JIGSAW: Interactive Mobile Visual Search with Multimodal Queries*

Yang Wang [†], Tao Mei [‡], Jingdong Wang [‡], Houqiang Li [†], Shipeng Li [‡]
[†] University of Science and Technology of China, Hefei 230027, P. R. China
[‡] Microsoft Research Asia, Beijing 100080, P. R. China
wyang1@mail.ustc.edu.cn; {tmei, jingdw, spli}@microsoft.com; lihq@ustc.edu.cn

ABSTRACT

The traditional text-based visual search has not been sufficiently improved over the years to accommodate the new emerging demand of mobile users. While on the go, searching on one's phone is becoming pervasive. This paper presents an innovative application for mobile phone users to facilitate their visual search experience. By taking advantage of smart phone functionalities such as *multi-modal* and *multi-touch* interactions, users can more conveniently formulate their search intent, and thus search performance can be significantly improved. The system, called JIGSAW (Joint search with ImaGe, Speech, And Words), represents one of the first attempts to create an interactive and multi-modal mobile visual search application. The key of JIGSAW is the composition of an exemplary image query generated from the raw speech via multi-touch user interaction, as well as the visual search based on the exemplary image. Through JIGSAW, users can formulate their search intent in a natural way like playing a jigsaw puzzle on the phone screen: 1) a user speaks a natural sentence as the query, 2) the speech is recognized and transferred to text which is further decomposed to keywords through entity extraction, 3) the user selects preferred exemplary images that can visually represent his/her intent and composes a query image via multi-touch, and 4) the composite image is then used as a visual query to search similar images. We have deployed JIGSAW on a real-world phone system, evaluated the performance on one million images, and demonstrated that it is an effective complement to existing mobile visual search applications.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information search and retrieval—*Search process*; H.5.2 [**Information Interfaces and Presentation**]: User interface—*User-centered design*

General Terms

Algorithms, Experiments, Human Factors

Keywords

Mobile visual search, query formulation, user interface.

1. INTRODUCTION

While on the go, consumers use their phones as a personal Internet-surfing concierge. Searching is becoming pervasive and is one of the most popular applications on mobile phones. People are more and more addicted to conducting searches on their phones. It is reported that one-third of search queries will come from smart phones by 2014 [1]. However, compared with text and location search by phone, visual (image and video) search is still not that popular, mainly because the user's search experience on the phone is not always enjoyable. On one hand, existing forms of queries (i.e., text or voice as queries) are not always user-friendly—typing is a tedious job, and voice cannot express visual intent well. On the other hand, the user's intent in a visual search process is somewhat complex and may not be easily expressed by a piece of text (or text transferred from voice). For example, as shown in Fig. 1(a), the query like "find a picture of a person with a straw hat and a spade" will most likely not result in any relevant search results from existing mobile search engines.

To facilitate visual search on mobile devices, the work described in this paper aims at a more natural way to formulate a visual query, taking full advantage of multi-modal and multi-touch interactions on mobile devices. As shown in Fig. 1(c), users can easily formulate a composite image as their search intent by naturally interacting with the phone through voice and multi-touch. Although similar applications exist, such as Goggles [12], iBing, and SnapTell [26], which support photo shots (using the built-in camera) as a visual query for instant search, as shown in Fig. 1(b), our work represents a complementary mobile visual search by which users can compose an arbitrary visual query (not necessarily an existing image) through natural user interaction.

It is known that visual search on a mobile device is different from that on a desktop. Compared with a desktop PC which predominantly supports text-to-search mode, a mobile phone provides a richer set of user interactions and thus achieves a more natural search experience. For example, beyond the traditional keyboard and mouse inputs, mobile phones are usually enabled to receive multi-modal inputs. The most common interface of this kind combines a

* This work was performed at Microsoft Research Asia.
Area chair: Daniel Gatica-Perez.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
MM'11, November 28–December 1, 2011, Scottsdale, Arizona, USA.
Copyright 2011 ACM 978-1-4503-0616-4/11/11 ...$10.00.

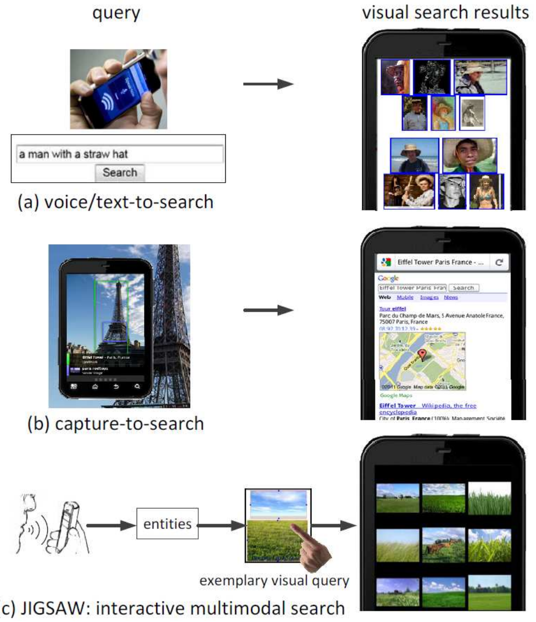


Figure 1: The main modes for mobile visual search: (a) voice/text-to-search, (b) capture-to-search, (c) JIGSAW (this work).

visual modality via the built-in camera with a voice modality via speech recognition. In addition, the multi-touch phone screen, which recognizes multiple simultaneous touch points, provides rich interaction between users and devices. All these advantages provide for a more natural interaction to formulate search intent and thus achieve a better search experience via mobile phone.

There exist some visual search applications for mobile devices. Table 1 is a survey of the recent visual search applications on various mobile platforms. All of them require users to first take a picture and then perform similar image searches in various vertical domains (i.e., capture-to-search mode). However, in many cases, the user's search intent is implicit and cannot be represented through capturing the surroundings. The user, nevertheless, can express his/her intent via a piece of voice description. For example, a user is looking for a restaurant with a red door and two stone lions in front of the door, however s/he forgot the name of the restaurant. Therefore, a client-side tool that can transfer a long textual query into a visual query with user interaction is required to determine the restaurant's name and location.

Although researchers in the computer vision community have proposed various techniques for mobile visual search [4] [5] [27] [32], most of them focus on the problems with capture-to-search mode. For example, Takacs and Yang *et al.* discuss how to represent the captured image with a set of visual descriptors [27] [32]. The transmission of image descriptors between mobile client and server is discussed in [5]. Chandrasekhar *et al.* focus on how to compact visual descriptors using compression techniques [4]. Different from many existing works, this paper studies the problem of visual query formulation on mobile devices, as well as the key techniques in this new search mode. The work is an effective complement to existing mobile visual search systems.

The proposed system, called JIGSAW (Joint search with ImaGe, Speech, And Words), takes advantage of the functionalities on mobile phones, such as multi-modal and multi-touch interactions, to help users formulate their (implicit) search intent more conveniently and thus promote visual search performance. The search procedure consists of the following phases: 1) the user speaks a natural sentence as the query to the phone, 2) the speech is recognized and transferred to text, and the text is further decomposed into keywords by entity extraction, 3) the user selects the preferred exemplary images (given by an image clustering process according to the entities) that can visually represent each keyword and composes a query image through multi-touch, 4) the composed image is then used as a visual query to search similar images. Therefore, the key component of JIGSAW is the composition of an exemplary image query generated from the raw speech via multi-touch user interaction, as well as visual search based on the exemplary image. Through JIGSAW, users can formulate their visual search intent in a natural way like piecing together a jigsaw puzzle on the phone screen. The techniques in JIGSAW include speech recognition, entity extraction, image clustering, large-scale image search, and user interaction.

Table 1: Recent mobile visual search applications.

App	Features
Goggles [12]	product, cover, landmark, namecard
Digimarc Discover [8]	print, article, ads
Point and Find [22]	place, 2D barcode
SnapTell [26]	cover, barcode
SnapNow [16]	MMS, email, print, broadcast
Kooaba [25]	media cover, print

Our contributions are twofold: 1) we propose an interactive and multi-modal visual search system on mobile devices, which takes advantages of natural multi-modal and multi-touch functionalities on the phone, and 2) we propose a context-aware approach to similar image search which takes the spatial relation among exemplary image patches into consideration.

The rest of this paper is organized as follows. Section 2 reviews related work for mobile and desktop visual searching. Sections 3 and 4 describe each component and implementation of JIGSAW. Experiments are presented in Section 5, followed by conclusions in Section 6.

2. RELATED WORK

We review related research on mobile visual search and interactive visual search.

2.1 Visual Search on Mobile

Directly applying keyword-based search to mobile visual search is straightforward yet intrusive. As we have mentioned, typing a long query is not always user-friendly on mobile devices. This is the reason that mobile users type on average 2.6 terms per search [6], which can hardly express their search intent. Compared with text-to-search, capture-to-search is becoming dominant in mobile visual search. It is more convenient for mobile users to take a photo and use it to search on the go. Goggles [12], Point and Find [22], and Snaptell [26], are recent visual search applications in this area.

There exist efforts on mobile visual search in the computer vision community. Most of these efforts have focused on the exploration of different visual descriptors. For example, local features such as Scale-Invariant Feature Transform

(SIFT) feature [18], MPEG-7 image signature, and Speeded Up Robust Feature (SURF) [2] have been devised to handle luminance and geometry variances. In addition to the exploration of visual descriptors, Chandrasekhar *et al.* discuss the issue of compressing these descriptors, so as to reduce the bandwidth and storage costs on mobile devices [4]. They propose to use Compressed Histogram of Gradients (CHoG) to quantize and encode gradient histograms with Huffman and Gagie trees to achieve a very low bit-rate transmission. It demonstrates that SIFT has the advantage over CHoG and MPEG-7 image signature in a system of CD cover search.

The proposed JIGSAW in this paper differs from existing mobile visual search systems in that it represents a new search mode in which mobile users can naturally formulate a visual query on the go.

2.2 Interactive Visual Search on PC

The most related work on generic visual search to JIGSAW is interactive search, in which users specify their search intent interactively. The advanced functionalities in Google and Bing's image search engines enable user to indicate search intent via various filters, e.g., "similar images," color, style, face, and so on. Tineye supports the uploading of an exemplary image as a query example for search [28], while Xcavator even enables users to emphasize certain regions on the query image as the key search components [30]. In a more advanced search engine prototype, such as GazoPa [11] and MindFinder [3], the search is performed by sketching a shape image. The "Concept Map" uses the position and size of a group of tags to filter the top text-based search results [31], while the "Color Map" enables the selection of multiple color hints on a composite canvas as a visual query [29]. However, user interaction on desktop is not as natural as that on mobile device. Therefore, an interactive mobile visual search system which takes advantage of multi-touch and multi-modal functionalities is desirable.

3. JIGSAW

JIGSAW is an interactive mobile visual search application that enables users to naturally formulate their search intent in an interactive way and combines different visual descriptors (e.g., SIFT, color, and edge) for visual search. Figure 2 shows the framework of JIGSAW. On the client-side, a user first speaks a natural sentence to initiate a voice query, e.g., a sentence like "find an iron tower on the grass." On the cloud-side, the system employs speech recognition (SR) to transfer the speech to a piece of text, and then extracts entities from the text. As a result, "tower" and "grass" are recognized as two entities that can be represented by two exemplary images. Directly using those entities as textual queries may not return relevant results, as it only searches the surrounding text and neglects the position and size of these exemplary images on the query canvas. Therefore, we propose to enable users to further specify search intent by touching the screen and dragging their preferred exemplary images, and then formulating a composite visual query. Those exemplary images are automatically generated using a clustering process according to the extracted entities. Finally, we exploit both the text and the composite visual query to search for relevant images, by considering the position and the size of the exemplary images. In the next sections, we will describe the details of each component.

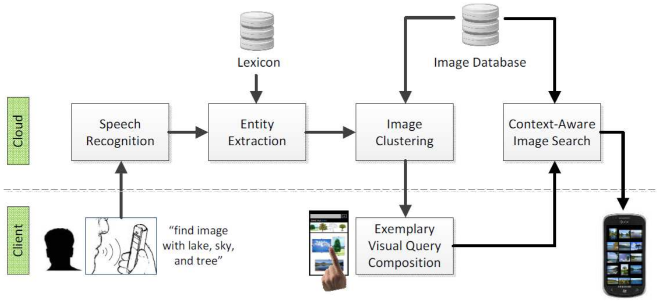

Lexicon
Image Database
Cloud
Speech Recognition
Entity Extraction
Image Clustering
Context-Aware Image Search
Client
"find image with lake, sky, and tree"
Exemplary Visual Query Composition

Figure 2: JIGSAW architecture. There are four major components: (1) speech recognition, (2) entity extraction, (3) interactive exemplary visual query composition, and (4) context-aware example-based image search.

3.1 Speech Recognition and Entity Extraction

On the mobile phone, voice is more natural than typing for users. We propose to leverage voice input to help users initiate a query. We employ a Hidden Markov Model (HMM)-based SR engine which is able to handle natural sentences and phrase fragments, and then translate the speech into text [23]. In general, speech recognition engines are usually constructed in a statistical modeling framework, in which the speech utterance will first be transformed into a compact and meaningful representation in the feature space. Then, the decoder takes the feature vectors as input and generates the probability of the hypothesized word sequence based on the acoustic and language models. In JIGSAW, we employed the SR service from a commercial local search engine. However, any state-of-the-art SR engine could be used in this capacity.

The text result from the speech recognition can be directly used as the query for image search. However, it is well-known that existing search engines cannot handle a long query very well. On the other hand, understanding a long sentence is still challenging. For example, issuing a textual query like "find an image with several green trees in front of a white house" may not result in any relevant search. Therefore, we process the original recognized text to extract entities (i.e., keywords like "tree" and "house") for a better visual search. Although general entity extraction is still an open problem despite many efforts [15], the entity extraction in JIGSAW can be reduced to detect the words that can be represented by some exemplary images, so that users can select their preferred images. Therefore, we focus on the detection of meaningful noun words/phrases, such as "building," "car," and "tree," while discarding the vague and general words/phrases like "law" and "holiday." To this end, an entity dictionary is constructed according to WordNet [20] by collecting the noun words which have concrete visual representations (117,798 nouns out of 155,287 words). The judgement on whether an noun has concrete visual representations is based on whether the noun can be represented by the images in the ImageNet [7]. By removing the nouns which have less than 100 images in the ImageNet, 22,117 unique words are kept. In addition, we include other entity names including celebrities, popular products, and landmarks to deal with the names such as "Super man" and "Eiffel." These entity names are obtained by mining the web and the top queries in a commercial search engine [24].

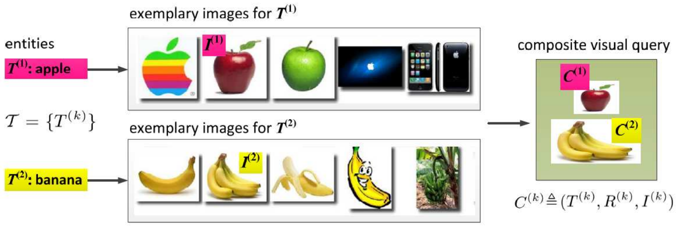

exemplary images for $T^{(1)}$
entities
$T^{(1)}$: apple
$I^{(1)}$
$\mathcal{T} = \{T^{(k)}\}$
exemplary images for $T^{(2)}$
$T^{(2)}$: banana
$I^{(2)}$
composite visual query
$C^{(1)}$
$C^{(2)}$
$C^{(k)} \triangleq (T^{(k)}, R^{(k)}, I^{(k)})$

Figure 3: An example for composing a visual query with multiple components $C^{(k)}$ from an initial query $\mathcal{T} = \{T^{(k)}\}_{k=1}^{K}$ with multiple entities $T^{(k)}$. Each entity corresponds to one component in the composite visual query. There are two recognized entities: $T^{(1)} =$ "apple," and $T^{(2)} =$"banana". For each entity, there is a list of candidate exemplary images to be selected by users. A composite visual query is generated by the selected image $I^{(1)}$ and $I^{(2)}$, where the position and the size ($R^{(1)}$ and $R^{(2)}$) of each component on the canvas are adjusted by the user.

Finally, words in the voice query will be assigned to their longest match. For example, "polar bear" and "Eiffel Tower" are phrases and thus cannot be split. If a user says "find an iron tower under grass," then the extracted keywords would be "tower" and "grass." These keywords are used independently in the following image clustering and search steps.

3.2 Interactive Formulation of Composite Visual Query

Searching for images with a single entity in a text-based search engine can return related results. But it is still challenging for the text-based image search to consider the spatial relation among different entities (i.e., both the position and the size of the entities in the composite visual query). It would be difficult to automatically arrange the entities (i.e., their corresponding exemplary images) on the query image canvas. Therefore, we propose to take advantage of the screen-touch function to enable users to composite a visual query via multi-touch interactions. For each entity, the system returns a set of representative images. Users can select one image per entity and drag onto the composite image canvas, as shown in Fig. 1(c). When users finish selecting the exemplary images, they can formulate their visual queries by adjusting the position and the size of each exemplary image on the composite canvas via multi-touch interactions. JIGSAW is designed for addressing the following ambiguities in the existing text-based image search systems: *polysemy*, *aspect*, *view point*, *position*, and *attributes*.

Polysemy means that a word has multiple meanings, such as apple (fruit or product), football (association football or American football). Aspect indicates that a word may have different concepts, such as apple (company or product), football (ball or game). View point means an object could have various appearances from different angles or perspectives, such as a car (side or front view), or an office (inner or outer). Position indicates in which position the object is expected within the target image. Attribute defines the properties of an entity, such as color, type, and decoration. All these ambiguities lead to difficulties in expressing search intent.

In this paper, we are investigating whether these ambiguities can be solved by introducing some user interactions.

Table 2: Key notations.

I	exemplary image
J	target image to be searched
R	rectangle region for a component on the canvas of the composite visual query
T	entity (i.e., keyword)
C	$C \triangleq (T, I, R)$, component in a composite visual query
K	# of entities
k	index of components ($k = 1, \ldots, K$)
$\boldsymbol{f}$	feature vector of an image
$\boldsymbol{h}$	feature vector of an image grid
$C^{(k)}$	$C^{(k)} \triangleq (T^{(k)}, I^{(k)}, R^{(k)})$, the k-th component
(i, j)	index of a grid in the target image J
$R_J^{(k)}$	$R_J^{(k)} = \bigcup_{i,j \in R^{(k)}} (i, j)$, the union of the grids in J
$e_J^{(k)}(i, j)$	visual similarity between $I^{(k)}$ and image region $R_J^{(k)}$
$d^{(k)}(i, j)$	user intent map for k-th component at (i, j)
$r_J^{(k)}$	relevance between composite query and target image J in terms of k-th component
r_J	overall relevance between composite query and target image J

Based on the above analysis of ambiguities, we introduced the interactive composition of a visual query by manipulating the multiple exemplary images on a given canvas. Specifically, when the user issues a voice query, the system will recognize a set of entities (keywords) $\mathcal{T} = \{T^{(k)}\}_{k=1}^{K}$, and return a list of exemplary images for each entity, as shown in Figure 3, where $T^{(k)}$ indicates one entity and K is the number of entities in the textual query $\mathcal{T}$. By selecting the desired exemplary images, as well as re-positioning and resizing them on the canvas of the composite query image, the user can formulate a visual query. Therefore, the composite visual query can be represented as a set of components $\mathcal{C} = \{C^{(k)}\}_{k=1}^{K}$, where each component $C^{(k)}$ corresponds to one entity $T^{(k)}$, as well as the selected exemplary image $I^{(k)}$ and the position and size $R^{(k)}$ of this image indicated by the user. Thus, $C^{(k)}$ can be further represented by a triplet $(T^{(k)}, I^{(k)}, R^{(k)})$. Table 2 lists the key notations.

3.3 Generation of Exemplary Images

In this section, we present an automatic approach to generating the exemplary images for each entity, by exploiting both ImageNet and image search engines results. It is impractical to allow users to manually select the exemplary images from a number of images (e.g., the top image search results from search engines). It is also not appropriate to directly use the images from ImageNet because of the cross-domain difference between ImageNet and general web images. Thus, we propose a clustering-based approach to generating the exemplary images for a given entity. The next subsections will introduce the visual features and similarity metric for clustering.

3.3.1 Visual descriptors

We adopt three types of visual features, including SIFT [18], color histogram, and gradient histogram, which have been proven to be effective for image retrieval. Since the local descriptor like SIFT may not perform well for some images, such as scene and human, we exploit the discriminative power of color and gradient histogram. For each image, a 128-dimensional SIFT descriptor is extracted at each key

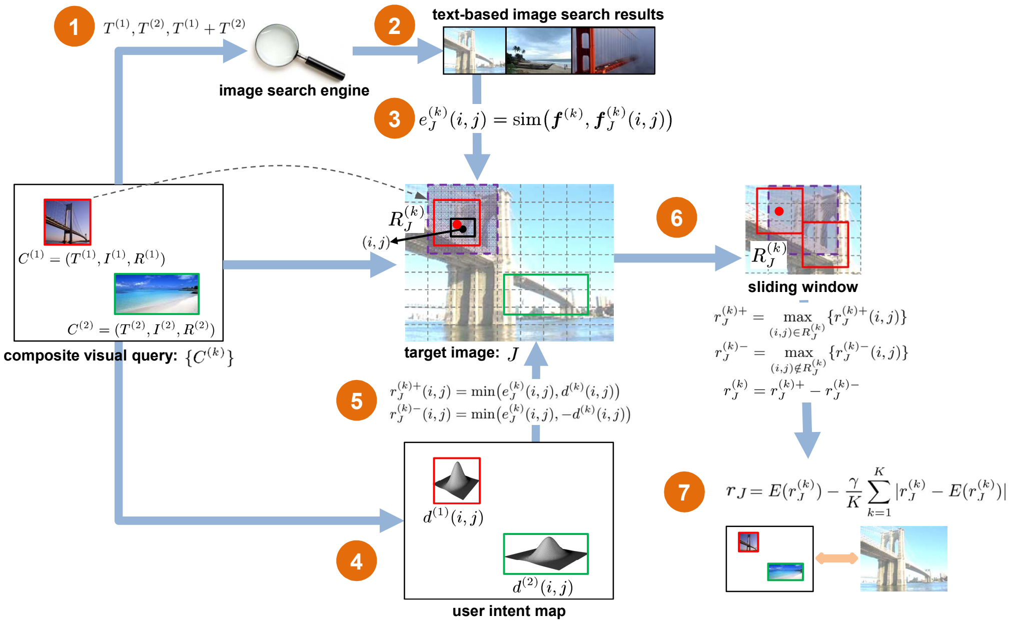

1
$T^{(1)}, T^{(2)}, T^{(1)} + T^{(2)}$
image search engine
2
text-based image search results
3
$e_J^{(k)}(i,j) = \mathrm{sim}\big(\boldsymbol{f}^{(k)}, \boldsymbol{f}_J^{(k)}(i,j)\big)$
$R_J^{(k)}$
(i,j)
$C^{(1)} = (T^{(1)}, I^{(1)}, R^{(1)})$
$C^{(2)} = (T^{(2)}, I^{(2)}, R^{(2)})$
composite visual query: $\{C^{(k)}\}$
target image: J
6
$R_J^{(k)}$
sliding window
$r_J^{(k)+} = \max_{(i,j)\in R_J^{(k)}} \{r_J^{(k)+}(i,j)\}$
$r_J^{(k)-} = \max_{(i,j)\notin R_J^{(k)}} \{r_J^{(k)-}(i,j)\}$
$r_J^{(k)} = r_J^{(k)+} - r_J^{(k)-}$
5
$r_J^{(k)+}(i,j) = \min\big(e_J^{(k)}(i,j), d^{(k)}(i,j)\big)$
$r_J^{(k)-}(i,j) = \min\big(e_J^{(k)}(i,j), -d^{(k)}(i,j)\big)$
7
$r_J = E(r_J^{(k)}) - \frac{\gamma}{K}\sum_{k=1}^{K} |r_J^{(k)} - E(r_J^{(k)})|$
$d^{(1)}(i,j)$
$d^{(2)}(i,j)$
4
user intent map

Figure 4: The process for computing the relevance r_J between the composite visual query and a target image.

point. Then, a vocabulary tree is constructed by hierarchical K-means [21], yielding about 6,000 visual words. Each 128-dimensional SIFT descriptor is hashed into a visual word by traversing the tree. Finally, an image is described by the weighted visual words. Color is also quantized to 192 bins in the HSV space, while the gradient is quantized to 8 directions and 8 intensities, yielding a 64-dimensional gradient histogram. As a result, an image is described by the concatenated histogram ($6256 = 6000 + 192 + 64$) of these three kinds of features. It should be noted that these features are normalized individually before the concatenation.

3.3.2 *Similarity metric*

Let $\boldsymbol{f}_i$ and $\boldsymbol{f}_j$ denote the normalized histograms of image i and j, respectively. The similarity between these two images is given by a weighted intersection kernel between two histograms

$$\mathrm{sim}(\boldsymbol{f}_i, \boldsymbol{f}_j) = \sum_{n=1}^{N} w_n \min(f_{i,n}, f_{j,n}), \tag{1}$$

where $f_{i,n}$ indicates the n-th element of histogram $\boldsymbol{f}_i$, w_n is the weight for measuring the contribution from the similarity on the n-th element, and N ($N = 6256$) is the dimension of the histogram. Since not all the elements in the histogram are equally important for comparing images, we introduce the weight to differentiate the contributions from different features. We first average features across all the images and obtain an average histogram $\overline{\boldsymbol{f}} = \{\overline{f}_n\}_{n=1}^{N}$, and then define the weight w_n as $w_n = 1/\overline{f}_n$. The more frequent the element across all images, the less important it is. This weighting function is able to not only mine informative elements in the histogram, but also balance different types of descriptors.

3.3.3 *Clustering-based exemplary image generation*

We employ a clustering-based approach to generating the exemplary images. We first collect the candidate images by selecting images from the ImageNet and the top 1,000 images from a commercial image search engine according to the entity keywords, respectively. Then, we compute a similarity matrix by comparing all pairs of images based on the visual descriptors and similarity metric described in the previous sections. Last, we adopt the affinity propagation (AP) algorithm to find the visual instances [10]. AP is a widely used unsupervised clustering method which can group features into a number of classes. We sort the clusters according to the number of images in descending order. The centers of the top clusters (we used the top 10) are selected as exemplary images for this entity. To avoid background clutter, a salient region detection process is conducted before feature extraction [19]. Only the visual descriptors within the salient regions are considered. Moreover, a Gaussian window is used to weight the descriptors to make the descriptors close to the centers more important.

3.4 Context-aware Exemplar-based Image Search

Given the composite visual query including recognized entities, selected exemplary images, and their intended positions, the task is to search target images which are contextually relevant to the query. By relevance, we mean that the target images are expected to contain both the entity keywords and visually similar objects in the desired positions. The relevance between visual query and target image consists of the visual similarity and the intent consistency based on user indicated position.

Therefore, we design the searching process in Figure 4, which consists of the following steps: 1) generating textual queries by combining the entity keywords $\{T^{(k)}\}$; 2) searching related images according to the textual queries from an image database; 3) computing the visual similarity $\{e_J^{(k)}(i,j)\}$ between each exemplary image and the corresponding region in the target image J, 4) generating user intent map $\{d^{(k)}(i,j)\}$ according to the positions of the exemplary images indicated by users; 5) computing the "positive relevance" $r_J^{(k)+}(i,j)$ and "negative relevance" $r_J^{(k)-}(i,j)$

for each component by considering both visual similarity and the user intent map; 6) computing the combined relevance $r_J^{(k)}$ for each component by considering the surrounding grids in image J (using a sliding window); and 7) computing the overall relevance r_J between the composite query $\mathcal{C}$ and the target image J. In this way, we can rank the related images returned in step 2 according to the overall relevance scores.

3.4.1 Region-based visual similarity $e_J^{(k)}(i,j)$

To compute the region-based visual similarity between an exemplary image in the composite query and the corresponding region in the target image J, we need a visual representation of the region in J. As it is not practical to compute the visual representation of a specific region in J in real-time (as users may frequently change the position and size of this component), we adopt an efficient grid-based search scheme and partition the target image J into small grids $\{(i,j)\}_{i,j=1}^{M}$. Suppose we select the k-th exemplary image (corresponding to the region $R^{(k)}$ in the composite query), and its center position corresponds to the grid (i,j) in J, then the corresponding region $R_J^{(k)}$ in J is given by the union of all the associated grids, i.e., $R_J^{(k)} = \bigcup_{(i,j)\in R^{(k)}}(i,j)$. In each grid, the feature histogram is obtained using the approach described in Section 3.3.1 and saved in advance. Now, the target image J can be represented as $\{\boldsymbol{h}_J(i,j)\}_{i,j=1}^{M}$, where $\boldsymbol{h}_J(i,j)$ is the visual descriptor for the grid (i,j). We choose $M = 9$ in our implementation. Then, the visual representation of $R_J^{(k)}$ can be obtained using the linear fusion of histograms from the related grids:

$$\boldsymbol{f}_J^{(k)}(i,j) = \sum_{(i,j)\in R_J^{(k)}} w_J(i,j)\boldsymbol{h}_J(i,j), \tag{2}$$

where $w_J(i,j)$ is a 2D Gaussian distributed weight centered at the given region, which assigns more importance on the grid close to the center.

Then, the region-based visual similarity between the k-th exemplary image and the region $R_J^{(k)}$ can be given by

$$e_J^{(k)}(i,j) = \mathrm{sim}\big(\boldsymbol{f}^{(k)}, \boldsymbol{f}_J^{(k)}(i,j)\big), \tag{3}$$

where $\boldsymbol{f}^{(k)}$ is the visual descriptor of the k-th exemplary image, while $\mathrm{sim}(\cdot)$ is given in equation (1). Note that in the above equation, we use both the index of (i,j) and k. This is because we will use a sliding window to compute the region-based similarity later to deal with the tolerance of position. Therefore, $e_J^{(k)}(i,j)$ indicates the visual similarity between the k-th exemplary image and the corresponding region centered at (i,j) in the target image J.

3.4.2 Region-based intent relevance $r_J^{(k)}$

The computation of the region-based relevance between the exemplary image $I^{(k)}$ and the corresponding region $R_J^{(k)}$ should take the user intent into account. Intuitively, user intent close to the center of each $R^{(k)}$ is stronger than that which is far away from the center. Moreover, user intent within the exemplary image $R^{(k)}$ is stronger than intent outside of if. We first define the user intent map which is a soft measurement of user intent in the composite query.

Let $(x^{(k)}, y^{(k)})$ denote the center of the k-th exemplary image in the composite visual query. To tolerate the uncertainty of this position specified by the user, we compute the following soft map to represent the user intent:

$$\begin{aligned} d(x,y) &= 2g(x,y) - 1, \\ g(x,y) &= \exp\left\{-\Big(\frac{x - x^{(k)}}{\theta\cdot w^{(k)}}\Big)^2 - \Big(\frac{y - y^{(k)}}{\theta\cdot h^{(k)}}\Big)^2\right\}, \end{aligned} \tag{4}$$

where $w^{(k)}$ and $h^{(k)}$ are the width and height of $R^{(k)}$, respectively, and θ is set to a constant $(8\ln 2)^{-1/2}$ to make g degrade to 0.5 at the border of $R^{(k)}$. Then, the intent consistency in terms of k-th component at grid (i,j) is given by

$$r_J^{(k)+}(i,j) = \min\big(e_J^{(k)}(i,j), d^{(k)}(i,j)\big). \tag{5}$$

This is called "positive relevance" as it mainly focuses on the grids within $R_J^{(k)}$. We also present a scheme to penalize the case that an entity exists in an undesired position (i.e., out of the user indicated region). By checking the relevance of each grid outside the region, the penalty score can be obtained by

$$r_J^{(k)-}(i,j) = \min\big(e_J^{(k)}(i,j), -d^{(k)}(i,j)\big) \tag{6}$$

This is called "negative relevance" as it penalties the grids outside $R_J^{(k)}$.

It is not easy for users to indicate their intent very precisely on the composite query canvas (e.g., the exemplary image may be positioned in an approximate position rather than an exact position, and not well resized). Therefore, we need to maintain tolerance to the position and the size of each exemplary image. To deal with the tolerance issue, we use a sliding window for which the size is the same as $R^{(k)}$ and place this window centered at all the grids (i,j) in $R_J^{(k)}$. In other words, the original exemplary image is re-positioned on these sliding windows to introduce some position tolerance. We are always searching for the best match among these sliding windows, as shown in Figure 4. As a result, the "positive relevance" and "negative relevance" between the k-th exemplary image and $R_J^{(k)}$ is computed by

$$\begin{aligned} r_J^{(k)+} &= \max_{(i,j)\in R_J^{(k)}} \{r_J^{(k)+}(i,j)\}, \\ r_J^{(k)-} &= \max_{(i,j)\notin R_J^{(k)}} \{r_J^{(k)-}(i,j)\}. \end{aligned} \tag{7}$$

The combined relevance between the k-th exemplary image and $R_J^{(k)}$ is

$$r_J^{(k)} = r_J^{(k)+} - r_J^{(k)-}. \tag{8}$$

3.4.3 Overall relevance r_J

After we obtained all the region-based relevance $r_J^{(k)}$, the overall relevance between $\mathcal{C}$ and J is computed by combining across all the components. To consider the variance of different components, the overall relevance is given by a fusion function sensitive to both the mean and the variance of each region-based relevance:

$$r_J = E(r_J^{(k)}) - \frac{\gamma}{K}\sum_{k=1}^{K}\big|r_J^{(k)} - E(r_J^{(k)})\big| \tag{9}$$

where $E(r_J^{(k)}) = \frac{1}{K}\sum_k r_J^{(k)}$ is the average relevance, and γ is a positive parameter controlling the penalization degree, which is empirically set to 0.8 in our implementation.

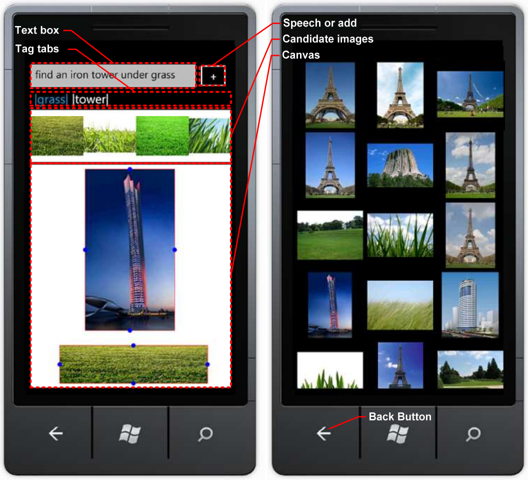


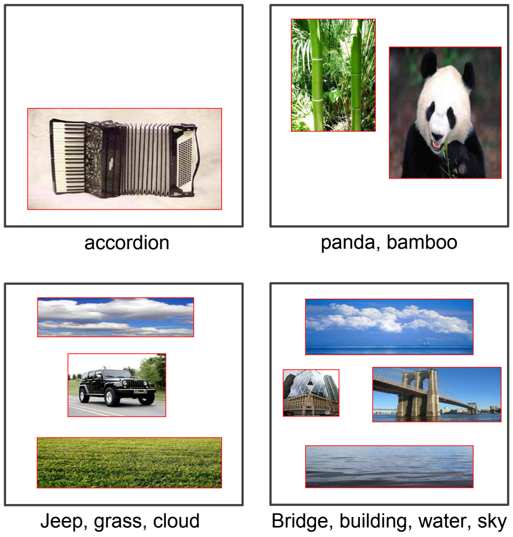
Figure 5: The user interface of JIGSAW on a Windows Phone 7 device. The left is the operation interface, while the right is the search result page.

4. IMPLEMENTATION

We deployed JIGSAW as an application on Windows Phone 7 devices. Figure 5 shows the user interface. On the top of the screen is the text box to accept a textual query. It can also display speech query after tapping the button on the right and record a piece of speech. By tapping the right button, the textual query in the text box will be parsed and the entity keywords will appear in the tag list. Below the tag list, there is an exemplary image list which shows the exemplary images searched according to the selected keyword. The user can drag any exemplary image onto the composite query canvas below, so that the exemplar will be displayed on the canvas. Both lists can slide horizontally by touching and sliding. Double tapping results in the deletion of an item in the list. On the canvas, the user can both reposition the exemplar by dragging or resizing it by two-point touch stretching. Finally, if the user is satisfied with the composite query, s/he can double-tap the canvas to trigger the search. The search results will be displayed in a new page.

5. EVALUATIONS

The dataset includes the images collected by more than 1,000 entities in a commercial image search engine. These entities are collected from some famous datasets, including Caltech-101 [9], Caltech-256 [13], and the 600 most popular entities from ImageNet [7]. To evaluate the usability compared to the capture-to-search mode, we also use a small number of nouns based on their popularity in Flickr (i.e., frequency of tags), including landmark, product, celebrities, and event. For each entity, we searched the top 500 images from the search engine. The combinations of each pair of entities were also issued as the queries to collect more comprehensive data. In order to cover more possible entities, we also included half a million Flickr images into our database. In total, there are about 1.5 million images in the database. It is challenging to index a large number of images by considering the grids in the targeted images. In JIGSAW, we first use the entities extracted from the speech to filter the images and keep about 1,000 images as candidates for the subsequent steps.


accordion

panda, bamboo

Jeep, grass, cloud

Bridge, building, water, sky

Figure 6: Examples of four composite visual queries.

Table 3: Distribution of the number of components.

# of components (entities)	1	2	3	4
# of test queries	40	30	20	10

5.1 Evaluation of Visual Search

To test the performance of image search in JIGSAW, 100 queries were collected as the query set. These queries included 42 concepts used in [31]. Furthermore, we collected some queries from search history in JIGSAW. For queries with more than one component, since not each combination has possible search results, we only selected 58 queries. Figure 6 shows some examples of these queries. As short queries are dominant in the search query logs, we leveraged more short queries in the experiment. Table 3 lists the number of queries with different number of terms.

We compared search performance among text-based image searches, the algorithm in the Concept Map [31], and the proposed JIGSAW. Six people with technical backgrounds were invited to evaluate the system. They tried the system and got familiar with it at first. Then they were asked to label the search results. Images were organized into three levels: 0—not relevant, 1—relevant, 2—very relevant. Each image was labeled by at least three different subjects. The median score was used as the ground truth. The Normalized Discounted Cumulative Gain (NDCG) [17] was used to evaluate the search performance. Limited by the screen size and bandwidth, mobile phones can display much fewer images than a PC, making the top ranked images crucial. NDCG scores are calculated based on the top 20 images.

Figure 7 shows the NDCG for different search methods under different numbers of keywords. Text based image search had the worst performance under all settings because the text-based search was not able to understand the image content only based on text queries, and user intent cannot be well expressed through a piece of text. We also included more common combinations of three keywords, so that the performance of text-based results were a little higher than that of two keywords. JIGSAW is also better than Concept Map because it can consider the spatial representation of the region. In the case of a single keyword, JIGSAW is better

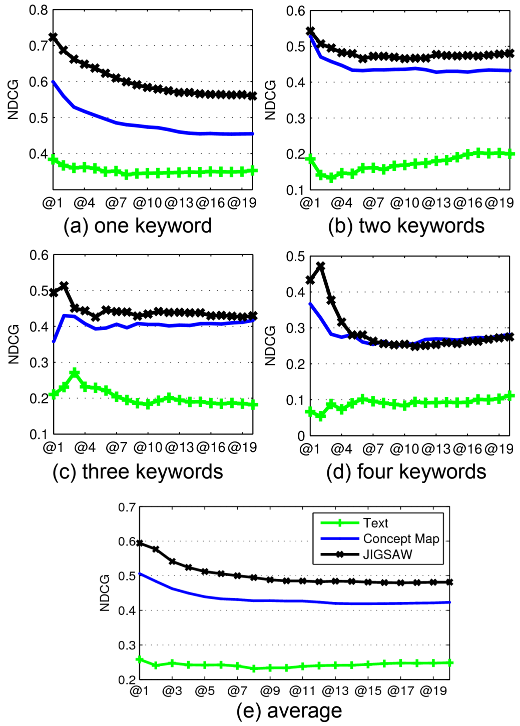


Figure 7: Comparison of the image search performance among three different approaches.

than Concept Map because JIGSAW uses combined grids instead of separated grids. The performance of JIGSAW and Concept Map get closer as the number of keywords increases because the intended regions become similar.

5.2 Subjective Evaluation of Usability

We invited 12 subjects to use our system on mobile phones. These subjects consist of two female and ten male college students, with the ages ranging from 22 to 27. Half of them already had some experience with multi-touch devices, while three of them had never used smart phones before. Two subjects had experience with visual search on mobile phones and one had experience with speech search on mobile phones. However, after three minutes of orientation and demonstration of JIGSAW, all of them became sufficiently familiar with how to use the system. Three of them felt the interface was very cool when they first saw it. Some of them were even surprised when they found that position and size affected the search results. Most of them were able to get familiar with our system within three to five minutes by completing one or two search trials. It is worth noting that among all subjects, one had difficulty typing on the touch screen, so he felt that speech input greatly helped him. Through the orientation, we found that the JIGSAW is acceptable for most of the subjects.

After learning the system well, they were asked to accomplish some tasks. In each task, one subject randomly choose a keyword from the 1,000 candidates. They first worked on a computer typing the keyword in the text search engine and picked one image within the top 500 results which they found interesting. They were then required to use JIGSAW to find the same image or similar images that satisfied them. We asked them to use a voice query first along with text if necessary, and choose visually similar exemplary image(s) by putting it on the canvas at their intended position. One, two, and three entities were used separately to search images, resulting in 36 tasks in total. In the experiment, only two tasks failed to find the desired images (i.e., the target image did not appear in top 40 results). In 20 tasks, the desired images were in the top 10 results. The time distribution required to successfully complete each task are shown in Figure 8. A questionnaire was also filled out by each user, asking for participants' input on usability, user friendliness, natural experience, and so on. Through these tasks, all of the subjects found JIGSAW successfully facilitates their search intent.

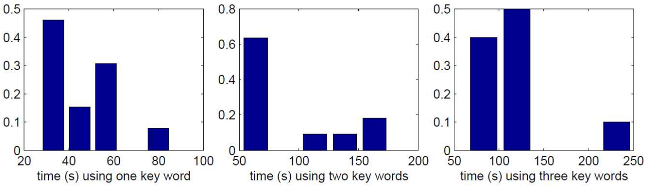


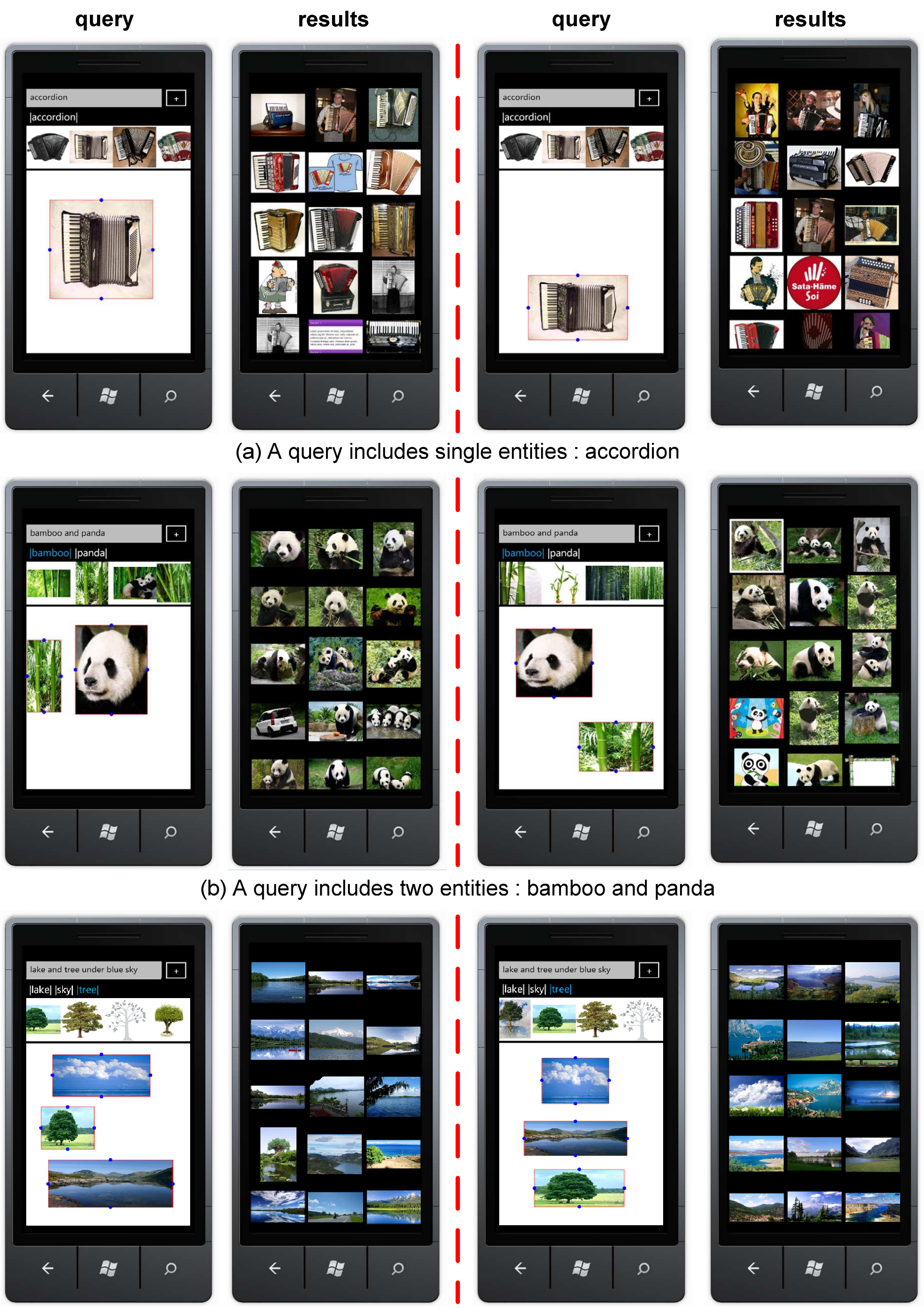
Figure 8: The time distribution for different users to complete a task.

Table 4: A summary of user study by comparing three mobile apps: (a) Google Image, (b) Google's Goggles, and (c) JIGSAW. Each metric is rated from 1 to 5 indicating the worst to the best level.

#Q	Question	(a)	(b)	(c)
1	attractiveness (overall)	2.5	3.5	4.7
2	naturalness (overall)	3.5	3.9	4.1
3	enjoyability (overall)	2.6	3.7	4.5
4	efficiency (overall)	3.0	3.7	3.9
5	input method (overall)	2.5	3.8	4.2
	interaction	2.3	3.7	4.6
6	+ novelty	1.8	3.5	4.6
7	+ naturalness	2.8	3.5	3.8
8	+ user friendly	3.1	3.5	3.8
9	+ efficiency	3.1	3.4	3.9
10	+ clarity of intent	2.8	3.6	4.4
11	effectiveness of search results	2.8	3.5	3.9
12	preference	3.2	4.0	4.1
13	ease to use	2.0	3.3	3.9
14	operation	2.8	3.3	4.2

According to our observations, to successfully complete a search task with one keyword usually takes about 30 sec. Each failed trial or extra effort will increase the time by about 20 sec. While according to [14], a typical image search on PC by text query requires about 140 sec. The average response time and number of operations are 1.9 sec. and 9.8 for one entity, 3.6 sec. and 16.6 for two entities, and 7.1 sec. and 26.9 for three, respectively. These show that JIGSAW can play a positive role in accelerating the image search process on mobile phones.

A quantitative evaluation comparing three applications (i.e., Google image search, Goggles, and JIGSAW) are listed in Table 4. This shows the advantage of JIGSAW over traditional text-based image search and the complementary power of capture-to-search applications. All of the subjects thought that JIGSAW is attractive and an effective complement to text-based search modes. 83% thought the composite visual query helped them to find images, and the interface is user friendly, and the process is enjoyable. 75% thought this system is natural and better to use than a text-based



(a) A query includes single entities : accordion

(b) A query includes two entities : bamboo and panda

(c) A query includes three entities: lake, tree, and sky

Figure 9: Some visual examples of JIGSAW results.

search engine. 67% think the voice input is easy to use. Moreover, they are looking forward to new features from JIGSAW, such as color selection or photo capturing. When asked whether they would install this application on their own smartphones, they gave a positive response with 3.9/5.0 on average, indicating that most of them would like to install and use it. Some also hope they would use it on a larger touchscreen. In addition, after an introduction of other visual search systems (e.g., Goggles), all subjects claimed that they would like to use both JIGSAW and other visual search systems to conduct searches for different scenarios.

5.3 Complexity Analysis

We tested the response time with the 100 queries. Figure 10 shows the time consumption with different numbers of keywords. The image search component does not spend much memory. For example, a single keyword with 1,000 candidate images (histograms) needs to be loaded onto the memory. The average number of sift points is 300 per im-

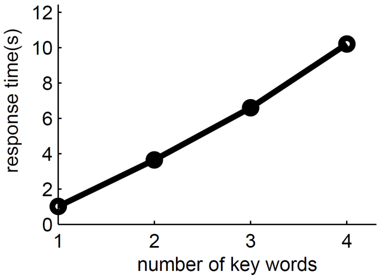


Figure 10: The system response time with different numbers of keywords.

age. Although with the color and edge descriptors, it takes 3.3Kb for each image and 3 Mb for all the 1,000 candidate images in total. This number will increase as the number of keywords increase because of the candidate images. When the number of keywords increases, more components need to be checked, which also multiples the search time. As shown in Figure 10, the time consumption is roughly linear to the number of keywords. Further optimization can be done to reduce the time consumption.

5.4 Visual Examples

In Figure 9, we show some results of JIGSAW, including the user intent which cannot be clearly expressed in the traditional text-based search engine. Such kinds of tasks cannot be accomplished in the capture-to-search mode.

6. CONCLUSIONS

We have proposed an interactive mobile visual search system which enables users to formulate their search intent through natural interaction with mobile devices. The proposed system, called JIGSAW, represents the first study on mobile visual search by taking the advantages of multimodal and multi-touch functionalities on the phone. Our experiments show that JIGSAW is an effective complement to existing mobile search applications, especially in cases where users only have partial visual features in mind for what they want to find. User's search experience on the phone is significantly improved by JIGSAW as it provides a game-like user interface for query formulation. Our future work will include: 1) deploying JIGSAW to tablet systems to create a better user experience, 2) leveraging user behaviors during interaction for better search performance, and 3) expanding entity lexicon to handle more queries.

7. REFERENCES

[1] http://www.pwc.com/gx/en/communications /review/features/mobile-data.jhtml.

[2] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: speeded-up robust features. In *Proc. of ECCV*, pages 346–359, 2008.

[3] Y. Cao, H. Wang, C. Wang, Z. Li, L. Zhang, and L. Zhang. MindFinder: interactive sketch-based image search on millions of images. In *Proc. of ACM International Conference on Multimedia*, pages 1605–1608, 2010.

[4] V. Chandrasekhar, G. Takacs, D. Chen, S. Tsai, R. Grzeszczuk, and B. Girod. CHoG: Compressed histogram of gradients—a low bit-rate feature descriptor. *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, pages 2504–2511, 2009.

[5] V. R. Chandrasekhar, S. S. Tsai, G. Takacs, D. M. Chen, N. M. Cheung, Y. Reznik, R. Vedantham, R. Grzeszczuk, and B. Girod. Low latency image retrieval with progressive transmission of CHoG descriptors. In *Proc. of ACM Multimedia Workshop on Mobile Cloud Media Computing*, pages 41–46, 2010.

[6] K. Church, B. Smyth, P. Cotter, and K. Bradley. Mobile information access: A study of emerging search behavior on the mobile internet. *ACM Transactions on the Web*, 1(1), May 2007.

[7] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei. Imagenet: a large-scale hierarchical image database. 2009.

[8] Digimarc Discover. "https://www.digimarc.com/discover/".

[9] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, 2007.

[10] B. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315(5814):972–976, 2007.

[11] GazoPa. http://www.gazopa.com/.

[12] Google Goggles. http://www.google.com/mobile/goggles/.

[13] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. 2007.

[14] M. Jia, X. Fan, X. Xie, M. Li, and W. Ma. Photo-to-Search: Using camera phones to inquire of the surrounding world. In *Proc. of Mobile Data Management*, 2006.

[15] X. Li. Understanding the semantic structure of noun phrase queries. In *Proc. of Annual Meeting of the Association for Computational Linguistics*, pages 1337–1345, 2010.

[16] LinkMe Mobile. http://www.snapnow.co.uk/.

[17] Y. Liu, T. Mei, and X.-S. Hua. CrowdReranking: exploring multiple search engines for visual search reranking. In *Proc. of ACM SIGIR conference on Research and Development in Information Retrieval*, pages 500–507, 2009.

[18] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[19] Y.-F. Ma and H.-J. Zhang. Contrast-based image attention analysis by using fuzzy growing. In *Proc. of ACM Multimedia*, pages 374–381, Nov 2003.

[20] G. Miller. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.

[21] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, pages 2161–2168, 2006.

[22] NOKIA Point and Find. http://pointandfind.nokia.com/.

[23] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):257–286, 1989.

[24] S. Sekine, K. Sudo, and C. Nobata. Extended named entity hierarchy. In *Proc. of LREC-2002*, 2002.

[25] Smart VisualTM Kooaba. http://www.kooaba.com/.

[26] SnapTell. http://www.snaptell.com/.

[27] G. Takacs, Y. Xiong, R. Grzeszczuk, and et al. Outdoors augmented reality on mobile phone using loxel-based visual feature organization. In *Proc. of ACM International Conference on Multimedia Information Retrieval*, pages 427–434, 2008.

[28] TinEye. http://www.tineye.com/.

[29] J. Wang and X.-S. Hua. Interactive image search by color map. *ACM Trans. on Intelligent Systems and Technology*, 3(1), 2012.

[30] Xcavator. http://www.xcavator.net/.

[31] H. Xu, J. Wang, X. Hua, and S. Li. Image search by concept map. In *Proc. of ACM SIGIR conference on Research and development in information retrieval*, pages 275–282, 2010.

[32] W.-K. Yang, A. Cho, D.-S. Jeong, and W.-G. Oh. Image description and matching scheme for identical image searching. *Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, Computation World*, pages 669–674, 2009.