Managing and Searching Personal Photo Collections

Ullas Gargi, Yining Deng, Daniel R. Tretter
Imaging Systems Laboratory
HP Laboratories Palo Alto
HPL-2002-67
March 18th , 2002*

E-mail: {gargiu, deng, tretter} @hpl.hp.com

photo collections, image similarity, face detection, camera metadata, EXIF

We present a design of a prototype system for managing and searching collections of personal digital images. The system allows the collection to be stored across a mixture of local and remote computers and managed seamlessly. It provides multiple ways of organizing and viewing the same collection. It also provides a search function that uses features based on face detection and low-level color, texture and edge features combined with digital camera capture settings to provide high-quality search that is computed at the server but available from all other networked devices accessing the photo collection. Evaluations of the search facility using human relevancy experiments are provided.

* Internal Accession Date Only
Approved for External Publication

Managing and Searching Personal Photo Collections

Ullas Gargi, Yining Deng and Daniel R. Tretter
Hewlett-Packard Laboratories, Palo Alto, California
{gargiu, deng, tretter}@hpl.hp.com

Abstract

We present a design of a prototype system for managing and searching collections of personal digital images. The system allows the collection to be stored across a mixture of local and remote computers and managed seamlessly. It provides multiple ways of organizing and viewing the same collection. It also provides a search function that uses features based on face detection and low-level color, texture and edge features combined with digital camera capture settings to provide high-quality search that is computed at the server but available from all other networked devices accessing the photo collection. Evaluations of the search facility using human relevancy experiments are provided.

1. Introduction

With the rapid acceptance of digital cameras as a consumer image capture tool, personal digital photo albums are becoming a common and highly valued form of personal information. Current photo applications or services replace film paper with electronic files and album books with electronic file folders. However, it is still a tedious task for typical users to manage personal digital photo collections, much like the old "photo shoeboxes."

One new problem is that files often need to be organized across different storage and viewing devices. Putting photos on a web server enables easy sharing among friends and families. But issues like slow Internet connections, storage cost, and privacy restrain people from uploading all their photos to the web. Local PC disks will still be a primary device for storing personal photos. In addition, many users would like to archive their photos on removable media such as CD. The problem is that applications on these different devices are not directly linked to each other. Photos, once uploaded, are no longer associated with their corresponding local copies. Instead, multiple copies of the same photo are stored in different places, and each can only be accessed using a separate application. When a collection grows to thousands of photos, it becomes very easy for a user to lose track of their files.

Data synchronization technologies are common for mobile digital devices, ranging from various proprietary products to open standards such as SyncML [4]. Such synchronization features are not seen in consumer photo management applications. Furthermore, application-specific tasks need to be performed above the data synchronization layer to ensure optimal performance. Many of these operations involve image transcoding due to limited bandwidth, storage, or display size. Instead of synchronizing the exact same file, a scaled-down version could be transferred and stored for a particular device. For example, a PDA does not have a high-resolution display, and it is not necessary to send a full resolution image to that device. The bandwidth and storage cost also make it more efficient to use thumbnail files instead.

Another problem that remains to be solved is photo searching and retrieval. Despite progress on content-based retrieval in recent years, current techniques are unable to identify semantic content and instead can only offer visual similarity based search. While keyword based search is available in some photo applications, most users are unlikely to type in labels for hundreds or thousands of photos.

This paper presents a prototype system for managing and searching personal image collections on multiple devices. Personal collections differ from stock photo collections used in traditional image retrieval systems in important ways:

1. Although increasingly larger due to the profligate capture that digital cameras encourage, the number of images in such a collection is usually limited compared to professional or business image databases. Therefore many of the traditional database performance issues do not arise.

2. The majority of images in such collections are acquired from digital cameras which insert metadata at the time of capture into the image headers. This is especially true of new images added to the collection. This is not true of legacy stock photo collections which are often scanned images from analog capture. This capture metadata can be used to improve search, as we demonstrate in this paper.

3. A limited number of digital cameras (usually one) is used; therefore the metadata fields present in all images are the same or form a consistent set.

4. The semantic search problem in this space is easier to objectively evaluate. The images in a collection have personal meaning to the owner and the relevancy score of a search result can be assigned

immediately by the user. This is the experimental approach we take to evaluate our system.

2. Previous Work

The PhotoFinder project uses novel visualization paradigms, dynamic queries and query previews to improve searching of personal photos[7]. Their work concentrates on annotated collections and uses Boolean and explicit queries rather than the simpler query-by-example point-and-click paradigm we use. In our experience fully manually annotated collections are rare.

There has been a lot of research work on visual similarity search, or content-based retrieval, in the past years. Because of the limitations of visual similarity search, researchers have tried to combine it with other image analysis technologies such as face detection and recognition [1] to improve retrieval results. Most previous work on using human faces in image retrieval has concentrated on face recognition [3][10]. The Fotofile system [6] was one of the first systems to do automatic face detection, recognition and continuous online learning. Work requiring no manual annotation and using just the results of face detection alone and the size and distribution of detected faces is rare. Srihari et al [2] segment out faces and corresponding bodies and compute low-level features from the resulting background image. Their work is oriented toward using associated text to index images. They do not appear to use statistics on the detected faces to match images. The ImageScape system [8] lets users drag icons depicting a face onto a query template image.

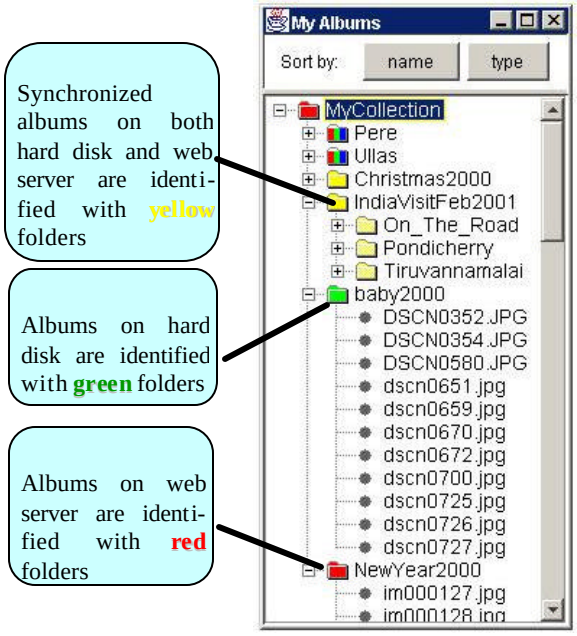

Fig. 0. A snapshot of a unified interface for accessing photos on multiple devices.

There is little work on using digital camera metadata as an additional resource to solve the similarity search problem. As a simple example, using the focal distance, we can differentiate blue sky from a blue shirt, which is difficult in traditional content-based retrieval schemes. Even the trivial yet enormously important image capture time information has not been used effectively.

3. Unified Collection Management

3.1. Unified Hierarchical Organization

Our system design allows for a user to easily manage a personal collection of images that is distributed between local computer storage, such as a PC hard disk, remote server storage, such as a photo sharing website, and archived subsets, such as on CD-R media.

A collection is organized into a number of top-level albums. Albums can have child albums; thus a hierarchical organization is possible. Each of these albums can be stored on the server, on local disk, or on an archive, or a combination of all three. The location of the album is indicated by color coding. Fig. 0 shows a snapshot of the user view of a collection. Transferring files between a PC and a server or between peer networked clients can be done by a simple drag-and-drop or cut-and-paste. Note that some upload clients for photo sharing web sites also allow drag-and-drop transferring of photos. However, those clients are only used for transferring photos, while we provide a system that let users manage their photo collections on different locations through a single client.

3.2. Separation of data and metadata

The organization information is stored in an XML format file that contains the hierarchical organization information as well as annotations supplied by the user. One design goal is to allow this metadata and organizational information to be combined with thumbnails of the image to create a portable compact version of the entire image collection. This organization index is small enough to be carried on mobile devices and saved on all locations. For example, every CD-ROM archive or every digital camera memory card can carry this index in addition to the actual images it contains. This allows browsing access to the entire photo collection even when certain storage devices are not available.

3.3. Virtual Albums

Our system allows multiple ways of viewing the same photo collection. With film prints and albums, there is only one way of viewing the photos. Typically photos are organized based on events such as birthdays, holidays, and vacation trips. But what about presenting photos with a different storyline: photos of me in college, our family's favorite photo moments, or all my trips to Africa? With digital photos, one could make duplicate copies and create new albums, which would increase storage and management problems. A new

feature, the virtual album, is introduced in the system to let users easily create customized views of their photo collections for different purposes. Again, as the collection grows to thousands of photos, selecting the right photos could be tedious and time-consuming. The search technologies described in this work can be used to generate new virtual albums instantly on demand.

Virtual albums are shown in special RGB folders. Photos in virtual albums contain links to images in other albums. The link does not have to be a static file link. It could be a URL and even point to a dynamically generated image, such as "photo of the day." The link location information can be transparent so that after an initial setup, as far the users are concerned, they are just accessing photos and do not have to worry about where they are located. A virtual album can also recursively contain virtual subfolders of file links. The file structure of a virtual album can be stored in a database or in separate file(s). Links are updated automatically if changes to the original files are made, for example when original photos are renamed or moved.

Virtual albums allow multiple views of the same photo collection without additional storage cost. Because links are used, if the original image is edited the changes are reflected in the virtual album as well. This has an advantage over duplicating files. Virtual albums can be converted to physical albums when needed. By using virtual albums, one can have an album titled "Greatest Hits" that contains images that lie in different places but are all brought together in one view.

3.4. Synchronized Local/Remote Storage

For identical albums that are located at different places, changes to one location copy that the user makes to an image or album are automatically propagated to all other locations. This includes changes to virtual images that point to real ones, as well as changes to real images that are pointed by virtual images. Changes made while the client and server are disconnected are remembered and executed the next time a connection is made.

4. Server-side Search for Distributed Collection

Although the collection is distributed, the feature index is stored on the server and updated whenever the collections are synchronized. Therefore advanced searching based on computationally intensive algorithms can be provided to the client end by using services over HTTP. Search results presented are obtained on the server but matching images can be displayed from the local store. This allows maximum flexibility in adding new features to the search and management facility without requiring frequent software upgrades on the part of the user. Keyword searching as well as searching based on face detection with queries such as "crowd" are supported. The query-by-example similarity search feature is designed to fulfill a user need when browsing, text-based, or other searches are insufficient.

4.1. Low-level Image Feature Representation

The visual similarity search function is based on a low-level image feature representation. An image feature is a high-dimensional vector with components comprising a smoothed color histogram, a color coherence vector, edge direction histogram and texture features.

To compute an image-pair distance, L_2 distances are computed for each component and then combined using static weights, fixed empirically. No index is used to speed up search simply because the size of any single collection is small enough to allow exhaustive search in interactive time.

4.2. Camera Metadata

Most digital cameras follow the PIMA/DIG/IIIA EXIF standard[11]. This standard defines the format of header fields in JPEG images saved on to storage cards in digital cameras. It allows for the insertion of various types of metadata by the camera processor, such as basic image parameters (height, width, compression), location/time of capture (location is available in a very few cameras), and capture settings (flash usage, focus distance, exposure time).

4.3. Use of Camera Metadata in Search

One of the problems with using camera metadata to refine a set of visually similar images is that the subsets of EXIF headers vary from camera to camera. We choose to use camera settings to improve the visual similarity search when available. Our method uses 3 different camera metadata fields. The time/date stamp is used to compute a value called CoarseTime, which consists of the number of days between Jan 1, 2000 and the date the photo was taken. We also use the Aperture FNumber and the SubjectDistance fields. When comparing two photos, we do the following:

1. Saturate abs(CoarseTime1 - CoarseTime2) at 30 days.

2. Set both F numbers to be between 2.4 and 4.8 (if < 2.4, set to 2.4; if > 4.8, set to 4.8).

3. Saturate both SubjectDistances at 20 meters.

4. Compute the camera-metadata similarity measure between two photos using the equation below (the parameter α is set to 30)

$$a * \left[\left(\frac{(CoarseTime1 - CoarseTime2)^2}{900} - 1 \right) + \left(\frac{(Fnum1 - Fnum2)^2}{5.76} - \frac{1}{2} \right) + \left(\frac{(Dist1 - Dist2)^2}{400} - \frac{1}{2} \right) \right]$$

During search, this similarity value is then added to the visual feature distance between the two photos to augment the results of visual similarity search. The result is that photos captured less than 30 days apart, or with

similar subject distance or F number, have a decreased feature distance, classifying them as more similar.

4.4. Use of Face Detection in Search

A neural-network based face detection algorithm [10][6] is used to detect faces within the entire database. This algorithm detects front facing upright faces with a small tolerance for off-vertical rotation. The bounding boxes and eye locations are computed offline and stored. When a similarity search using face detection is requested, an image-pair similarity is computed based on the number, size and location of the normalized bounding boxes of detected faces in the two images. A greedy algorithm matches face pairs in largest-size-first order, deleting faces that have been matched. The final similarity value between two face lists adds the face-pair similarities and subtracts the cost of left-over faces in either image.

$$Sim_{face}(I_1, I_2) = MatchedFacesValue - ResidualFaceCost$$

where without loss of generality image I_1 has fewer or equal number of faces as I_2.

$$MatchedFacesValue = \sum_{face\ i \in I_1} BestMatch(i, I_2)$$

The *BestMatch()* function finds the best remaining match in image I_2 for face i in image I_1.

$$BestMatch(i, I_2) = \max Match(i, j),\ j \in I_2\ (unmatched\ faces)$$

Match values are computed as a weighted sum of FaceExistenceMatch, FaceSizeMatch, and FaceLocationMatch. FaceExistenceMatch is always 1.0. This is to guarantee that an image with a face is always more interesting that one without. FaceSizeMatch is equal to the normalized intersection area of the face. Therefore large faces contribute more than smaller ones. FaceLocationMatch is the complement of the normalized distance between centroids.

ResidualFaceCost subtracts, for every unmatched face, the ratio of the size of the unmatched face to that of the smallest matched face (from either image), multiplied by a penalty factor. For a query with faces, images without faces will be penalized by a fixed amount and for a query without faces, images with faces will be penalized a variable amount, depending on the size and number of faces they have.

This method weights larger faces more than smaller ones and attempts to match an image to other images containing similar numbers, sizes and locations of faces. As in the visual similarity matching, speed optimizations are possible but not essential for our application. The face similarity value is then used to modify the visual similarity.

Apart from image similarity matching, face detection also allows higher-level queries to be executed. For example, our system allows searching for images based on the presence of crowds and portrait shots.

5. Experimental Evaluation

Four methods of similarity search were tested based on their ability to find images *relevant* to a given image. The four methods were

1. EXIF capture-time similarity: a simple absolute difference in capture times. Note that more sophisticated methods taking weekday/weekend, holidays, etc. into account are possible.

2. Visual similarity alone

3. Visual similarity augmented by camera metadata

4. Visual similarity augmented by camera metadata and face detection similarity

Figure 1. Photo Collection Search Interface

Four users AP, AS, OT and PO were asked to choose a set of query images from their personal collections of sizes 694, 535, 471, and 3140 respectively (see Table 1. Characteristics of Photo Collections). They picked 5 favorite or memorable images by browsing the hierarchy or by iterative random search and 5 more images from a random set of 20. In a real-use scenario these images would be obtained by browsing or text searching. For each image, the 4 different similarity functions were used to present matching images. The test was blind -- the users did not know which similarity function was being used. For each search performed, the users were shown only the top 20 matches. The subject then counted how many of the 19 non-original matches were relevant. The definition of relevancy was left to the user, but three scenarios were suggested to convey the idea:

1. whether the found images would be reasonable replacements for the original if it were lost or deleted,

2. whether they would put the found images together with the original in a themed slideshow

3. whether they would group the found photos together with the original in a printed photobook.

Table 1. Characteristics of Photo Collections

Subject	Camera(s) Used	No. of photos
AP	Nikon Coolpix 950, HP PhotoSmart 715	694
AS	Olympus C3040Z	535
OT	Sanyo VCP-SX500	471
PO	HP PhotoSmart 618, HP Jornada Pocket Camera	3140

In our opinion this kind of task-oriented searching is more realistic than arbitrary search. Note that this differs from task-oriented search for a single target image [12]. Subjects were encouraged to enjoy the experience of browsing through their collections, in addition to performing the experiment.

This evaluation method only measures precision not recall. We chose it for two reasons: first, as a practical matter, obtaining recall requires complete ground-truthing and is therefore difficult. Second, we believe that this is the only metric that really matters for a feature in a software application or service, because user satisfaction is mainly governed by the reasonableness of what they see when they press the search button (for non-specific targeted search). Note also that the criteria that subjects used to define relevancy varied from image to image and was often influenced by the first set of results. The alternative of asking the subject to define the relevancy criteria before seeing any results was rejected for the same two reasons as before: the making of an already hard subject task harder, and post-facto determination of user satisfaction being the ultimate metric. We are not so much interested in trying to model human perception as in building a search function that people will find useful. Figure 1. Photo Collection Search Interface shows the interface.

6. Results and Discussion

Tables 3 through 6 shows the reported relevancy scores for each query image for each user. The numbers are the number of images in the top 20 hits judged relevant by the subject (other than the original image itself). The 'Top' column shows the number of times that method was the top or tied-for-top performing one. The 'Total Hits' column counts the total number of relevant images across all queries. The Olympus camera used by subject AS did not record capture time metadata for some reason. Some other photos were modified and saved, losing the EXIF header information. Therefore there are no time-similarity results for those photos. Table 7 shows aggregate scores for each method and improvement by using camera metadata and face detection over visual similarity alone. It is clear that time-based similarity outperforms all other methods when it is available. However, visual similarity has the advantage of always being available, regardless of the image source or condition. Using camera metadata and face detection improves performance over visual similarity alone.

A particular subject would state different factors when asked to describe the criteria used to measure relevancy for different images. "Within the same event" favored time-based similarity, "With this person" was another, with "With this person alone" being a variant. Also, different subjects gave different weightings to the presence of human faces (for example, subject AS gave a higher weighting to face similarity).

It should be noted that the relatively small temporal extent of the collections probably biases in favor of time-based similarity. This is because memory of the period is still fresh so event-based clustering is the natural mental model. However, with larger collections taken over a span of many years, there may be correlations between photos taken a long time apart that time-based similarity will not reveal. The other methods used would allow these correlations to be revealed and the photos grouped and presented in novel ways that are 'surprising' to the user, for example, a screensaver or slideshow using themes based on facial, visual and camera-metadata similarity.

It is interesting to note the occurrence of human faces in photos. The numbers are shown in Table 2. Human Face Occurrences in dataset. Note that the actual numbers are higher given the false negative rate of the face detector used, its inability to detect off-vertical or non-frontal faces, and the fact that many photos were not oriented the right way up. Even then face occurrence rates of as high as 62% are seen (OT), subject to the false positive rate of the detector.

The method of combining different metrics we used could use user input to improve performance. From our observations people use different criteria to judge semantic relevancy but these can be grouped into event, appearance or person-based similarity. Using an initial dialog box before the search would allow the user to indicate which kind of results they would like with appropriate weighting factors then being applied. Face

and person recognition would also be obvious refinements in this scheme.

Table 2. Human Face Occurrences in dataset

Subject \ NumFaces	>= 1	>= 2	>=3	> 3
AP	66	20	5	3
AS	280	88	37	19
OT	310	184	110	66
PO	731	266	95	36

7. Conclusions and Summary

We have presented a system for managing and searching personal photo collections. The system allows a collection to be stored in a distributed manner on a variety of storage media but kept synchronized and with the entire collection available for at least browsing access from any media. Virtual albums allow users to organize their photos in a variety of flexible ways. We believe that being able to access one's photo collection from a variety of devices and in a variety of organizational views allows users to derive value from their photos. We have also presented a photo search scheme based on time (event), face detection statistics, and visual similarity modified by using the camera metadata. Users use a variety of criteria when searching for relevant photos and different similarity metrics map well to these criteria.

From our experiments, we see that in many cases users tend to group their photos based on time events. One avenue for future work would be to investigate how to use visual similarity and face detection methods to enhance time-based search. Also, it is hypothesized that the use of non-time features such as visual, facial and camera-metadata similarity will be more important for large-time-period collections, so testing with such collections would be interesting. Finally, the use of such methods may allow photos to be grouped and presented to the user in novel ways that increase digital photo usage.

References

[1] H. Oh, J. Park, D. Chang, and G. Oh, "Content-based retrieval system for image using human face information,", *Proc. of SPIE, Storage and Retrieval for Media Databases*, vol. 3972, p. 12-20, 2000.

[2] R. Srihari, Z. Zhang, and A. Rao, "Image background search: Combining object detection techniques with content-based image retrieval (CBIR) systems," *Proc. IEEE Workshop on Content-Based Access of Image and Video*, p. 97-101, 1999.

[3] Z. Zhang, R. Srihari, and A. Rao, "Applications of image understanding in semantics-oriented multimedia information retrieval," *Proc. Of Intl. Symp. On Multimedia Software Engineering*, p. 93-96, 2000.

[4] SyncML Sync protocol, v. 1.0.1, www.syncml.org.

[5] R. Ronfard, C. Garcia, and J. Carrive, "Conceptual Indexing of Television Images Based on Face and Caption Sizes and Locations," Proc. Intl. Conf. on Visual Information Systems, 2000.

[6] A. Kuchinsky, C. Pering, M. L. Creech, D. Freeze, B. Serra, J. Gwizdka, "FotoFile: A Consumer Multimedia Organization and Retrieval System", *Proc. ACM CHI99 Conference on Human Factors in Computing Systems*, pp. 496-503, May 1999.

[7] Kang, H., Shneiderman, B. Visualization Methods for Personal Photo Collections: Browsing and Searching in the PhotoFinder. *Proc. IEEE International Conference on Multimedia and Expo* (ICME2000), New York City, New York.

[8] M.S. Lew, Next Generation Web Searches for Visual Content, IEEE Computer, November, 2000, pp. 46-53.

[9] Rowley, H and Kanade, T. "Human Face Detection in Visual Scenes", Cmu Tech Report CMU-CS-95-158R, Nov 1995.

[10] A. Pentland, R.W. Picard, and S. Sclaroff. Photobook: Content-based manipulation of image databases. *International Journal of Computer Vision*, 18(3):233-254, June 1996.

[11] The EXIF Image Format standard. http://www.pima.net/standards/it10/PIMA15740/exif.htm

[12] Platt, J.C., Czerwinski, M., Field, B.A., PhotoTOC: Automatic clustering for browsing personal photographs. Microsoft Research Technical Report MSR-TR-2002-17.

Table 3 Relevancy Scores from Subject AP

Image ID	57666	57853	58025	57977	57889	58077	57777	58167	57805	57673	Top	Total/avg Hits
Time	1	4	9	6	3	9	14	2	5	4	6/10	57/5.7
Visual	0	3	18	2	6	3	4	1	1	10	2/10	48/4.8
Visual + Camera	3	3	18	4	5	9	6	2	3	12	5/10	65/6.5
Visual+Face + Camera	2	2	17	2	5	3	4	1	1	10	0/10	47/4.7

Table 4 Relevancy Scores from Subject AS

Image ID	54389	54627	54293	54516	54229	54543	54391	54638	54264	54566	Top	Total/avg Hits
Time	-	-	-	-	-	-	-	-	-	-	0/0	0/0.0
Visual	2	16	10	10	14	8	3	3	17	7	3/10	90/9.0
Visual + Camera	4	15	12	11	8	8	6	3	19	7	4/10	93/9.3
Visual+Face + Camera	15	17	17	11	8	5	8	5	18	2	6/10	106/10.6

Table 5 Relevancy Scores from Subject OT

Image ID	54856	54801	55196	55161	54739	55050	55175	54821	55207	55066	Top	Total/avg Hits
Time	-	-	19	12	-	4	-	-	19	-	4/4	54/13.5
Visual	12	19	15	2	9	0	14	18	12	10	6/10	111/11.1
Visual + Camera	11	19	15	4	11	0	14	18	15	10	7/10	112/11.2
Visual+Face + Camera	9	16	13	0	17	0	13	17	12	9	1/10	106/10.6

Table 6 Relevancy Scores from Subject PO

Image ID	57386	57536	32737	54087	54139	57523	32713	56741	32664	56469	Top	Total/avg Hits
Time	19	6	5	5	2	5	5	-	6	-	8/8	53/6.625
Visual	0	0	3	4	0	1	4	7	1	5	2/10	25/2.5
Visual + Camera	3	3	5	5	0	4	3	7	3	5	4/10	38/3.8
Visual+Face + Camera	0	2	4	4	0	2	2	7	3	5	2/10	29/2.9

Table 7 Relevancy Scores for All Subjects

	Time	Visual	Visual+Camera Metadata	Visual+ Face+Camera Metadata
Number of times top method	18/22 = 81.8%	13/40 = 32.5%	20/40 = 50.0%	9/40 = 22.5%
Total/avg. No. of hits	164 / 7.5	274 / 6.9	308 / 7.7	288 / 7.2