iScope: Personalized Multi-Modality Image Search for Mobile Devices

Changyun Zhu*, Kun Li†, Qin Lv‡, Li Shang†, Robert P. Dick§

* ECE Department
Queen's University
Kingston, ON K7L 3N6, Canada
changyun.zhu@ece.queensu.ca

† ECEE Dept. and ‡ CS Dept.
University of Colorado at Boulder
Boulder, CO 80309, U.S.A.
{kun.li, qin.lv,
li.shang}@colorado.edu

§ EECS Department
University of Michigan
Ann Arbor, MI 48109, U.S.A.
dickrp@eecs.umich.edu

ABSTRACT

Mobile devices are becoming a primary medium for personal information gathering, management, and sharing. Managing personal image data on mobile platforms is a difficult problem due to large data set size, content diversity, heterogeneous individual usage patterns, and resource constraints. This article presents a user-centric system, called iScope, for personal image management and sharing on mobile devices. iScope uses multi-modality clustering of both content and context information for efficient image management and search, and online learning techniques for predicting images of interest. It also supports distributed content-based search among networked devices while maintaining the same intuitive interface, enabling efficient information sharing among people. We have implemented iScope and conducted in-field experiments using networked Nokia N810 portable Internet tablets. Energy efficiency was a primary design focus during the design and implementation of the iScope search algorithms. Experimental results indicate that iScope improves search time and search energy by 4.1× and 3.8× on average, relative to browsing.

Categories and Subject Descriptors

C.5 [**COMPUTER SYSTEM IMPLEMENTATION**]: Portable Devices; H.3 [**INFORMATION STORAGE AND RETRIEVAL**]: Information Search and Retrieval

General Terms

Algorithms, Design, Management

Keywords

Retrieval, Management, Energy, Performance

1. Introduction

Personal, portable communication and computation devices are now part of hundreds of millions of lives, often in the form of smart-phones. Emerging mobile applications and services are the main driving forces behind the increasing prevalence of personal mobile systems. From Daniel Henderson's 1993 prototype, *intellect*, which can receive and display images and video media [1], to the first photo taken by Philippe Kahn in 1997 using a camera phone and shared instantly with more than 2,000 families [2], the functionality and adoption of personal portable devices have continuously increased. Today's personal portable devices, such as the iPhone from Apple, Blackberry from RIM, and Android phone from Google, have integrated a variety of system functions, such as global positioning system (GPS), cameras, sensors, large touch screens, and easy-to-use interfaces. Global mobile phone subscriptions have reached 3.3 billion in 2007 [3]. Users are able to capture information anywhere and anytime. In addition, these devices are heavily used for information sharing and social interaction.

Mobile devices are the first-level interface for capturing and sharing multimedia data such as images. They are therefore a natural image data management platform. Managing image data on mobile devices, however, is a challenging problem. A picture may be worth a thousand words, but without knowing the words that describe it, search can be difficult. Manual image annotation is tedious and time consuming. *Content-based image retrieval (CBIR)*, which automatically extracts representative features from raw data and uses the extracted features to locate content of interest, largely automates managing and exploring image data [4]. Despite the recent progress in CBIR, managing image data on personal mobile systems is challenging due to large data set size, content diversity, heterogeneous user interests and usage patterns, and resource constraints.

- **Energy-induced constraints:** Energy consumption is a foremost design concern in battery-powered mobile systems. Scarce energy resources largely limit the performance and functionality of software applications running on portable devices. Existing CBIR techniques have high computation complexity and storage requirements. User interaction and communication bring high time and energy cost during personal image search. Energy-induced design constraints introduce serious challenges to the design and implementation of image management systems on personal mobile devices.
- **User-specific search scenarios:** Unlike general-purpose search techniques developed for the World Wide Web, image management on personal mobile devices is a highly personal, user-centric task. Typically, a mobile device has one owner; the data captured and stored on a device depend on its owner's interests. Search patterns are also user specific.

This work was supported in part by the National Science Foundation under awards CNS-0347941 and CNS-0720691.

MobiSys'09, June 22–25, 2009, Kraków, Poland.

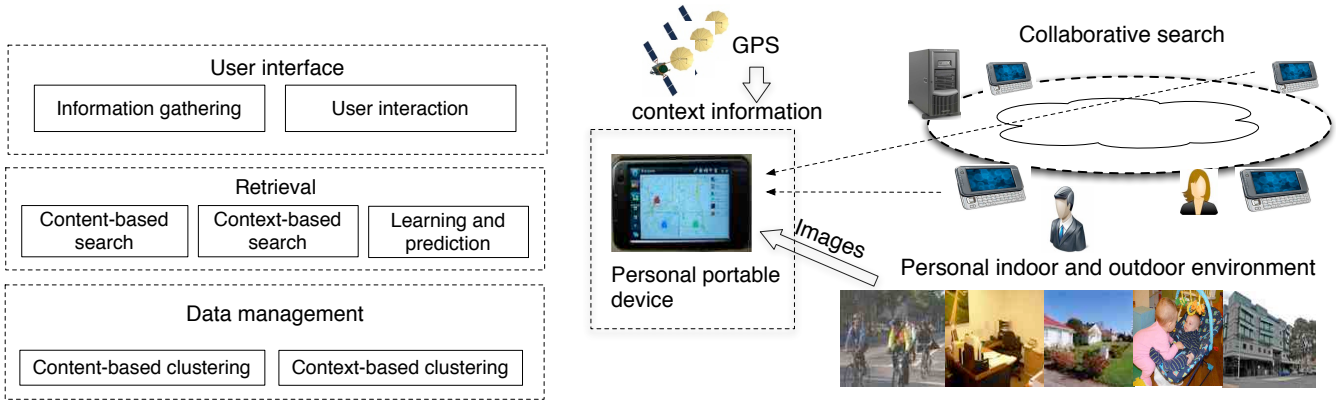

Figure 1: Overview of iScope's system architecture.

iScope adapts to an individual user's data characteristics and search patterns to improve search performance, quality, and energy efficiency.

• **Distributed data sharing:** Supporting data sharing in distributed mobile environments requires efficient, distributed data management and search techniques. Communication-induced energy overhead is of great importance. We investigate the time and energy overhead of remote image retrieval and propose collaborative search and metadata caching techniques to allow efficient image sharing and retrieval in distributed mobile environments.

In this work, we propose iScope, a personal content management platform. iScope is a user-centric design targeting energy-constrained distributed mobile environments. It leverages both personal context information and efficient content search techniques, as well as online learning techniques, to deliver personalized, energy-efficient content search services. It provides a collaborative search environment, enabling distributed image search on mobile devices, thus facilitating information discovery and social interaction. We have implemented a prototype of iScope and conducted in-field experiments using Nokia N810 portable Internet tablet devices. The proposed software platform will soon be publicly released for free academic and personal usage.

The rest of this article is organized as follows. Section 2 provides an overview of the iScope system architecture. Section 3 presents multi-modality image data management. Section 4 conducts resource characterization of portable platform and investigates the resource usage of image search process. Section 5 and Section 6 describe the benefits of personalized and collaborative image search. Sections 7, 8, and 9 describe the result of experiments with the iScope prototype, survey related work, and conclude.

2. Overview of the iScope System Architecture

This section presents an overview of iScope's system architecture. Figure 1 illustrates the system architecture of iScope, which consists of the following components.

• *Multi-modality data management:* Personal image data contains a rich set of content information, such as color, texture and shape, and user-specific context information, such as location, time and ownership. In iScope, the context and content information of personal image data are used in unison to enable efficient image management. Images are partitioned based on content features and context metadata. The proposed hierarchical clustering-based multi-modality data management design allows efficient traversal of the data set across different feature dimensions and resolutions, enabling efficient management of personal data sets and run-time user queries. (Section 3)

• *User-centric adaptive image search:* iScope offers personalized image search by leveraging content-based search algorithms with user-specific context information. Users differ from each other on image interests and performance expectations. iScope incorporates run-time learning techniques for online prediction and adaptation of the search process based on implicit user feedback, improving search quality and minimizing search costs, e.g., energy consumption. (Section 5)

• *Distributed collaborative search:* iScope supports remote image search and metadata caching among distributed image data sets spanning multiple mobile devices, with the goal of enabling efficient information sharing and effective social interaction in mobile social networks. (Section 6)

iScope supports personal image search on mobile devices through an iterative search process. Personal images are organized using the clustering-based multi-modality data management on a local mobile device or multiple distributed devices. At each retrieval step, given a user's feedback, e.g., a query image or context information, iScope traverses through the hierarchical multi-modality data clusters stored either locally or remotely, predicts and identifies a potential match, and returns the candidate thumbnail images of the matched cluster to the user. The search process continues until the target image(s) are found. Figure 2 illustrates the interactive search process. In this example, a user looks for a photo taken during his hiking trip last year. Starting with a photo of his recent paint-ball trip, the user conducts three context search operations based on location, time, and again on location. The returned photo contains content similar to the target image. The user then uses content-based search to find the target hiking-trip photo.

3. Multi-Modality Data Management

iScope uses a multi-modality image management scheme that uses both content and context (e.g., time, location, and ownership) information associated with images. This section explains how these data are obtained and used in multi-dimensional, multi-scale image clustering to support rapid browsing and run-time user queries.

3.A Content-Based Image Clustering

Given the raw content of an image, various types of image features may be extracted, such as color, texture, and shape.

Figure 2: The image search example using iScope.

Functions of these features can be used to quantify image similarity. In this work, we consider the following image features:

- *Color histogram:* For each image, a color histogram is extracted in the hue–saturation–value color space. A Euclidean distance function is used. We experimented with two different setups, one using 162-dimensional ($18 \times 3 \times 3$) histograms, and the other using more compact 32-dimensional ($8 \times 2 \times 2$) histograms.
- *Wavelet coefficients:* [5] This technique performs multi-resolution wavelet decomposition on each image and extracts the most significant coefficients (363 coefficients in our study). Similarity between two images is measured by the number of coefficients they have in common.
- *Region-based features:* [6] This method segments each image into regions and extracts a 14-dimensional feature vector (9-dimensional color moments and 5-dimensional bounding box) from each region. The distance between two region feature vectors is defined as ℓ_1 distance and the distance between two images is defined by the improved Earth Mover's Distance (EMD), a metric used to evaluate the similarity of two multi-dimensional distributions. EMD measures the minimal amount of work needed in order to transform one distribution into the other. Due to the high complexity of EMD calculation, we also studied a simplified distance computation, called *Region-abc*, which is defined as the sum of the best-matched distances for each individual region.
- *SIFT:* [7] This method detects hundreds of local salient regions and extracts a 128-dimensional, scale-invariant feature vector for each local region. Euclidean distance is used to determine the feature vector difference, and the distance between two images is measures by the total number of matched local feature vectors.

For each of the six content-based image features (*Color-162*, *Color-32*, *Wavelet*, *Region-EMD*, *Region-abc*, and *SIFT*), we apply the following modified agglomerative hierarchical clustering algorithm. Our approach yields a clustering tree structure with hierarchical levels of resolutions instead of a single-layer clustering structure, which enables compact data management and efficient search process. Starting with single-image clusters, this algorithm recursively merges the two closest clusters if the merged cluster has at most M images, where M is the number of thumbnail images that can be displayed on the screen of a portable device. M is set to 24 in our study. On top of the content-based clusters, we also construct a content-based cluster relationship graph in which each node represents a content cluster and each edge represents the distance between the centroids of two clusters. Using this graph, we can quickly identify other clusters that are likely to contain images with similar content to those in a given cluster, thus permitting efficient content-based image browsing and retrieval.

3.B Context-Based Image Clustering

In addition to image content information, our system also uses various types of context metadata to improve image data management quality and efficiency.

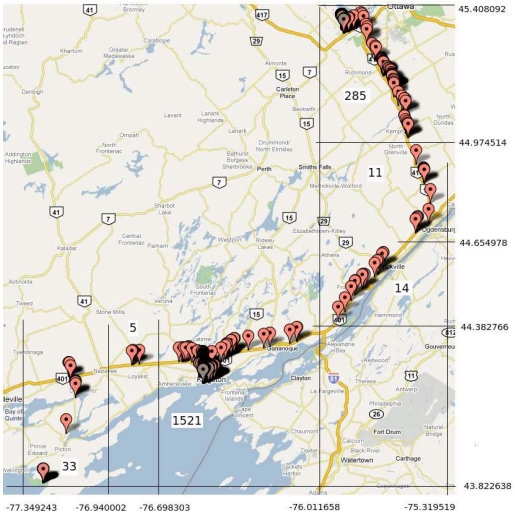

Figure 3: Geographical distribution of a personal image data set.

Geographical location information of images can be captured by mobile devices equipped with GPS receivers, permitting easy computation of the spatial correlation among images. Figure 3 shows an example geographical distribution of a user's image data set. Using agglomerative hierarchical clustering, we start with single-image clusters and recursively merge the closest pair of clusters until all images belong to the root cluster. Unlike the content-based clustering technique described above (which maintains a set of flat clusters), context-based clustering maintains the entire cluster hierarchy, except for small clusters whose parent clusters have few enough (M) images to be displayed on the screen of a mobile device.

Similarly, hierarchical time clusters can be constructed, each containing images captured within a certain time period. Temporal correlation among images has been observed in many scenarios and can help identify images of certain activities (e.g., wedding) or images taken at a certain time (e.g., Macy's Thanksgiving day parade). In this work, distance in the time domain is measured as the absolute time difference between pairs of images.

For distributed image sharing, ownership information also plays an important role. A user may obtain images from other users, and through the ownership information, identify

other users with similar interest (e.g., classic cars). This can be used to restrict image browsing or searching to a specific set of friends, improving the efficiency of image search and enabling more effective social interaction.

3.C Interactions of Multi-Modality Clusters

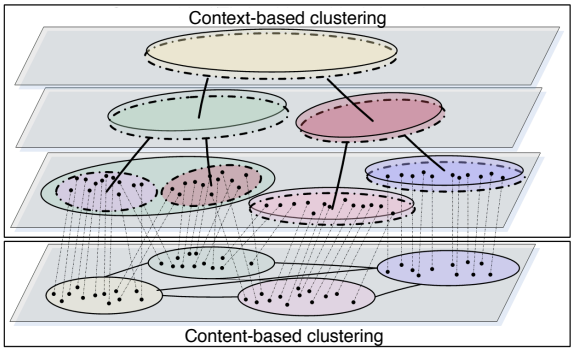

Figure 4: Interactions of content-based and context-based clusters.

Figure 4 illustrates the iScope multi-modality image clustering method, using metadata information such as content, location, and time. At the bottom of the figure, images are clustered based on their content similarity. There are also links between content clusters indicating the closeness of cluster centroids. At the top of this figure, hierarchical geographical clusters (solid-line ellipses) and time clusters (dotted-line ellipses) are maintained at different resolutions, reflecting spatial or temporal correlations among images. Images belonging to the same content cluster may reside in different geographical or time clusters, and vice versa. For instance, a user may take a set of similar (or dissimilar) pictures at the same location during a certain period of time. Or, a user may have taken a lot of pictures of her dog at various time and locations. As a result, using inter-connected multi-modality clusters makes it easier to capture higher-level image semantics. A user can quickly navigate these clusters by following different types of correlations (similar content, location, time, or ownership) in order to locate the images of interest. In addition, through adaptive user prediction (Section 5), iScope may automatically determine the most promising correlation without explicit user specification.

Clustering large amounts of image data using different types of metadata can be time consuming and memory intensive. To improve efficiency, a hybrid approach is used in which expensive clustering (re)computation is performed on wall-powered server machines when a mobile device synchronizes with a server, and small incremental cluster updates are performed on mobile devices as new images are being added to a data set.

4. Mobile Platform Characterization

This section describes the performance and energy characterization of image search in personal mobile systems.

4.A Measurement Setup

The measurement platform includes a Nokia N810 portable Internet tablet, HP Harrison 6201B direct current power supply, NI-PC-6034E acquisition card, and hosting workstation. iScope has been prototyped on Nokia N810, which is representative of modern personal mobile networked multimedia embedded systems. In particular, N810's 4.3 in LCD touch screen allows the design and evaluation of user-interactive search techniques for personal mobile devices. To measure energy and power consumption, we replace the battery of the mobile platform under test with an HP Harrison 6201B direct-current power supply. Current is computed by measuring the voltage across a 5 W, 250 mΩ, Ohmite Lo-Mite 15FR025 molded silicone wire element resistor in series with the power supply. This resistor was designed for current sensing applications. High-frequency voltage samples are taken using a National Instruments 6034E data acquisition board attached to the PCI bus of a host workstation. The maximum sampling rate of the data acquisition card is 200,000 samples per second. This allows high-resolution power and energy analysis of the mobile system.

4.B Algorithm Characterization

Most content-based search algorithms are computation intensive. This section characterizes the running time of content-based search algorithms on the mobile platform. Table 1 shows the measurement results of the six content-based search algorithms (Section 3) on the mobile platform and a wall-powered Intel quad-core server. The N810's performance is more than an order of magnitude lower than that of the wall-powered server. Therefore, direct use of existing content-based search algorithms in mobile systems would result in high energy consumption and high latencies, thus the need to develop content-based search algorithms for mobile platforms with high energy efficiency and good performance. Table 1 also demonstrates that different search algorithms have dramatically different resource requirements. In iScope, content-based clustering is performed offline, and online content-based search involves only simple cluster lookup operations. As a result, computation time and energy consumption of content-based search is negligible compared with user interactions (see further analysis and results in Section 7). Therefore, more computation-intensive but more accurate algorithms can be selected in order to optimize the overall performance and energy efficiency of the personal content search process.

4.C Hardware Power Characterization

This section presents the power consumptions of the major components in Nokia N810, including the TI OMAP embedded microprocessor, LCD touch screen, and Wi-Fi interface. The results are shown in Table 2. The peak (idle) power consumption of the microprocessor is 0.80 W (0.01 W). The power consumption of the touch screen is 1.04 W or 0.47 W, depending on whether or not it is being touched. The send (receive) power consumption of the wireless interface is 2.00 W (1.76 W). This study shows that the power consumption of the display is comparable to that of the microprocessor and wireless interface. This observation is critical in a user-interactive search process, in which the search system iteratively refines its search results based on user feedback until a satisfactory image is found. During the interactive search process, the energy consumption of human-machine interface components, e.g., the LCD touch screen, can be significant. On the other hand, the energy consumption of the wireless interface must also be carefully

Table 1: Feature Extraction and Search Performance Comparison

Algorithms		Color-32	Color-162	Wavelet	Region-abc	Region-EMD	SIFT
Search (ms)	N810:	2.95	9.74	79.69	250.32	1010.75	652442.40
	Intel quad-core:	0.10	0.25	1.69	9.04	33.97	8647.95
	Ratio:	29.50	38.96	47.15	27.69	29.75	75.44
Extraction (ms)	N810:	311.45	309.03	1137.55	833.50	833.50	32460.27
	Intel quad-core:	5.13	5.06	20.26	32.68	32.68	934.42
	Ratio:	60.71	61.07	56.15	25.50	25.50	34.74

considered during distributed collaborative image search and sharing among multiple mobile devices.

Table 2: Power Consumption (W)

Processor active	Processor idle	Display w/o touch	Display w touch	Wireless send/receive
0.80	0.01	0.47	1.04	2.00/1.76

4.D Time and Energy Characterization of Image Retrieval

Next, we characterize the performance and energy consumption of the image search process. This study helps clarify the time breakdown and energy consumption distribution among the various steps of the image search process. Given an initial query image, users look for a target image using content-based search algorithms through an interactive search process. A performance comparison of the content-based search algorithms is shown in Table 1. The image data set includes approximately 2,000 images gathered using Nokia N810 portable Internet tablets.

Table 3 and Table 4 show the time breakdown and energy consumption distribution of the search process, which has the following components: (1) the initialization stage, including user interface initialization and query image selection; (2) online processing of the content-based search algorithm, including inter-image similarity calculation; and (3) user exploration, including browsing, thinking, and selection. The measured time and energy breakdowns among these three components are 10.4%–18.7%–70.9% and 8.0%–36.6%–55.4%, respectively. Note that, in this study, image similarity is calculated at run time; this could also be done offline. Therefore, the user exploration stage dominates in both latency and energy consumption. This study demonstrates that personal image management and search should focus on reducing the energy consumed in the user exploration stage by minimizing its latency. To this end, iScope employs multi-modality data management and user-centric adaptive search algorithms, which are explained in Sections 5 and 6.

5. User-Centric Adaptive Image Search

This section describes the proposed user-centric image search techniques which leverage content and context information, as well as online adaptive user prediction during image search.

5.A User Interface

One of the main difficulties standing in the way of greater benefit from any intelligent search algorithm is difficulty of use. Most existing browsing-based user interfaces, although inefficient, are straightforward to use. iScope has the goal of making content-based image search accessible to large population of mobile system users spanning different age groups

Table 3: Time Distribution of One Image Search Process

Total time: 80.4 s				
Query dialog 8.4 s		Algorithm computing 15.0 s	User exploration 57.0 s	
Query idle 8.0 s	Query click 0.4 s		Screen idle 53.3 s	Screen click 3.7 s

Table 4: Power Distribution of One Image Search Process

Total energy 52.2 J				
Query dialog 4.2 J		Algorithm computing 19.1 J	User exploration 28.9 J	
Query idle 3.8 J	Query click 0.4 J		Screen idle 25.0 J	Screen click 3.9 J

with different interests and technical backgrounds: an easy-to-use interface is essential.

We have designed a user interface that is accessible to people with no technical background. It supports queries via a straightforward search process. Figure 5 shows the prototype user interface implemented on a Nokia N810 device. The figure on the left shows the starting page, which displays the list of the social group members and the geographical distribution of the image data set. Two types of navigation are supported: (1) navigation across different dimensions, e.g., time, location, content, and ownership, corresponding to the search algorithm's traversal across different dimensions of metadata clustering and (2) zooming in/out along a particular dimension, corresponding to search traversal along a cluster hierarchy. Using this interface, an end user can conduct image search through an interactive navigation process. For instance, using a query image of a person running in Boulder, a user can search for a stadium in Toronto. First, content-based search is used to look for photos with people running. Then, location-based search is used by selecting Toronto on the map to reduce the candidate data set. Manual browsing is then used to find one candidate image containing running people in a stadium. Finally, content-based search is used to search for stadiums in Toronto.

5.B Search Process

To search for an image, a user starts with an existing query image, related context information, or browses in an initial cluster to identify a specific query image. The user then selects a search domain (e.g., content, location, or time) and issues a query. Given the initial query, iScope quickly locates the corresponding cluster that contains the query image in that domain. As described in Section 3, images assigned to the same cluster are similar in a particular domain.

Figure 5: User interface running on Nokia N810. (Left) starting page. (Right) search results with last row based on adaptive prediction, and new query specification.

Promising images can be easily identified by returning other images residing in the query image's cluster. These temporary results are presented to the user, who checks the images' context information and provides feedback on whether they are relevant. The user can then continue the interactive search in two different ways. The user may stay in the same search domain and check the upper-level cluster (for geographical or time clustering) or the neighboring clusters (for content-based clustering). Alternatively, the user may pick one of the positive examples as the new query image and start another query, switching to another search domain if needed. This iterative search process continues until the desired image is located.

All the search steps and user feedback are recorded by iScope and used to tune the automatically-generated clustering structures as follows.

- If an image is selected as the target image or an intermediate target image, it is merged into the same cluster as the query image.
- If the number of images of a cluster exceeds the number of images that can be displayed on the touch screen, the least relevant images will be identified and removed from the original cluster thereby forming a new cluster.
- An empty cluster will be removed from the clustering structure.

5.C Adaptive User Prediction

In addition to explicit user feedback on relevant or irrelevant images, other types of implicit user feedback may also be captured, such as the overall search and navigation path, backtracking operations, etc. This information can be used to guide the run-time learning techniques and provide adaptive user prediction to optimize the user search process. Specifically, iScope makes user-specific prediction based on previous search history, current query image, and intermediate search results, in order to return images that are likely to be of interest.

Our method works as follows. After each round of search, the system records the trace $(q, h_1, h_2, \ldots, h_x, p)$, in which q is the initial query image, $h_1, h_2, \ldots, h_x$ are the intermediate images, and p is the final target image. This image-level trace is then converted to a cluster-level trace, i.e., each image is converted to its corresponding cluster and search domain. Cluster-level traces, instead of image-level traces, are used for prediction because users are unlikely to search for the same image repeatedly, but are likely to search for different images in a cluster (e.g., a specific event or a trip). Given a set of cluster-level traces, at runtime, iScope uses the images selected by the user so far in this round of search as a basis for prediction. Let $(i_1 i_2 i_3)$ be the corresponding clusters. Using Bayes' theorem, we calculate the conditional probability of each candidate cluster C containing the target image:

$$P(C|i_1 i_2 i_3) = \frac{P(i_1 i_2 i_3|C)P(C)}{P(i_i i_2 i_3)}. \quad (1)$$

Since $P(i_1 i_2 i_3)$ is the same for all candidate clusters C, we only need to compute $P(i_1 i_2 i_3|C)P(C)$. Again, using Bayes' theorem, we have

$$P(i_1 i_2 i_3|C) = P(i_1|C)P(i_2|i_1 C)P(i_3|i_1 i_2 C). \quad (2)$$

Using the naïve Bayes probabilistic model, i.e., $i_1 i_2 i_3$ are conditionally independent of each other, we have

$$P(i_1 i_2 i_3|C) = P(i_1|C)P(i_2|C)P(i_3|C). \quad (3)$$

We first locate all the cluster-level traces that contain C, then check how many times i_1, i_2, and i_3 have co-occurred with C in these traces. To compute $P(C)$, we count the number of occurrences of C in all the cluster-level traces O_C, and the total number of cluster occurrences in the traces O. Thus,

$$P(C) = O_C/O. \quad (4)$$

Using the formulas above, we can compute the probability of each candidate cluster containing the target image. In iScope, the two images most frequently used in the image-level traces are then selected from each of the top three clusters. These six suggested images are presented as the bottom row in the search results (see example in Figure 5).

6. Collaborative Image Search

iScope supports collaborative image search targeting distributed mobile environments. The proposed design allows individual users to share their image data sets within their social groups, e.g., friends and family members. It thus allows each member to search a much larger data set than a single mobile device can hold, thereby facilitating information sharing and stimulating social interaction. Previous work has shown that collaborative search utilizing social networks (e.g., friends or social groups) can improve search efficiency and generate more relevant search results [8, 9, 10, 11, 12]. While previous work focused mostly on keyword-based search of Web data, iScope focuses on collaborative content-based and context-based search in distributed mobile systems. Privacy and security are important for data sharing systems. Although this is not the focus of our work, iScope can leverage existing third-party infrastructures for authentication and privacy/data protection [13, 14, 15, 16].

The proposed collaborative search technique conducts parallel search among the socially-associated mobile devices. A search query may be processed by multiple mobile devices, each of which hosts a different image data set manually shared by its owner or automatically cached by the device itself. Each member of a social group shares a subset of her image data set with the whole group. The shared data set is initially stored on her own device and organized separately from the rest of her personal data. More specifically, the metadata management using the proposed multi-modality clustering method is separated from the rest of the owner's personal data for better privacy and security. This approach yields smaller data set, potentially allowing more efficient image search. The distributed shared data sets are ready to support collaborative image search within a social group. After a group member issues a query, her local device conducts local search within her own data set. In the meantime, the query is broadcast to other devices within the social group. Each remote device collaboratively conducts local search within its shared data set and return the results, e.g., metadata and/or the raw images, to the querying device. The user interface for distributed search is identical to that of local search; the remote search process is transparent to the end user.

When a sufficient proportion of data are shared, collaborative search can increase speed relative to local search because the shared data are partitioned and processed in parallel on multiple devices. However, the improvement is bounded by communication overhead. Using the collaborative search scheme, the latency of each remote query is constrained by the communication latency. In addition, collaborative search imposes energy overheads on remote devices. If this overhead is not carefully controlled, individuals may be reluctant to share their data with others, over fear of reduced battery life due to hosting the searches of others.

In this work, we propose an online metadata caching method to minimize the communication overhead of collaborative search. We observed that individual users tend to show more interest in specific subsets of the shared data, and the subsets of interest vary among users. For instance, Alice and her friend Bob took a hiking trip. Alice may be more interested in the photos taken by Bob during the trip than Bob's other shared data. The proposed caching method leverages the "data locality" property, and caches the metadata received remotely at run time, merging the metadata into the user's own data set for future reference. In addition, to support collaborative search, image ownership is introduced as a dimension in the multi-modality data clustering method. When local search requires access to a remote image, it first checks metadata referencing remote storage and then issues a fetch request to the corresponding device, which in turn returns the raw image.

As described in the previous section, metadata clustering is hierarchical. The proposed metadata caching method follows a bottom-up approach, i.e., when all the sub-clusters of a remote cluster has been cached locally, the remote cluster itself is then cached. In addition, each cached remote metadata item and the corresponding cluster also maintain an access history, which tracks how many times, and the most recent time, the corresponding image(s) have been accessed. This information is used to determine the caching policy for the raw images, which are much larger than metadata. When a device has insufficient storage, the raw images with low accesses counts, or long durations since their most recent access, are deleted.

Caching raw images can further speed up the search process and minimize communication energy consumption. However, it raises the concern that a query may result in multiple hits and replies, hence introducing unnecessary network traffic and energy overhead. Our current solution works as follows: when a cache hit occurs, if the locally stored raw image belongs to a remote device, a query and the corresponding image ID are issued to the owner device, which then fetches the image from its local storage (no image search is necessary) and sends the raw image back to the querying device. Our experimental evaluation in Section 7 indicates that the proposed metadata caching method improves system performance and energy efficiency.

7. Experimental Evaluation

In this section, we evaluate iScope, the proposed personalized image management and search system. Section 7.A summarizes the implementation of our prototype and describes the image data sets used in the experiments. Section 7.B evaluates multi-modality data clustering algorithm. Section 7.C evaluates personalized image search on an individual device. Section 7.D evaluates collaborative search in a distributed mobile environment.

7.A Implementation and Image Data Sets

iScope has been implemented on a Nokia N810 device. The multi-modality image data management method, as well as content-based and context-based search techniques are implemented in C and Python. The GTK+ library was used to develop the graphical user interface. The implementation consists of 23,925 lines of C code and 669 lines of Python code.

Sets of images captured using personal portable devices, such as camera phones, are significantly different from general-purpose image data sets. We have constructed an image data set with 7,923 Flickr images captured by six different camera phone users. The Flickr data set is used in the evaluation of content-based search techniques in Section 7.B, because it is more comprehensive (requiring that user study participants gather 8,000 images each would be costly) and this evaluation does not require any context information. However, these Flickr images lack personal context information, such as location and time stamps. In order to evaluate the impact of this context metadata, it was necessary to gather our own image data sets. We developed a software tool for Nokia N810 portable Internet tablets that allows users to manually or automatically take photos using the built-in camera. The software uses the built-in GPS device and clock to tag photographs with locations and timestamps. Ten volunteers took photos during their daily activities. In total, they gathered more than 9,000 images during a period of four months. The images were taken in six cities of three different countries: Canada (Kingston, Ottawa, and Toronto), the United States (Boulder, and San Jose), and the United Kingdom (London). The gathered image data sets, along with the location, time, and ownership information, are stored on N810 devices. They are used to evaluate the impact of distance measurement on content and context clustering quality, as shown in Section 7.B and in the user study shown in Sections 7.C and 7.D.

Table 5: Quality–Speed–Space Comparison of Different Content-Based Search Algorithms

	Search quality		Speed (ms)		Space (feature vectors)	
	Recall	Average precision	Feature extraction	Search	#dimensions	size (byte)
Color-32	0.44	0.25	311.45	2.95	32	128
Color-162	0.46	0.27	309.03	9.74	162	648
Wavelet	0.40	0.32	1137.55	79.69	363	1,452
Region-abc	0.61	0.46	833.50	250.32	14	640
Region-EMD	0.62	0.46	833.50	1010.75	14	640
SIFT	0.43	0.33	32460.27	652442.40	128	84,480

Table 6: Clustering Quality Comparison of the *min*, *max*, and *avg* Distance Measures

	Content			Location			Time		
	min	max	avg	min	max	avg	min	max	avg
InterDist	1200.8	124.2	218.2	46.7	30.7	45.7	720.3 $\times 10^6$	549.8 $\times 10^6$	750.7 $\times 10^6$
IntraDist	0.122	0.454	0.413	0.003	0.004	0.003	81	100	67
InterDist/*IntraDist*	9876.3	273.8	527.8	17928.4	8369.1	15740.1	8.89 $\times 10^6$	5.50 $\times 10^6$	11.20 $\times 10^6$

7.B Multi-Modality Data Management

iScope combines both content-based image features and context metadata to support efficient image data management. We first evaluate the effectiveness of the six content-based algorithms described in Section 3, using the 8,000 Flickr image data set captured by six different users with their camera phones. For each user's subset, we selected 10 query images, and for each query image, we manually identified its set of similar images within the same subset (i.e., taken by the same user). Table 5 compares the quality, speed, and space requirements of the six algorithms. The following quality measures are used: *recall* and *average precision* [6]. Higher quality values indicate better search quality. Region-abc and Region-EMD have the best search quality and are reasonably compact. Region-abc is much faster than Region-EMD due to its simplified distance computation. The combination of multiple content-based features was also experimented, but the search quality was only marginally better than Region-abc. Therefore, Region-abc was used as the representative content-based technique for the other experiments.

According to the results, the *precision* and *recall* for state-of-the-art content-based image retrieval (CBIR) techniques still need to be further improved. iScope, on the other hand, leverages both content-based and context-based techniques to optimize the personal image search. This study demonstrates the importance of leveraging the context information during personal image search.

Next, we evaluate the impact of distance measures on content and context clustering quality using our own image data sets. In our hierarchical clustering algorithm, a distance measure is needed to determine which sub-clusters should be merged. We experimented with three different distance measures: *min*, *max*, and *avg*, which measure the minimum, maximum, and average distances of objects belonging to two different clusters, respectively. A good clustering algorithm should generate clusters that are compact (small intra-cluster distance) and have good separation (large inter-cluster distance). Given a set of k clusters $X_1, X_2, \ldots, X_k$, we define the average intra- and inter-cluster distances.

$$IntraDist = \frac{1}{k} \sum_{1 \le i \le k} \left(\frac{1}{|X_i|} \sum_{x \in X_i} dist(x, \bar{X}_i) \right) \quad (5)$$

$$InterDist = \frac{2}{k(k-1)} \sum_{1 \le i \le j \le k} dist(\bar{X}_i, \bar{X}_j) \quad (6)$$

where $\bar{X}_i$ is the centroid of cluster X_i and $dist()$ is the distance function for two objects. Table 6 shows the quality of different clustering algorithms for content- and context-based clustering. A higher *InterDist* to *IntraDist* ratio indicates better separation and compactness, i.e., better clustering quality. Although the *min* distance measure results in a better ratio, it generates skewed hierarchies with many levels, and is therefore impractical. Instead, we use the *avg* distance measure for our hierarchical clustering algorithms, which has good clustering quality and generates well-balanced hierarchies.

7.C Personalized Image Search

Here, we describe our user study based evaluation of the proposed personalized local image search on the Nokia N810 platform. Ten individuals from Queens University and the University of Colorado at Boulder volunteered for the studies. Most were daily mobile device users with basic computer skills. All the participants were graduate students and between 20–28 years old, two were female and eight were male. It was our goal to compare the performance of different search algorithms (1) when used by individuals on image data sets they had gathered themselves, but also (2) when used on large data sets. Due to practical constraints on user time demands, we designed two studies, each of which had five participants.

In the first study, the image data sets (Section 7.A) were contributed by the same volunteers who would later search for images within them. These data sets contained 1079, 1235, 1497, 1542, and 2100 images, respectively. The size variations resulted from the differences in the rate at which individuals gathered images. Although the sizes of these data sets differ, the study protocol was identical. In the second study, a larger image data set containing 4,389 images drawn from three participants in the first study was used.

For each participant, 30 query images were randomly selected from the corresponding image data set, and the 30 target images corresponding to the query images were then manually specified. It is possible to select a set of images to be targets for one query image. However, using query–target image pairs provides a simple, more deterministic evaluation process of the proposed technique. The time spent by each study participant ranged from four to eight hours. To evaluate the proposed design, we further consider the following cases:

- *Browsing-based search:* To date, browsing is the most

Table 7: Time Usage of Browsing-Based Search

	user 1	user 2	user 3	user 4	user 5	user 6	user 7	user 8	user 9	user 10
Computation time (s)	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
User time (s)	130.5	51.3	41.0	120.9	125.2	449.4	137.1	141.1	595.7	219.0
Overall time (s)	130.8	51.7	41.3	121.3	125.6	449.8	137.5	141.5	596.0	219.4
Avg. steps per image	31.4	40.1	20.2	18.8	86.9	104.7	101.7	111.7	112.7	107.7

Table 8: Energy Usage of Browsing-Based Search

	user 1	user 2	user 3	user 4	user 5	user 6	user 7	user 8	user 9	user 10
Computation energy (J)	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
User energy (J)	68.7	26.3	20.8	58.1	63.6	216.2	69.4	70.9	292.9	114.4
Overall energy (J)	69.1	26.7	21.2	58.5	64.0	216.6	69.9	71.3	293.4	114.9

common image search method in commercial mobile platforms. In this experiment, images are sorted by time. Given a query image, the user searches for the target image by browsing through the image data set.

- *Clustering-based search:* This approach leverages the proposed multi-modality clustering data, content, and context-based search techniques. However, the proposed adaptive user prediction technique is disabled.

In this study, iScope was configured to display 24 (4×6) thumbnail images at a time on the N810 touch screen.

Table 7 and Table 8 show the overall performance and energy consumption, as well as the time and energy usage breakdown, of browsing-based search. As described in Section 4, the time and energy usage of an image search process can be divided into algorithm processing (Computation time) and user's operation (User time) components. Using manual browsing, the time and energy overhead of the search algorithm (image index computation) is negligible. User operations dominate the search process. On average, more than 99% of time and energy is consumed by manual browsing. Table 7 also shows the average number of steps required by each user per image search. The manual browsing based search process is tedious and slow (on average >100 steps per image for each of the five large image sets), resulting in significant time and energy overhead. We conclude that in the image search process, user interaction is the most time and energy consuming stage. Therefore, minimizing the number of required search steps has the greatest potential of for minimizing the time and energy usage.

Figures 6 and 7 compare the time and energy usage of the browsing-based method, clustering-based method, and clustering+prediction (iScope). Compared to the browsing-based method, clustering-based search reduces search time and energy usage by 48.3% and 46.2% (on average), 9.3% and 10.3% (minimum), and 90.5% and 90.0% (maximum). Leveraging the proposed adaptive user prediction technique, iScope further reduces the search time and energy usage by another 22.1% and 21.6% on average, compared to the clustering-based approach. Overall, compared to the browsing based approach, iScope achieves performance improvements of 4.1× (on average), 1.3× (minimum), and 11.1× (maximum). It reduces energy consumption by 3.8× (on average), 1.3× (minimum), and 10.4× (maximum). These experiments also suggest that the benefits of iScope increase when it is used on larger data sets. It enabled 1.9× latency reduction and 2.0× energy reduction when used for a 1,079 image data set and 11.1× latency reduction and 10.4× energy reduction when used for a 4,389 image data set. Note that the user studies were conducted on different volunteers, and the content of different image data sets also vary significantly.

Figure 8 and Figure 9 show the required number of search steps and the average duration of each search step for the three search techniques. The performance improvements and energy savings of iScope are primarily due to the significant reduction in the required number of steps for each image search query. In order to estimate the statistical confidence in our hypotheses about the impact of search algorithm on time and energy, we use the two-tailed Student's t-test. The results of this analysis imply that the mean times for iScope and browsing mode differ with 97.3% probability and that the mean energy consumptions differ with 97.0% probability. Note that the t-test requires some assumptions, e.g., that the variances of the two populations are equal.

The proposed multi-modality clustering and adaptive content and context based searching techniques allow use of the implicit connections between the query and target images, thereby improving search quality and time. Consider the search processes shown in Figure 10. In this case, the query image shows User 3's apartment in Kingston, and the target image shows User 5's apartment in Boulder. Starting from the query image, through context (location), content, and context (location) search operations, User 5 reached an image containing a business building in Boulder. At this point, one context (location) search followed by a predictive content search (done automatically by iScope) was sufficient to reach the desired image. Note that, in this case, even though both the query image and the target image contain similar "content", i.e., apartment, using only the content-based approach would result in an excessively long search process due to the significant differences in color scheme and background content.

This study raises an interesting research question. We have often heard people complaining, "I have seen this somewhere, but just cannot remember where." Recent studies, such as the SenseCam project [17], show that using image recording to enable review of one's daily life can ameliorate human memory loss symptoms. iScope explicitly leverages underlying connections among images. Its use may therefore have the potential to help people strengthen these connections. Currently, we are in the process of evaluating the possibility of applying iScope to related medical applications.

7.D Collaborative Image Search

The distributed, collaborative image search technique described in Section 6 was also evaluated. Communication latency and energy overhead are of primary concern in col-

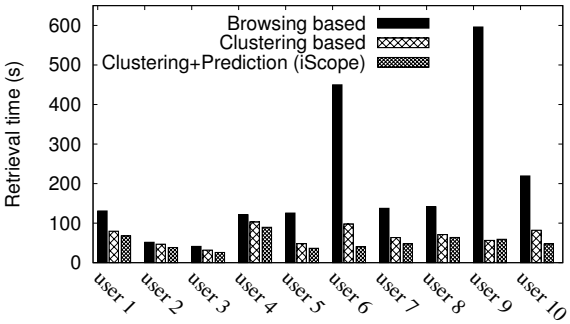

Figure 6: Time comparison of search techniques.

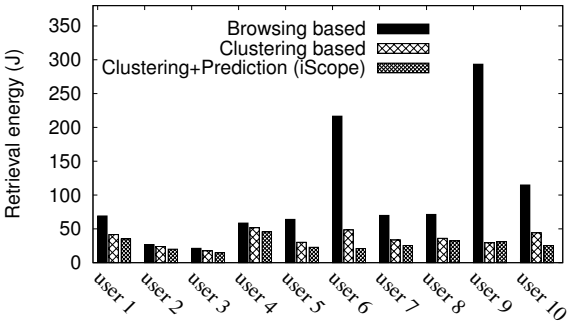

Figure 7: Energy comparison of search techniques.

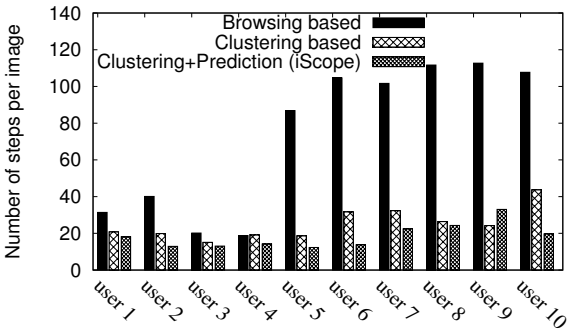

Figure 8: Average number of search per query image.

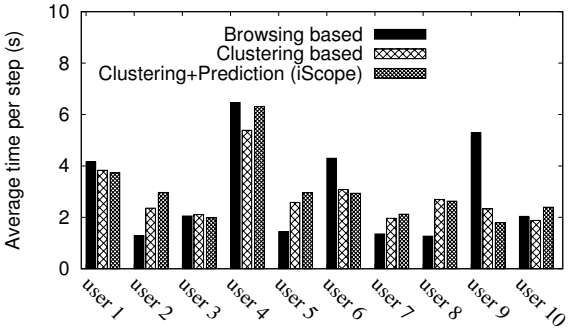

Figure 9: Average time usage per search step.

laborative search. The proposed caching technique aims to minimize these overheads by limiting remote access during collaborative search.

The following experiments consider N810 devices connected via a campus 802.11b network. The user studies described in the previous section were extended to the distributed environment. Detailed image search traces were gathered during the preceeding local search experiments. These traces contain detailed timing information for the interactive image search processes, e.g., the number of search steps of each image query, the time usage of each search step and the breakdown between algorithm processing time and user time. The traces were replayed in the distributed, networked system composed of N810 devices. This technique has the benefits of eliminating ordering effects and random variation between the two studies. It also allows a more direct comparison of local search with distributed collaborative search than would be possible by repeating the study with a new set of users. Timing and system state information was gathered at run time. For instance, networking latency and energy consumption are gathered when remote device accesses are invoked. The power consumption of the N810 in each system state (e.g., receiving data via the 802.11b interface, running a search algorithm, and waiting for user input) was measured using the equipment described in Section 4. These system state dependent power consumption values were used in combination with the timing and system state values measured during trace execution to determine the energy consumption during distributed collaborative search.

We first evaluate the potential communication performance and energy overhead introduced by remote access. In this experiment, the image data set is placed on remote devices and the proposed caching technique is disabled. Therefore, every image search step requires remote device access. Figure 11 and Figure 12 show the energy usage and latency breakdown of the remote search scenario. Compared to image search on a local standalone device, remote image search introduces significant latency and energy overheads. The latency increases by 65.5% on average (27.1% minimum and 96.4% maximum) for the ten participants in user studies. The corresponding total energy consumption increases by 607.5% (275.5% minimum to 877.7% maximum), which includes the energy consumption of the querying device and the remote devices. Note that, since all the remote devices can potentially respond to each query, the worst-case latency and energy overhead increases linearly with the number of mobile devices (four devices are used in this experiment). This study illustrates the importance of reducing the communication overhead during distributed collaborative search.

The proposed caching technique was designed to improve the performance and energy efficiency of collaborative search. More specifically, local caching can reduce the frequency of remote queries, thus minimizing network latency and optimizing overall performance. Caching improves energy consumption for remote devices, because fewer requests require remote processing. It also reduces communication time and energy consumption. However, caching increases the amount of local metadata and raw data, thereby potentially increasing the local costs of search. Since the energy consumption during communication generally dominates that during computation, caching improves the net energy efficiency.

Figures 13 and 14 compare the performance and energy usage of collaborative search with (right bars) and without (left bars) the proposed metadata caching technique. These results demonstrate that metadata caching improves system performance and energy efficiency. When both five-member user studies are considered, the latency reduction is 34.4% on average (18.9% minimum and 43.7% maximum) and the energy consumption reduction is 71.2% on average (59.2%

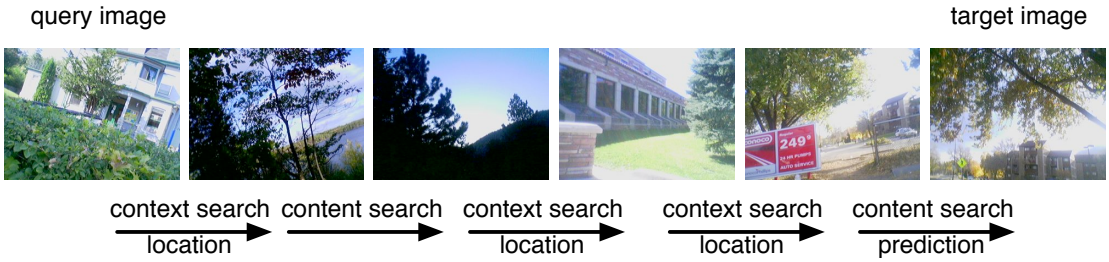

Figure 10: Image search example using content-based and context-based search, and adaptive user prediction.

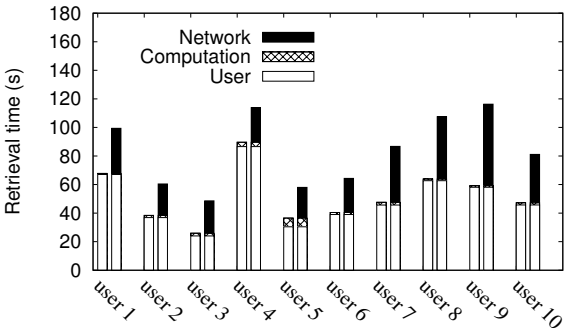

Figure 11: Time breakdown, local (left bars) remote (right bars) retrieval.

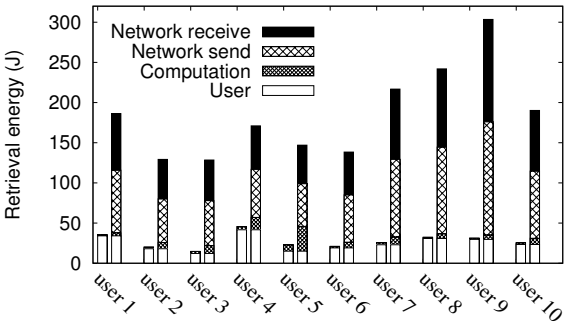

Figure 12: Energy breakdown, local (left bars) remote (right bars) retrieval.

minimum and 78.7% maximum). These performance and energy consumption improvements result from high cache hit rates during the search processes. Table 9 shows the average cache hit rates over the user studies, which average 81% and range from 69% to 90%. This study also demonstrates that the cache hit rate decreases with increasing data set size – image search of a larger data set tends to be more diverse, lowering the cache hit rate. Note that the cache hit rate is affected by the query image distribution within the image data set. In practice, we believe that personalized queries generally have content and/or context correlation, which is reflected as data locality during image search, enabling a high cache hit rate. In contrast, the query images used in this experiment are randomly selected. Therefore, we believe iScope's caching techniques will be even more effective in real usage scenarios. Figure 15 shows the cache hit rate profiles of the ten participants in user studies; the cache hit rate increases during the study for each participant – initially, the local device only contains its own data set and its cache is empty, resulting in a low cache hit rate. As queries are processed, more metadata is cached, improving the cache hit rate.

8. Related Work

Our work draws upon research in several areas concerning image management: content-based image retrieval, multimodality image management, power-aware image retrieval, user feedback, and distributed image sharing. In this section, we survey work most related to our work.

Content-based image retrieval (CBIR) has been an active research area for over a decade [4]. Kim et al. used CBIR for a visual-content recommender [18]. Yeh et al. used mobile images for content-based object search [19]. The Mobile MUVIS project studied content-based image retrieval on mobile devices [20]. CLOVER [21] searches leaf images (sketched or photographed on a mobile device) on a remote server. Photo-to-Search [22] queries the Web directly using images taken on a mobile device. These systems typically use a client-server model where the server does most of the work and the client simply takes queries from the user and displays search results from the server. In this work, we envision more active roles for portable devices.

In addition to the raw content of image data, researchers have also considered other types of information in order to augment image management and search tasks. Fogarty et al. proposed CueFlik, a web image search system [23]. CueFlik allows user-specified search rules based on image visual characteristics to facilitate web image search. Sacco surveys recent research on dynamic taxonomies and faceted search [24]. Jeon and Manmatha used cross-media relevance models for automatic image annotation and retrieval [25]. Yeh et al. used mobile images to search the Web for location recognition through both CBIR and keyword search [26]. Cai et al. [27] describe the use of visual, textual, and link information for clustering Web image search results. MediAssist [28] manages personal image collection based on context, content, and semi-automatic annotation. Wang et al. proposed using multi-modality ontolog for web image retrieval [29]. Veeraraghavan et al. [30] proposed a unifying file system abstraction, called quFile for mobile data management. Location-aware sensing and computation is supported by the Place Lab toolkit [31]. Horozov et al. [32], used location information for recommending pictures to mobile users. Gurrin et al. [33] used GPS information to label images and derive context metadata such as weather and daylight conditions. Jesus et al. [34], used geographical queries to retrieve personal pictures when visiting points of interest. Nokia proposed a sensor-based mobile augment reality system called MARA [35], which utilizes both camera and GPS information. The SenseCam project [36] combines image content with Bluetooth and GPS contextual information to facilitate activity classification. MAMI [37] allows users to annotate and search for digital photos via speech input combined with time, date and location information. Although the past

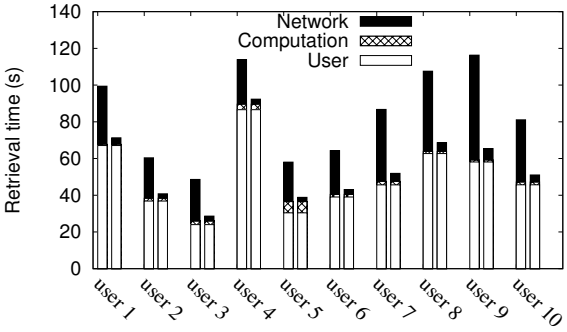

Figure 13: Time breakdown, w. (right bars) w.o. (left bars) caching.

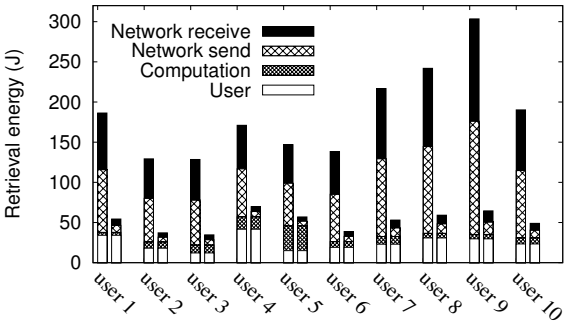

Figure 14: Energy breakdown, w. (right bars) w.o. (left bars) caching.

Table 9: Average Cache Hit Rate for Collaborative Image Search

	user 1	user 2	user 3	user 4	user 5	user 6	user 7	user 8	user 9	user 10	average
Cache hit rate (%)	90	85	84	86	79	69	75	81	85	76	81

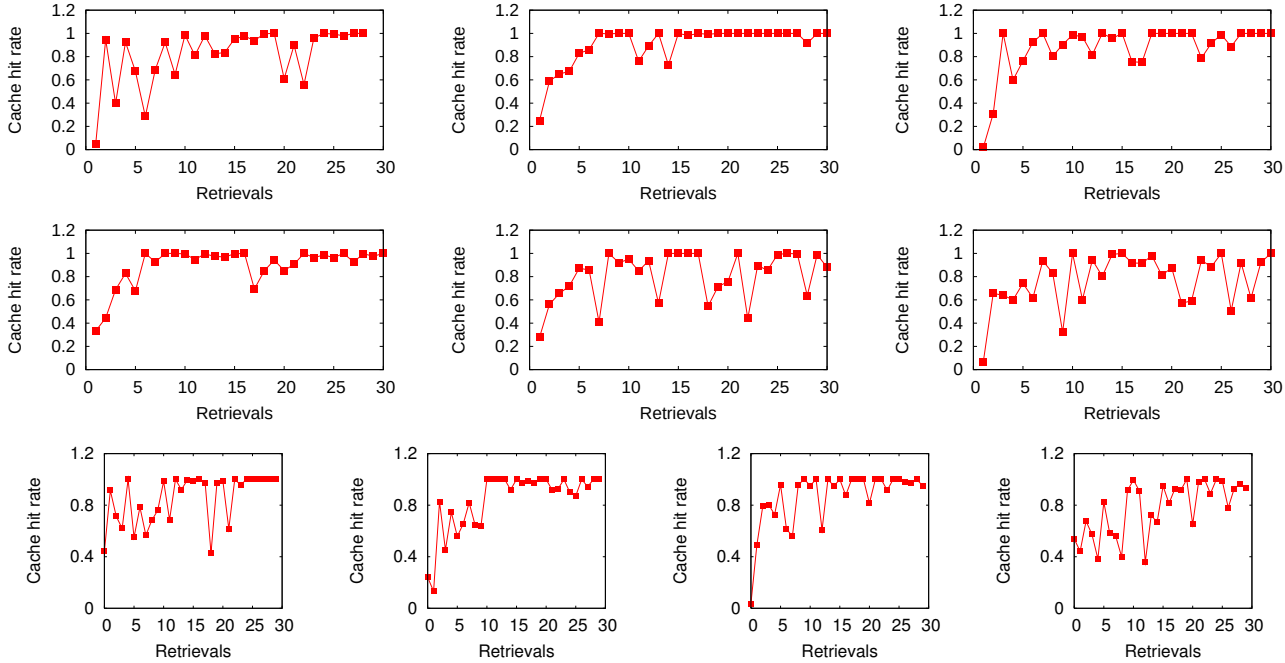

Figure 15: Collaborative search: cache hit rate profile for the ten users.

work utilizes context information, it failed to carefully consider the energy issue, which is the primary constraint of battery-powered systems.

Energy consumption is of primary concern for mobile devices, especially with integrated GPS and sensors. A variety of energy optimization techniques have been proposed for portable devices. Chakraborty et al. evaluated several heuristics for delaying communication via GPS-based movement prediction, in order to reduce energy consumption of wireless communication [38]. Korhonen and Wang studied data packet transmission in WLANs and proposed an energy-saving streaming mechanism in [39]. Karagiannis et al. [40] examined the distribution of inter-contact times between mobile devices, human mobility patterns, and their effect on the performance of forwarding schemes. Recently, Kumar et al. characterized the relationship between query accuracy and energy consumption for CBIR in a mobile system, and proposed an adaptive feature loading scheme to save energy [41]. This work focused on the energy consumption of CBIR processing. However, our study has shown that for image search on mobile devices, power consumption is mainly due to various components such as touch screen and GPS, instead of processor or storage.

Relevance feedback has attracted much attention in the information retrieval community, and has been shown to provide improved performance in many search systems. Yang et al. [42] proposed a semantic feedback mechanism for image retrieval. Hoi et al. [43] proposed a unified framework which incorporates both log data of user's feedback and regular relevance feedback. Liu et al. [44] proposed a new method called relevance aggregation projections (RAP) Most user feedback mechanisms aim at precision/recall improvement

and ignore speed, which is an important factor for performance measurement and energy consumption in mobile systems. Saha et al. [45] presented a human perception based similarity measure along with a relevance feedback indexing scheme. In contrast with past work, our study shows that, in many cases, the adjacent user search steps show little correlation. Therefore, a naive Bayes classifier based prediction algorithm is designed and used for user prediction.

Distributed data sharing for mobile devices has been a popular research topic. One of the early systems is Bayou [46, 47]. Some recent systems have utilized the peer-to-peer model for data sharing and information discovery [48, 49]. Smith et al. has developed AURA, a mobile platform for obtaining object information via bar code scanning and sharing that information with others [50]. Gaonkar et al. [51] presented a system called Micro-Blog, in which geo-tagged multimedia and sensing data can be shared and queried for mobile social collaboration. Pering et al. [52, 53] have studied personal media sharing on mobile devices. Miluzzo et al. [54] developed CenceMe, an application that derives personal activity information using mobile sensing devices and share that information at social networking websites. Context-sensitive information sharing needs were studied in [55]. In our work, a caching technique is proposed to minimize the communication overhead during collaborative search.

In contrast with past work, iScope is a user-centric, energy-efficient personal content management platform. Energy optimization is a primary focus of this work. Our study shows that user interaction and communication dominate system energy consumption. iScope leverages both content and context information, as well as learning techniques, for personalized, energy-efficient image management, search, and sharing.

9. Conclusions and Future Work

User-centric, energy-efficient multimedia content management is of great importance for personal mobile systems. In this work, we have described and evaluated iScope, a user-centric system for personal image data management, search, and sharing on mobile devices. iScope uses new techniques for multi-modality clustering of both content and context information for efficient image data management, as well as user-centric search algorithms with adaptive prediction tailored to individual users. It also supports distributed image sharing and search with online metadata caching. We have implemented a prototype of iScope on networked Nokia N810 portable Internet tablets, and experimentally evaluated it via user studies. Our results show that, on average, iScope improves on the search speed and energy consumption of browsing by 4.1× and 3.8×, respectively. We also found that the use of metadata caching in distributed image search reduces search latency by 34.4% and reduces energy consumption by 71.2%.

The future work includes exploration of more efficient parallel search algorithms to further reduce the communication overhead of collaborative search. In addition, we are interested in determining whether implicit multi-modality search techniques, such as iScope, have the potential to improve human memory or counteract memory loss. Finally, we will further investigate prediction algorithms to incorporate the sequential dependencies of user feedback during personal image search.

10. References

[1] "American museum," http://americanhistory.si.edu/.

[2] "Wikipedia," http://en.wikipedia.org/wiki/Philippe_Kahn/.

[3] "Global mobile forecast to 2012," in *Informa Telecomms & Media Report*, November 2007.

[4] R. Datta, et al., "Image retrieval: Ideas, influences, and trends of the new age," *ACM Computing Surveys*, 2008.

[5] C. E. Jacobs, A. Finkelstein, and D. H. Salesin, "Fast multiresolution image querying," in *Proceedings of SIGGRAPH 95*, August 1995, pp. 277–286.

[6] Q. Lv, M. Charikar, and K. Li, "Image similarity search with compact data structures," in *Proceedings of the 13th ACM Conference on Information and Knowledge Management (CIKM)*, Washington D.C., USA, November 2004, pp. 208–217.

[7] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference on Computer Vision*, September 1999, pp. 1150–1157.

[8] P. A. Chirita, et al., "Search strategies for scientific collaboration networks," in *P2PIR '05: Proceedings of the 2005 ACM workshop on Information retrieval in peer-to-peer networks*, Bremen, Germany, 2005, pp. 33–40.

[9] M. Dalal, "Personalized social & real-time collaborative search," in *WWW*, 2007, pp. 1285–1286.

[10] S. Bao, et al., "Optimizing web search using social annotations," in *WWW '07: Proceedings of the 16th international conference on World Wide Web*, Banff, Alberta, Canada, 2007, pp. 501–510.

[11] B. E. Kolko, E. J. Rose, and E. J. Johnson, "Communication as information-seeking: the case for mobile social software for developing regions," in *WWW*, 2007, pp. 863–872.

[12] P. Heymann, G. Koutrika, and H. Garcia-Molina, "Can social bookmarking improve web search?" in *Proceedings of 1st ACM International Conference on Web Search and Data Mining (WSDM'08)*, 2008.

[13] A. Joels, "RFID security and privacy: A research survey," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 381–393, Feb. 2006.

[14] M. J. Beller, L.-F. Chang, and Y. Yacobi, "Privacy and authentication on a portable communications system," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 6, pp. 821–829, Aug. 1993.

[15] F. M. Heikkila, "Encryption: Security considerations for portable media devices," *Security & Privacy, IEEE*, vol. 5, no. 4, pp. 22–27, July 2007.

[16] A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensor networks," *Communications of the ACM*, vol. 47, no. 6, pp. 53–57, June 2004.

[17] E. Berry, et al., "The use of a wearable camera, sensecam, as a pictorial diary to improve autobiographical memory in a patient with limbic encephalitis: A preliminary report."

[18] C. Y. Kim, et al., "VISCORS: A visual-content recommender for the mobile web," *IEEE Intelligent Systems*, vol. 19, no. 6, pp. 32–39, 2004.

[19] T. Yeh, et al., "A picture is worth a thousand keywords: Image-based object search on a mobile platform," in *Proc. of the Conference on Human Factors in Computing Systems (CHI)*, April 2005, pp. 2025–2028.

[20] I. Ahmad, et al., "Content-based image retrieval on mobile devices," in *Proceedings of SPIE (Multimedia on Mobile Devices)*, vol. 5684, San Jose, CA, USA, January 2005.

[21] Y. Nam, E. Hwang, and D. Kim, "CLOVER: mobile content-based leaf image retrieval system," in

MULTIMEDIA '05: Proceedings of the 13th annual ACM Intl. Conf. on Multimedia, 2005, pp. 215–216.

[22] M. Jia, et al., "Photo-to-Search: Using camera phones to inquire of the surrounding world," in *MDM '06: Proceedings of the 7th International Conference on Mobile Data Management*, 2006, pp. 46–46.

[23] F. James, et al., "Cueflik: interactive concept learning in image search," in *Conference on Human Factors in Computing Systems'08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, Apr. 2008, pp. 29–38.

[24] G. M. Sacco, "Research results in dynamic taxonomy and faceted search systems," in *DEXA '07: Proceedings of the 18th International Conference on Database and Expert Systems Applications*, 2007, pp. 201–206.

[25] J. Jeon, V. Lavrenko, and R. Manmatha, "Automatic image annotation and retrieval using cross-media relevance models," in *SIGIR*, 2003, pp. 119–126.

[26] T. Yeh, K. Tollmar, and T. Darrell, "Searching the web with mobile images for location recognition," in *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, July 2004, pp. 76–81.

[27] D. Cai, et al., "Hierarchical clustering of WWW image search results using visual, textual and link information," in *12th annual ACM international conference on Multimedia*.

[28] N. O'Hare, et al., "Using text search for personal photo collections with the MediAssist system," in *SAC '07: Proceedings of the 2007 ACM symposium on Applied computing*, 2007, pp. 880–881.

[29] H. Wang, L.-T. Chia, and S. Liu, "Image retrieval ++– web image retrieval with an enhanced multi-modality ontology," *Multimedia Tools Appl.*, vol. 39, no. 2, pp. 189–215, Nov. 2008.

[30] K. Veeraraghavan, et al., "quFiles: a unifying abstraction for mobile data management," in *HotMobile '08: Proceedings of the 9th workshop on Mobile computing systems and applications*, 2008, pp. 65–68.

[31] T. Sohn, et al., "Experiences with place lab: an open source toolkit for location-aware computing," in *ICSE '06: Proceedings of the 28th international conference on Software engineering*, 2006, pp. 462–471.

[32] T. Horozov, N. Narasimhan, and V. Vasudevan, "Using location for personalized POI recommendations in mobile environments," in *SAINT '06: Proceedings of the International Symposium on Applications on Internet*, 2006, pp. 124–129.

[33] C. Gurrin, et al., "Mobile access to personal digital photograph archives," in *Proc. ACM Int. Conf. Human Computer Interaction with Mobile Devices and Services*, 2005, pp. 311–314.

[34] R. Jesus, et al., "Geographic image retrieval in mobile guides," in *GIR '07: Proc. of the 4th ACM workshop on Geographical information retrieval*, Nov. 2007, pp. 37–38.

[35] M. Kahari and D. J. Murphy, "MARA, sensor based augmented reality system for mobile imaging device," in *ISMAR*, 2006.

[36] D. Byrne, et al., "Using bluetooth & GPS metadata to measure event similarity in sensecam images," 2007.

[37] X. Anguera, J. Xu, and N. Oliver, "Multimodal photo annotation and retrieval on a mobile phone," in *MIR '08: Proceeding of the 1st ACM international conference on Multimedia information retrieval*, 2008, pp. 188–194.

[38] S. Chakraborty, et al., "On the effectiveness of movement prediction to reduce energy consumption in wireless communication," *IEEE Transactions on Mobile Computing*, vol. 5, no. 2, pp. 157–169, 2006.

[39] J. Korhonen and Y. Wang, "Power-efficient streaming for mobile terminals," in *NOSSDAV '05: Proceedings of the international workshop on Network and operating systems support for digital audio and video*, 2005, pp. 39–44.

[40] T. Karagiannis, J.-Y. L. Boudec, and M. Vojnović, "Power law and exponential decay of inter contact times between mobile devices," in *MobiCom '07: Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, 2007, pp. 183–194.

[41] K. Kumar, et al., "Energy conservation by adaptive feature loading for mobile content-based image retrieval," in *Proceedings of the thirteenth international symposium on Low power electronics and design*, 2008, pp. 153–158.

[42] C. Yang, M. Dong, and F. Fotouhi, "Semantic feedback for interactive image retrieval," in *MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*. ACM, 2005, pp. 415–418.

[43] S. C. H. Hoi, M. R. Lyu, and R. Jin, "A unified log-based relevance feedback scheme for image retrieval," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, pp. 509–524, 2006.

[44] W. Liu, W. Jiang, and S.-F. Chang, "Relevance aggregation projections for image retrieval," in *CIVR '08: Proceedings of the 2008 international conference on Content-based image and video retrieval*. New York, NY, USA: ACM, 2008, pp. 119–126.

[45] S. K. Saha, A. K. Das, and B. Chanda, "Image retrieval based on indexing and relevance feedback," *Pattern Recogn. Lett.*, vol. 28, no. 3, pp. 357–366, 2007.

[46] A. J. Demers, et al., "The bayou architecture: Support for data sharing among mobile users," in *Proceedings of the Workshop on Mobile Computing Systems and Applications*, Santa Cruz, CA, USA, December 1994, pp. 2–7.

[47] D. B. Terry, et al., "Managing update conflicts in bayou, a weakly connected replicated storage system," in *Proceedings 15th Symposium on Operating Systems Principles (SOSP)*, Cooper Mountain, CO, USA, December 1995, pp. 172–183.

[48] S. Sobti, et al., "Segank: A distributed mobile storage system," in *Proceedings of the 3rd USENIX Conference on File and Storage Technologies (FAST)*, March 2004.

[49] D. Peek and J. Flinn, "Ensemblue: integrating distributed storage and consumer electronics," in *OSDI '06: Proceedings of the 7th symposium on Operating systems design and implementation*, 2006, pp. 219–232.

[50] M. A. Smith, D. Davenport, and H. Hwa, "AURA: A mobile platform for object and location annotation," in *UbiComp*, 2003.

[51] S. Gaonkar, et al., "Micro-Blog: sharing and querying content through mobile phones and social participation," in *MobiSys '08: Proceeding of the 6th international conference on Mobile systems, applications, and services*, 2008, pp. 174–186.

[52] T. Pering, et al., "Face-to-face media sharing using wireless mobile devices," in *ISM '05: Proceedings of the Seventh IEEE Int. Symposium on Multimedia*, 2005, pp. 269–276.

[53] T. Pering, et al., "Musicology: Bringing personal music into shared spaces." in *MobiQuitous*, 2007, pp. 1–8.

[54] E. Miluzzo, et al., "Sensing meets mobile social networks: the design, implementation and evaluation of the cenceme application," in *SenSys '08: Proceedings of the ACM conf. on Embedded network sensor systems*, 2008, pp. 337–350.

[55] D. Dearman, M. Kellar, and K. N. Truong, "An examination of daily information needs and sharing opportunities," in *CSCW '08: Proceedings of the ACM 2008 conference on Computer supported cooperative work*, 2008, pp. 679–688.