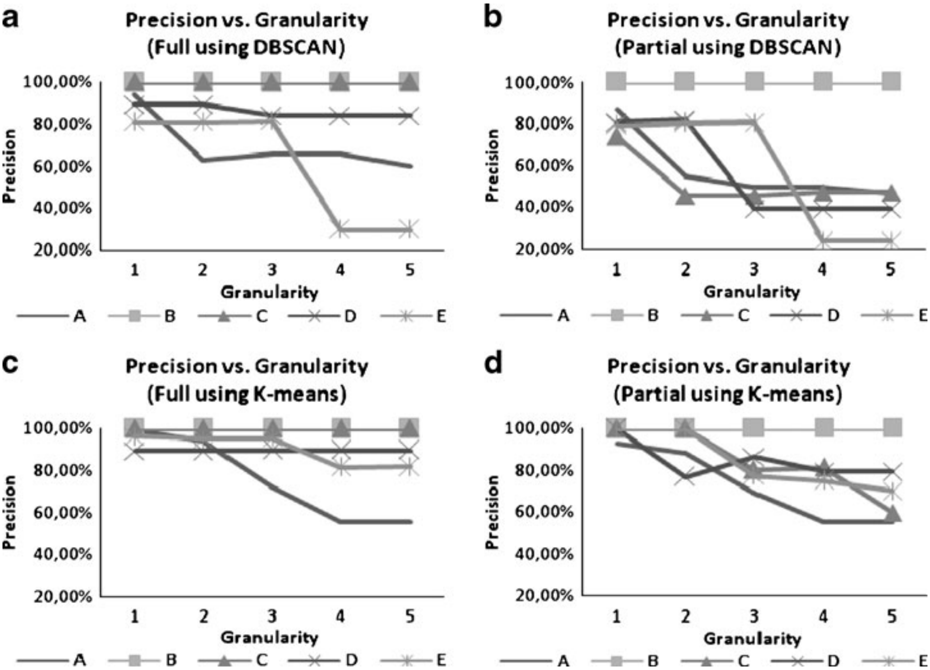


Fig. 8 Precision of the event photo annotation algorithm: **a** DBSCAN full analysis; **b** DBSCAN partial analysis; **c** K-means full analysis; **d** K-means partial analysis

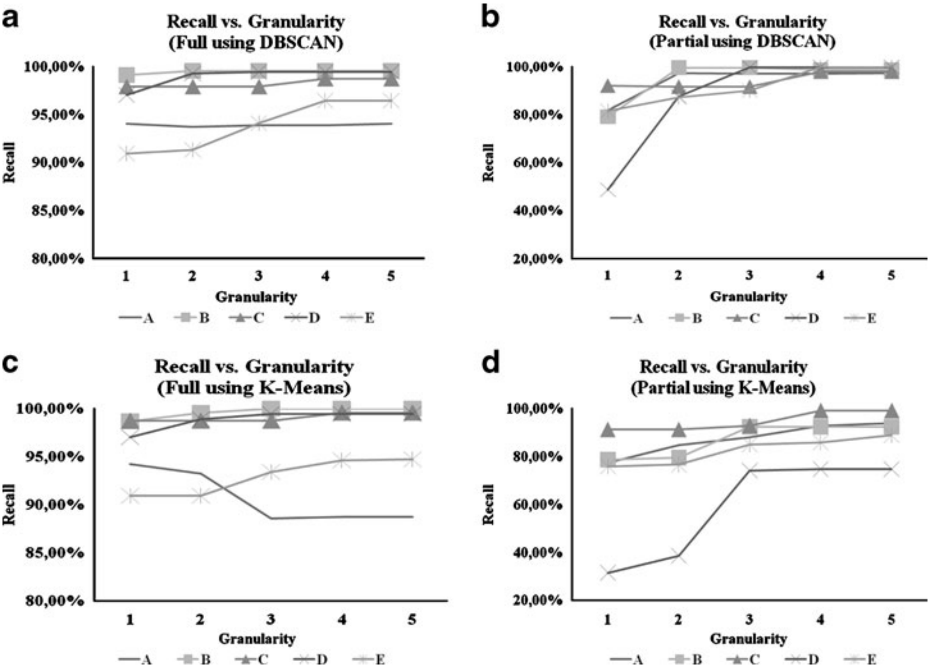


Fig. 9 Recall of the event photo annotation algorithm: **a** DBSCAN full analysis; **b** DBSCAN partial analysis; **c** K-means full analysis; **d** K-means partial analysis

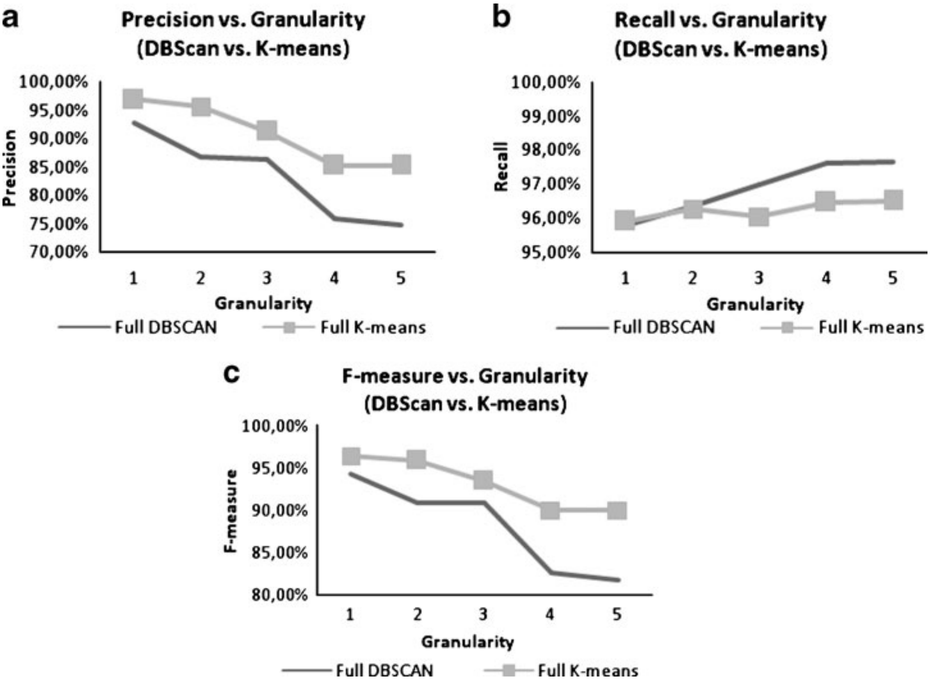


Fig. 10 Metrics versus granularity: **a** precision of DBSCAN and K-means; **b** recall of DBSCAN and K-means; **c** F-measure of DBSCAN and K-means

average precision of 96.96% was obtained for the first level of granularity; whereas the precision for the fifth level of granularity was 85.34%.

The use of the third step of the event photo annotation algorithm proved essential, as the results of precision and recall metrics improved substantially by using this step. The precision increased 18.67% when using DBSCAN, and it increased 5.9%, when using K-means (see Fig. 8b and d).

Figure 9 shows the recall curves for the five collections of photos used in our experiments. In these graphics, recall increases as granularity increases. This is due to the larger number of photos on the same event. Observe that recall increased 4.47% and 14.2% with DBSCAN and K-means, respectively (see Fig. 9b and d). This result confirms the results obtained with the user's questionnaires.

Figure 10 shows the comparison between the DBSCAN and K-means algorithms. The results indicate that K-means achieved better precision than DBSCAN; however, the latter

Table 2 Statistic of the evaluated collections

	Collection
Total number of photos	1,505
Total number of photos with annotations	1,097
Total number of different people in the photos	206
Total number of annotations	2,416
Average number of people per photo	2.2
Average number of photos per person	11.73

Table 3 Hit rate of the proposed algorithms

Number of suggestions	Connected	Disconnected
1	34.2%	18.8%
2	51.8%	35.6%
3	62.7%	49.2%
4	69.0%	58.2%
5	74.1%	64.4%
6	77.7%	68.5%
7	79.5%	71.2%
8	81.0%	72.1%
9	82.1%	72.6%
10	83.2%	73.1%

achieved a better recall. Yet, as the difference between the recall of the two algorithms is small, precision becomes more important. As seen in Fig. 10a, the input parameters have significant influence on the obtained results.

7.2 People annotation

Tests were also performed to measure the effectiveness of the connected and disconnected algorithms for the suggestion of people photo annotations by using mobile devices. The metric used to perform this measurement was the hit rate, as defined in Section 4.2. Recall that $p_f \subseteq P$ denotes the set of people in P who appear in a photo f and that $H_f \subseteq P$ indicates people suggested to be in f. Let $i \in P$. We say that a *hit occurs for i* iff $i \in p_f \cap H_f$, and that a *miss occurs for i* iff $i \in p_f - H_f$. The data collection used has 1,505 photos, all containing spatial and temporal metadata, as described on Table 2

Once the information about who was shown in a given photo has been collected, the experiments to annotate the photos were performed without human intervention. The system generates a set H_f of suggestions and then the misses and hits are obtained with respect to the set p_f of people which took part in photo f.

The first analysis performed considered the hit rate of the algorithms, based on an experiment using a set H that contains N suggestions, so that $1 \leq N \leq 10$. For the connected algorithm, hit rates of 74.09% and 83.24% were observed, for $N=5$ and $N=10$, respectively. For the disconnected algorithm, the hit rate was smaller than that of the connected algorithm, as the disconnected algorithm adopts simpler estimators. In this case, the hit rates were 64.4% and 73.1%, for $N=5$ and $N=10$, respectively.

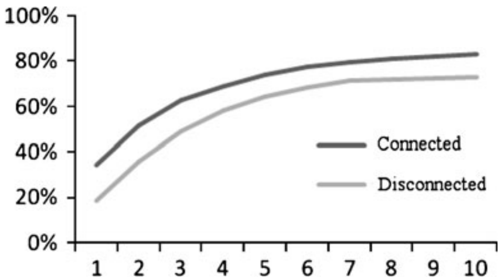


Fig. 11 Analysis of the hit rate of connected vs. disconnected algorithms

Table 4 Rate of the algorithms concerning the number of people annotated

Number of people	Connected	Disconnected
10	93.1%	75.1%
20	89%	74.3%
30	86.3%	72.8%
40	83.9%	71.5%
50	82.6%	70.6%
All	74.1%	64.4%

Table 3 and Fig. 11 present the results obtained in the experiment. Figure 11 shows the hit rate for each algorithm, for different size values of the suggestion list. Note that the connected algorithm obtained an average result 11.16%, far better than that of the mobile device.

A second experiment was performed in order to measure the performance of the connected and disconnected algorithms by using a list of five suggestions and considering only the annotations of the M most recurrent people in the annotations, with $10 \leq M \leq |P|$. In this case, the results were better than the previous ones.

Table 4 and Fig. 12 present the results obtained in this experiment. Figure 12 shows the hit rate of the algorithms with respect to the number of different people annotated. Observe that both algorithms had better hit rates as the number of annotated people decreased. One should notice again that the connected algorithm presented better improvement when compared to the disconnected algorithm.

8 Conclusions

This article describes PhotoGeo, a prototype tool designed to help the user to retrieval photos by means of the information about who, when and where of the photos; and annotate photos with these information.

The system incorporates two major algorithms that help the user with the annotation of photos. The first algorithm performs event photo annotation (*when* and *where*), and the second algorithm performs people photo annotation (*who*). The event photo annotation algorithm uses a collaborative mapping tool and the user's calendar to improve the precision, coverage and the naming of the events. The people photo annotation algorithm uses two methods, called connected and disconnected. In the connected method, the mobile device connects to the data server of the PhotoGeo system, and uses the proposed estimators to provide a list of suggested annotations. In the disconnected method, the list of suggestions is generated in the mobile

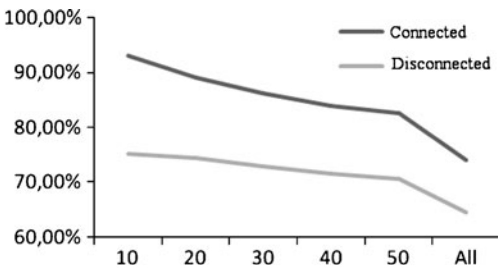


Fig. 12 Analysis of hit rate concerning the number of people annotated

device, and uses a simplified version of some of the estimators adopted for the connected method. This is mainly due to the limited resources provided by mobile devices. The article also includes experiments to validate the algorithms.

As future work, we intend to test other algorithms for spatial and temporal clustering in the event photo annotation module. Besides, we will investigate how to turn the input parameter automatically set, based on a case-based reasoning technique in order to obtain a more effective separation for each user. As for the people photo annotation module, we intend to synchronize the information used in both connected and disconnected methods, and explore the data present in the collaborative mapping tool as an estimator for the ranking. Furthermore, face recognition techniques will be added to verify if the results may be improved.

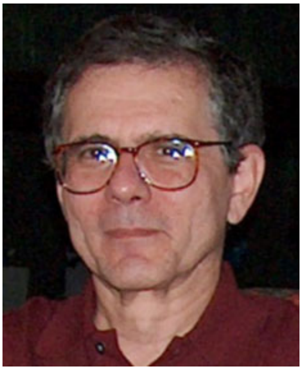
References

1. Ames M, Eckle D, Naaman M, Spasojevic M, House NV (2010) Requirements for mobile photoware. Personal and Ubiquitous Computing 14(2):95–109
2. Baptista CS, Kemp Z (2000) An integrated metamodel for knowledge representation in geolibraries. In: Proceedings of the IEEE Advances on Digital Libraries, Washington D.C., USA, pp 151–159
3. Cooper M, Foote J, Girgensohn A, Wilcox L (2005) Temporal event clustering for digital photo collections. ACM Transactions on Multimedia Computing, Communications, and Applications 1 (3):269–288
4. Davis M, King S, Good N, Sarvas R (2004) From context to content: leveraging context to infer media metadata. In: Proceedings of the 12th Annual ACM International Conference on Multimedia, New York, USA, pp 188–195
5. Ester M, Kriegel HP, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, Portland, USA, pp 226–231
6. Figueirêdo HF, Silva TE, Paiva AC, Queiroz JER, Baptista CS (2010) Infrastructures for development of context-aware mobile applications. Handbook of research on mobility and computing: evolving technologies and ubiquitous impacts. IGI Global
7. Frohlich D, Kuchinsky A, Pering C, Don A, Ariss S (2002) Requirements for photoware. In: Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work, New Orleans, USA, pp 166–175
8. Goodchild MF, Hill LL (2008) Introduction to digital gazetteer research. International Journal of Geographical Information Science 22(10):1039–1044
9. Graham A, Garcia-Molina H, Paepcke A, Winograd T (2002) Time as essence for photo browsing through personal digital libraries. In: Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries, Portland, USA, pp 326–335
10. JEITA CP-3451 (2002) Exchangeable image file format for digital still cameras: Exif Version 2.2. Japan Electronics and Information Technology Industries Association. http://www.exif.org. Accessed 13 March 2010
11. MacQueen JB (1967) Some Methods for classification and analysis of multivariate observations. In: Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, USA, pp 281–297
12. Naaman M, Harada S, Wang Q, Garcia-Molina H, Paepcke A (2004) Context data in geo-referenced digital photo collections. In: Proceedings of the 12th Annual ACM International Conference on Multimedia, New York, USA, pp 196–203
13. Naaman M, Yeh RB, Garcia-Molina H, Paepcke A (2005) Leveraging context to resolve identity in photo albums. In: Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries, Denver, USA, pp 178–187
14. O'Hare N, Smeaton AF (2009) Context-aware person identification in personal photo collections. IEEE Transactions on Multimedia 11(2):220–228
15. Pham T, Maillot NE, Lim J, Chevallet J (2007) Latent semantic fusion model for image retrieval and annotation. In: Proceedings of the 16th ACM Conference on Information and Knowledge Management, Lisboa, Portugal, pp 439–444
16. Rodden K, Wood KR (2003) How do people manage their digital photos? In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Ft. Lauderdale, USA, pp 409–416
17. Toyama K, Logan R, Roseway A (2003) Geographic location tags on digital images. In: Proceedings of the Eleventh ACM International Conference on Multimedia, Berkeley, USA, pp 156–166
18. Viana W, Miron A, Moisuc B, Gensel J, Villanova-Oliver M, Martin H (2010) Towards the semantic and context-aware management of mobile multimedia. Multimedia Tools and Applications (Online First):1–39

Hugo Feitosa de Figueirêdo is a Ph.D student at the Computer Science Department of the Federal University of Campina Grande. He is member of the Database and Information Systems Laboratory where he is doing research on context-aware applications and spatial database under the supervision of Dr. Cláudio de Souza Baptista.

Yuri Almeida Lacerda is a Professor at the Federal Institute of Ceará, Brazil. He received a Master degree in Computer Science from University of Campina Grande, Brazil, in 2009. His research interests include database, digital libraries, geographical information systems and multimedia databases.

Anselmo Cardoso de Paiva is an Associated Professor at the Computer Science Department at the University of Maranhão, Brazil. He received a Doctor degree from PUC-Rio, Brazil in 2001. His research interests include computer graphics, geographical information systems, medical images system, and information systems.



Marco Antonio Casanova has a Ph.D. in Applied Mathematics from Havard University (1979), an M.Sc. in Computer Science from the Pontifical Catholic University (PUC) of Rio de Janeiro (1976), and a B.Sc. in Electronic Engineering from the Military Institute of Engineeruing (1974). He joined IBM Brazil in November 1982, where he is now Manager of the IBM Brazil Cientific Center. From 1980 to 1982, he was Assistant Professor at the Department of Informatics of PUC Rio, where he also acted as Graduate Program Coordinator during 1981–82.

Cláudio de Souza Baptista is an Associate Professor at the Computer Science Department and Director of the Information Systems Laboratory at the University of Campina Grande, Brazil. He received a PhD degree in Computer Science from University of Kent at Canterbury, United Kingdom, in 2000. His research interests include database, digital libraries, geographical information systems and multimedia databases. He has authored more than 40 papers in international conferences, book chapters and journals.