(12) STANDARD PATENT APPLICATION (11) Application No. **AU 2005242219 A1**
(19) AUSTRALIAN PATENT OFFICE

(54) Title
Document annotation

(51) International Patent Classification(s)
G06F 17/30 (2006.01)

(21) Application No: **2005242219** (22) Date of Filing: **2005.12.12**

(43) Publication Date: **2007.06.28**
(43) Publication Journal Date: **2007.06.28**

(71) Applicant(s)
Canon Information Systems Research Australia Pty Ltd

(72) Inventor(s)
Wan, Ernest;Amielh, Myriam;Vendrig, Jeroen;Mak, Eileen

(74) Agent / Attorney
Davies Collison Cave, 255 Elizabeth Street, Sydney, NSW, 2000

2005242219 12 Dec 2005

Abstract

A method for use in annotating documents provided in a collection. The method comprises determining, an annotation gain for a number of document clusters, the annotation gain being at least partially indicative of the effect of annotating at least one document associated with the respective cluster. A recommendation can then be determined using the determined annotation gains, the recommendation being indicative of at least one cluster. An indication of the recommendation can then be provided.

Fig. 1

2005242219 12 Dec 2005

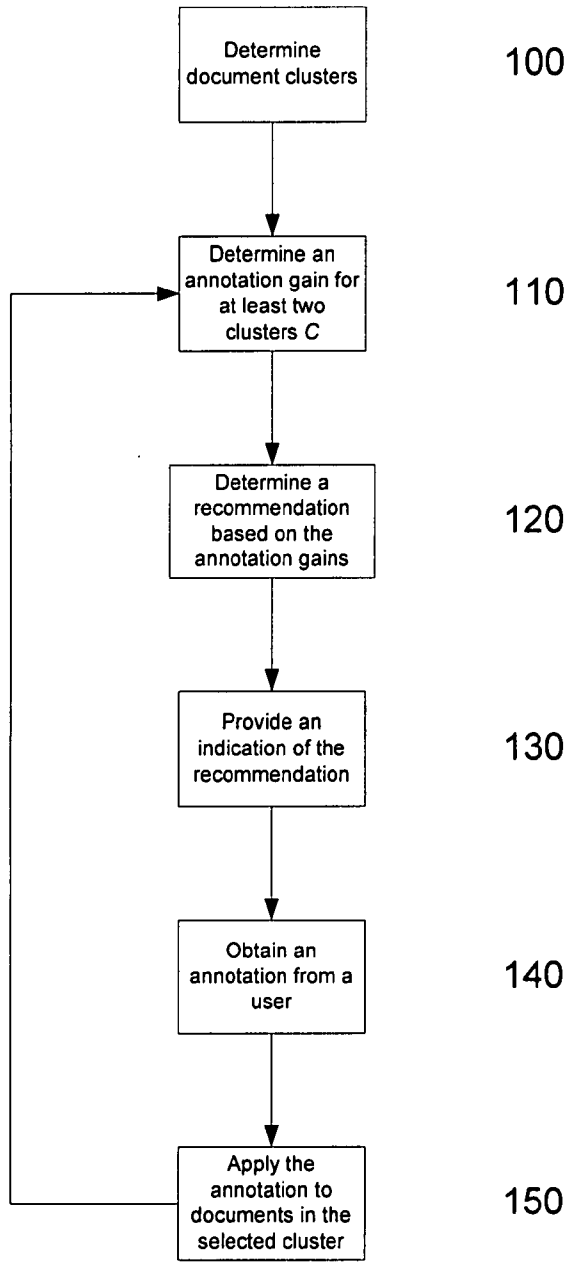

Fig. 1

2005242219 12 Dec 2005

AUSTRALIA

PATENTS ACT 1990

COMPLETE SPECIFICATION

NAME OF APPLICANT(S)::

Canon Information Systems Research Australia Pty Ltd

ADDRESS FOR SERVICE:

DAVIES COLLISON CAVE
Patent Attorneys
Level 10, 10 Barrack Street, Sydney, New South Wales, Australia, 2000

INVENTION TITLE:

Document annotation

The following statement is a full description of this invention, including the best method of performing it known to me/us:-

DOCUMENT ANNOTATION

Field of the Invention

The present invention relates to a method and apparatus for use in annotating documents provided in a collection, and in particular to a method and apparatus for determining a recommendation of a next cluster to be annotated next to maximise improvement in annotation coverage.

Description of the Background Art

The reference to any prior art in this specification is not, and should not be taken as, an acknowledgment or any form of suggestion that the prior art forms part of the common general knowledge.

Document collections are growing quickly in the digital age. For example, the advent of Digital Cameras has made the creation of personal libraries of photographs considerably easier by allowing consumers to view, edit, select and process digital images without being dependent on a photo development service. As a result, users of digital cameras frequently generate and store large collections of personal digital images.

Conversely, the growing volume of document collections makes it harder to find and to use the documents. For example, in personal photo libraries it is uneasy and time-consuming for users to assemble images into albums, slideshows, diaries or calendars, to search for specific images, and to browse through image databases. It is therefore highly desirable to provide methods of organising and accessing sizeable collections of documents.

Using document analysis to automatically generate useful metadata has been tried with limited success. The analysis problem is especially hard when the modality of the documents, for example visual for an image, does not correspond to the modality of the annotations, for example textual for a key word. Therefore, manually entered metadata remains the more effective way to make document collections easily searchable. For that reason, reducing the annotation workload is crucial when collecting user's annotations.

Supervised and semi-supervised learning is an alternative approach for facilitating annotation of multimedia content. An active annotation system can include an active learning component that prompts the user to label a small set of selected example content that allows the labels to be propagated with given confidence levels. Supervised learning approaches require a substantial amount of input data before proving effectiveness. Although it is possible to restrict the annotation vocabulary to learn conceptual models faster, this tends to make the annotation system less appealing.

Moreover, users of digital cameras and personal computers are generally not enthusiastic to spend time making notes on their content. Annotating content is usually regarded as a tedious task that users tend to avoid despite of the benefit. Consequently, a user's initial interest and effort are usually limited and it is therefore important to make use of the user's input as efficiently as possible.

A pro-active approach to annotation consists of automatically partitioning a collection of documents into clusters sharing enough semantic information. User Annotations can then be physically or virtually propagated through document clusters. Thus, it becomes possible to make use of the annotation effort on a cluster basis instead of individual documents. Documents can be grouped from various aspects. For instance, images can be grouped based on the event to which they relate, the people they contain, or the location of the scene. Various methods of time-based and/or content-based partitioning have been developed for the purpose of grouping images according to a particular aspect.

Another approach for making the annotation process easier is the use of summaries. When requesting User's annotations, groups of documents may be summarised, for example because their sizes are not suitable for a particular Graphical User Interface. For example, several approaches for summarising collections of images have been developed in particular for the purpose of video. For example, some methods automatically select a small number of key-images by computing the percentage of skin-coloured pixels in the image or by estimating the amount of motion activity.

However, conventional annotation systems follow an approach where presentation of documents and clusters does not focus on making use of limited user efforts efficiently. For example, they show clusters ordered temporally. If this is used in assigning annotations, this

assumes that the temporal order of the photos defines their relative importance, such that for example, photos of the first or last day of a holiday would be deemed more relevant than photos half way the holiday. As a result, the user ends up either annotating documents that are not necessarily the most relevant, or browsing through the clusters so as to find the relevant clusters. In both cases the required user effort makes it not very attractive for users to annotate a collection.

In addition, conventional systems assume that any annotation is equally valuable. This is often not the case. For example, if a thousand images are annotated with the same value, the annotations are of limited use in applications such as retrieval and browsing. There is not much point in having the user annotate a thousand and first photo with the same annotation.

Consequently conventional annotation systems do not provide the user with needed direction and they fail to make the process of annotating attractive and worthwhile to users.

Summary of the Present Invention

It is an object of the present invention to substantially overcome, or at least ameliorate, one or more disadvantages of existing arrangements.

In a first broad form the present invention provides a method for use in annotating documents provided in a collection, the method comprising, in a processing system:

a) determining, for a number of document clusters, an annotation gain, the annotation gain being at least partially indicative of the effect of annotating at least one document associated with the respective cluster;
b) determining a recommendation indicative of at least one cluster in accordance with the determined annotation gains; and,
c) providing an indication of the recommendation.

Typically the method comprises, in the processing system, determining the annotation gain for a cluster using at least one of:

a) an expected change in annotation coverage obtained by annotating the cluster; and,
b) a user interest indicative of user interactions with at least one of:
    i) documents in the cluster; and,
    ii) documents similar to documents in the cluster.

Typically the method comprises, in the processing system, determining the annotation coverage using at least two of:

a) a document annotation extent;

b) an annotation discriminative power;

c) an annotation effectiveness; and,

d) an annotation efficiency.

Typically the method comprises, in the processing system:

a) determining an annotation impact for a respective cluster by:

i) determining a current annotation effect indicative of the effect of annotating the cluster with existing annotations associated with other documents; and,

ii) determining an added annotation effect indicative of the effect of annotating the cluster with a new annotation; and,

b) using the annotation impact in determining the annotation gain.

Typically the method comprises, in the processing system:

a) for at least one group of documents annotated with an existing annotation:

i) determining a similarity metric between documents in the cluster and the documents in the group;

ii) determining an expected annotation coverage obtained if the documents in the cluster are annotated with the existing annotation; and,

b) determining the current annotation coverage of the document collection; and,

c) determining the current annotation effect using the similarity metric, the current annotation coverage and the expected annotation coverage.

Typically the method comprises, in a processing system:

a) determining a similarity metric between documents in the cluster and a most similar group of previously annotated documents;

b) determining an expected annotation coverage obtained if the documents in the cluster are annotated with a new annotation; and,

c) determining the current annotation coverage of the document collection; and,

d) determining the added annotation effect using the similarity metric, the current annotation coverage and the expected annotation coverage.

Typically the method comprises, in the processing system:

a) determining a confidence score at least partially based on a similarity score determined for each pair of documents in the cluster; and,

b) using the confidence score in determining the annotation gain.

Typically the method comprises, in the processing system, determining the confidence score using at least one of:

a) an indication of user interactions with the documents in the cluster; and,

b) metadata associated with the documents in the cluster.

Typically the method comprises, in the processing system:

a) monitoring user interactions with documents in the cluster; and,

b) determining a user interest score based using the monitored user interactions.

Typically the method comprises, in the processing system:

a) determining a number of interaction types;

b) for each interaction type, determining:

i) a number of user interactions; and,

ii) a relative importance; and,

c) using the determined number of user interactions and the relative importance to determine the user interest score.

Typically the user interaction types comprise at least one of:

a) viewing a document;

b) printing a document;

c) editing a document;

d) e-mailing a document;

e) publishing a document; and,

f) linking to a document.

Typically the method comprises, in the processing system, determining a user interest score using at least one of:

a) a user profile specified by the user; and,

b) a user profile automatically constructed by the processing system.

Typically the method comprises, in the processing system, determining the annotation gain following at least one of:

a) annotation of at least one document;

b) a change in user interest for at least one document;

c) a change in cluster membership, including at least one of:

i) adding a document to a cluster;

ii) removing a document from a cluster; and,

iii) moving a document from one cluster to another cluster;

d) a change in configuration of clusters, including at least one of:

i) adding a cluster;

ii) removing a cluster; and

iii) combining clusters; and,

e) adding a document to the collection; and,

f) removing a document from the collection.

Typically the method comprises, in the processing system:

a) determining a cluster summary for at least one cluster;

b) requesting user annotations for the at least one cluster summary;

c) receiving user annotations via an input; and,

d) applying the user annotations to documents in the cluster represented by the cluster summary.

Typically the method comprises, in the processing system, applying the user annotations by at least one of:

a) duplicating the user annotations;

b) referencing the user annotations; and,

c) storing at least a portion of the user annotations as metadata associated with the documents.

Typically the method comprises, in the processing system, determining clusters using at least one of:

a) a clustering algorithm; and,

b) user input commands received via an input.

Typically the method comprises, using annotations comprising at least one of:

a) at least one category-value pair;

b) a confidence score;

c) information used for calculating a confidence score; and

d) an indicator indicative of the means by which an category-value pair was obtained.

Typically the document is at least one of:

a) an image; and,

b) a video segment.

In a second broad form the present invention provides apparatus for use in annotating documents provided in a collection, the apparatus including a processing system for:

a) determining, for a number of document clusters, an annotation gain, the annotation gain being at least partially indicative of the effect of annotating at least one document associated with the respective cluster;

b) determining a recommendation indicative of at least one cluster in accordance with the determined annotation gains; and,

c) providing an indication of the recommendation.

Typically the processing system comprises at least one of:

a) a store for storing at least one of:

i) documents;

ii) annotations;

iii) references to annotations;

iv) a cluster configuration;

v) cluster summaries;

vi) information at least partially indicative of a confidence score;

vii) information at least partially indicative of user interactions;

b) a display for displaying at least one of:

i) documents;

ii) document clusters;

iii) annotations;

iv) a cluster summary; and,

v) the recommendation; and,

c) an input for receiving at least one of:

i) annotations; and,

ii) user input.

In a third broad form the present invention provides a computer program product for use in annotating documents provided in a collection, the computer program product being formed from computer executable code, which when executed on a suitable processing system is for:

a) determining, for a number of document clusters, an annotation gain, the annotation gain being at least partially indicative of the effect of annotating at least one document associated with the respective cluster;

b) determining a recommendation indicative of at least one cluster in accordance with the determined annotation gains; and,

c) providing an indication of the recommendation.

Brief Description of the Drawings

An example of the present invention will now be described with reference to the accompanying drawings, in which: -

Figure 1 is a flow chart of an example of a process of recommending documents for annotation;

Figure 2 is a schematic diagram of an example of a processing system;

Figures 3A to 3C are a flow chart of a specific example of a process of selecting documents for annotation;

Figure 4 is a graph of an example of an effectiveness function;

Figure 5 is a flow chart of a specific example of a process for determining an annotation coverage;

Figure 6 is a schematic diagram of a number of documents;

Figures 7A and 7B are schematic diagrams of an example of a graphical user interface for presenting documents for annotation; and,

Figure 8 is a flow chart of a specific example of a process for assessing user interest in documents.

Detailed Description Including Best Mode

An example of the process for determining an annotation recommendation will now be described with reference to Figure 1.

At step 100 a number of document clusters are determined. The document clusters may be determined in any one of a number of ways and may for example be defined manually by a user, predefined, or automatically assigned by a processing system as will be described in more detail below.

At step 110 an annotation gain is determined for at least two clusters *C*. The annotation gain is a value that is at least partially indicative of the improvement in annotation coverage that will be provided if some or all of the documents within the cluster are annotated.

At step 120 the annotation gain is used to determine a recommendation as to which one or more clusters *C* should be annotated. The recommendation is then provided to a user at step 130 allowing the user to select a cluster and provide an annotation at step 140. It will be appreciated that in this regard the user does not need to select the same cluster as recommended although in general this will be preferred. At step 150 the annotation is applied to one or more selected documents in the cluster with the process returning to step 100 to determine annotation gains for each cluster.

The documents within clusters generally share certain relationships or characteristics. As a result, the above described process uses the annotation gain for each cluster to estimate the benefit for the user in annotating the documents within the corresponding cluster. This can take into account the user's interest and the ease of using the data in the future, thereby helping the user to focus on clusters and documents which are best annotated first.

In general, the process is performed via a human-computer interface presented on a general-purpose computer system, to allow documents and/or clusters to be viewed, a recommendation to be provided, and annotations to be collected. An example of a suitable general-purpose computer system is shown in Figure 2.

The computer system 200 is formed by a computer module 201, input devices such as a keyboard 202 and mouse 203, and output devices including a printer 215, a display device 214 and loudspeakers 217.

The computer module 201 typically includes at least one processor unit 205, and a memory unit 206, formed for example from semiconductor random access memory (RAM) and read only memory (ROM). The module 201 also includes an number of input/output (I/O) interfaces including an audio-video interface 207 that couples to the video display 214 and loudspeakers 217, and an I/O interface 213 for the keyboard 202 and mouse 203 and optionally a joystick (not illustrated). An I/O interface 208, such as a network interface card (NIC) is also typically used for connecting to the computer to a network (not shown), and/or one or more peripheral devices.

A storage device 209 is provided and typically includes a hard disk drive 210 and a floppy disk drive 211. A magnetic tape drive (not illustrated) may also be used. A CD-ROM drive 212 is typically provided as a non-volatile source of data.

The components 205 to 213 of the computer module 201, typically communicate via an interconnected bus 204 and in a manner that results in a conventional mode of operation of the computer system 200 known to those in the relevant art. Examples of computers on which the described arrangements can be practised include IBM-PC's and compatibles, Sun Sparcstations or the like.

The processes of clustering documents, calculating annotation gains, providing annotation recommendations and annotating documents is typically implemented using software, such as one or more application programs executing within the computer system 200. Typically, the application programs generate a GUI (Graphical User Interface) on the video display 214 of the computer system 200 which displays documents, clusters, annotations, or recommendations.

In particular, the methods and processes are affected by instructions in the software that are carried out by the computer. The instructions may be formed as one or more code modules, each for performing one or more particular tasks. The software may be stored in a computer readable medium, and loaded into the computer, from the computer readable medium, to

allow execution. A computer readable medium having such software or computer program recorded on it is a computer program product. The use of the computer program product in the computer preferably affects an advantageous apparatus for annotating clusters of documents.

The term "computer readable medium" as used herein refers to any storage or transmission medium that participates in providing instructions and/or data to the computer system 200 for execution and/or processing. Examples of storage media include floppy disks, magnetic tape, CD-ROM, a hard disk drive, a ROM or integrated circuit, a magneto-optical disk, or a computer readable card such as a PCMCIA card and the like, whether or not such devices are internal or external of the computer module 201. Examples of transmission media include radio or infra-red transmission channels as well as a network connection to another computer or networked device, and the Internet or Intranets including e-mail transmissions and information recorded on Websites and the like.

Clustering

The term cluster refers to one or more documents selected from the collection of documents or parts thereof according to a particular relationship criterion or to certain characteristics.

Clusters may be generated automatically by a clustering algorithm executed by the computer system 200, created manually by a user, or formed through a combination of automated or manually processes.

In the event that clustering is performed automatically by the computer system 200, the computer system 200 typically executes a predetermined clustering algorithm stored in memory. This can operate to cluster documents using a range of factors, such as a similarity between documents in the collection, a temporal interval between documents in the collection, a spatial interval between documents in the collection, user interactions with the documents, a document quality and fuzzy logic rules.

Thus, in one example, a collection of text documents can be partitioned according to their creation dates, according to their authors or according to the proper nouns used in the document. Alternatively, a collection of digital images may be partitioned according to the

distribution of their time stamps, the event they are associated with, the location they are related to, the people they show, their capture parameters, or the lighting conditions.

In one example, the clustering process produces a hierarchy of clusters where a parent cluster is further partitioned into one or more sub-clusters, wherein each additional level of the hierarchy adds a new set of criteria on the relationship/characteristics of the documents. Such clustering may as well produce overlapping clusters with shared documents.

Thus, for example, a clustering process based on elapsed time between images can produce a number of image clusters forming a first level of hierarchy. Each cluster can be further-divided into clusters of high-similarity images, based on some image analysis results. In this case, the first level of hierarchy represents a temporal relationship, while the second level of hierarchy can be interpreted as representing a location relationship.

Cluster Summaries

When a computer system is used to annotate clusters, it is typically to display the clusters to the user on the display 214, using a GUI. However, in some case the GUI bears graphical limitations meaning it is impractical to display all of the documents within a cluster. Consequently, the computer system 200 typically presents cluster summaries, indicative of the documents in the cluster.

Thus, for example, in the case of digital images, such summaries might include, but are not limited to, montages such as mosaics of images, or sets of selected images. The goal of a summary is to convey the characteristics that are common to the documents in the cluster so that the user can have an idea of the content of the cluster without going through all documents.

In one example, cluster summaries formed from at least one image that is most representative of the cluster to minimize the risk of assigning an irrelevant annotation to other images in the cluster. The optimal number of representative images is estimated through the number of sub-clusters that can be extracted by performing further sub-clustering using more stringent criteria on image similarities.

The sub-clusters produced are typically ordered according to their size, and the *N* largest sub-groups are selected where *N* is the number of most representative images required by the annotation system. For each selected sub-group, images are ranked according to the features they share with other images of said sub-group, their quality, and a measure of the level of user interest on the image. Then, a most representative image is selected based on the ranking.

Annotation

The term *annotation* refers to voice records, keywords, comments, notes or any type of descriptive metadata that can be attached to a document by a user. An annotation may comprise one or more *category-value pairs*, e.g. category="location"-value="France". In the example, the description focuses on measures making use of a category-value pair in an annotation. However, computations may equally apply to the use of multiple category-value pairs of an annotation.

In general, an annotation is applied to and associated with a document. If an annotation is applied to or associated with a document that already has an annotation, the category-value pairs of both annotations can be merged into one annotation.

In one example, documents are annotated by having the computer system 200 acquire annotations from the user. A request may be done actively, that is requesting an annotation for a specific cluster, or passively, that is allowing the user the opportunity to annotate any of the clusters.

The timing of requesting annotations depends on the application workflow. For example, annotations may be requested when uploading photos from a camera to the computer. In another example, the user may be prompted when browsing documents. The user may even be requested to provide annotation when waiting for another process to be completed.

In one example, all clusters are ordered and presented according to their annotation gain. In an alternative embodiment, only those clusters with the highest annotation gain are presented, say the top 3, or the clusters with an annotation gain exceeding a predetermined threshold, say 0.8.

Propagation

Typically, annotations are assigned to all documents in the cluster being annotated. In this example, when the user annotates a cluster summary, the annotation is applied to all documents of the cluster for which the summary was produced. Hence, the annotation may be applied to a document that the user did not see or inspect while providing the annotation.

In one example, annotations are classified into two types. Annotations are treated as the *explicit* type when they are entered by the user and as the *implicit* type when they are propagated from other documents. Accordingly, if a cluster summary is presented to the user, the annotation is explicit for those documents that were visible in the summary, while the annotation is implicit for those documents that are not visible in the summary.

In one example, annotations can be applied by duplication. In another example, annotations are applied by referencing. User annotations and their references can be stored separately from the collection or internally. For instance, in the case of digital images, at least a portion of user annotations or references may be stored as metadata, such as EXIF (Exchangeable Image File format) data of the individual images. EXIF is a standard for storing metadata in image files, comprising many predefined data fields as well as custom fields.

The advantage of duplicating annotations is that the annotation can be used independently of the annotation system. For example, a user may e-mail an image with a duplicated annotation in the EXIF data to a friend. The friend may be able to use the annotation of the image, even though he does not have the annotation system and any separately stored annotation. The advantage of storing a reference is that related annotations may be tracked and that changes to annotations, such as the correction of a spelling mistake can be propagated more easily.

An indicator may be kept to indicate whether an annotation is of the explicit type or of the implicit type. For example, an indicator may be used in the EXIF data of an image to record the type of annotation. Instead of or in addition to an indicator, a confidence score may be used. The confidence score may be computed from a number of factors, including the annotation type, the similarity between documents that have an explicit annotation and documents that have an implicit annotation, and user exposure to the documents. Information at least partially indicative of the confidence score, such as the values that serve as input for computing a confidence score, may be stored in addition to or instead of the confidence

score. For example, the confidence score may be computed from the RGB histogram intersections of images and/or from the proximity of the creation time stamps. In another example, the confidence score may be computed as the number of photos in a cluster summary presented divided by the total number of photos in the cluster. In this case, the first level of hierarchy represents a temporal relationship, while the second level of hierarchy can be interpreted as representing a location relationship.

Annotation Gain

The annotation gain is used to help the user focus on which clusters and documents are best annotated first. In one example, this can be determined by the computer system 200 based on a user interest aspect and/or a system aspect.

The user interest aspect is concerned with ensuring that the clusters and documents being annotated are relevant to the user. The system aspect is concerned with improving annotation coverage of the collection, i.e. improving the accessibility of documents through applications that may make use of the annotations, such as a retrieval system, a browsing system or a summary system. This allows the annotation process to balance user annotation effort and the utility of the annotations.

Accordingly, the user is typically encouraged to apply annotations to clusters for which annotation is expected to improve the utility of the annotations in the collection. For example, if many photos in the collection have the very general annotation "event=holiday", it is considered beneficial to have the user annotate a cluster that has an annotation different from "event=holiday" such as "event=birthday", or a sub-cluster in which the photos have characteristics in common in addition to "event=holiday", e.g. "event=visit to Lake Tekapo".

The process works particularly well as an interactive system, in which the annotation gain is computed iteratively. The system can then make optimal use of the annotation impact measure (to be described later) on which the annotation gain is based. The user may enter several annotations for several clusters in one session, or the user may continue annotating in more than one session, or several users may contribute annotations in several sessions, with the annotation gain being recomputed when the annotation of one or more documents in the collection have changed. This can occur, for example because the user adds or removes a

category-value pair, or because a third party has provided a duplicate of a document in the collection with annotations associated by the third party.

The annotation gain may also be recomputed when user interest for one or more documents in the collection has changed, for example because of editing or browsing of the documents. The annotation gain may also be recomputed following a change in configuration of the clusters, such as when one or more clusters have changed or been recomputed, for example to add, remove or combine clusters. Similarly, the annotation gain may be recomputed when cluster membership is changed, such as when documents are added to or removed from clusters, or moved from one cluster to another cluster. The annotation gain can also be recomputed when documents are added to or removed from the collection.

Specific Example

A specific example of the process for determining an annotation recommendation using the computer system 200 will now be described with more detail with respect to Figures 3A to 3C.

At step 300 a manageable size variable *m* is selected. The manageable size variable *m* is selected to represent a manageable number of documents for annotation. This is used to measure the effectiveness of an annotation, with the annotation being more effective the closer the number of documents that share an annotation is to *m*.

The manageable number will depend on the context in which the annotations are to be used. Consequently, the value of *m* may be a factory setting, say 12 documents or 10% of the total number of documents in the collection, or it may be set by a user or the computer system, depending on factors such as the number of documents that can be displayed on a GUI, a number of documents that may be printed on a piece of paper, the resolution of the computer system display 214, or the like.

At step 305 an annotation coverage is determined. The annotation coverage typically depends on two or more of the following:

- the extent to which documents are annotated,
- the discriminative power of the annotations,
- the effectiveness of the annotations, and

- the efficiency of the annotations.

Each of the four components of annotation coverage (extent, discriminative power, effectiveness, efficiency) will now be described.

For the purpose of this description, it is assumed that the documents are annotated with category-value pairs, in which a specific value is assigned to a respective category. Thus, for example, the category may be event, with the assigned value being holiday, thereby providing a category-value pair a of "event=holiday".

For this example, D is the total collection of documents. Let the || operator compute the number of documents in a collection, cluster or group of documents, for example: $|D|$ is the number of documents in the total collection. Let a be a category-value pair, and let A_d be the set of all category-value pairs associated with document d. Let A_D be the set of all category-value pairs found in the collection. Let D_a be the set of all documents that have category-value pair a.

The extent measures what fraction of documents have an annotation with at least one category-value pair. In one example, the computer system 200 determines this using the following equation:

$$extent(D) = \frac{\sum_{d \in D} \min(1, |A_d|)}{|D|} \qquad (1)$$

The discriminative power measures an annotation's ability in separating clusters. The discriminative power for a category-value pair a is found in the collection can be determined by the computer system 200 as follows:

$$disc\_pow(a, D) = \frac{1}{|D_a|} \qquad (2)$$

A retrieval system or a browser may query a collection of documents for documents that contain a certain annotation a. If the number of documents returned is too large, for example spanning more result pages than a user would be willing to sift through, the annotation is considered to have a relatively low effectiveness for accessing documents. If there is no

annotation for which a document is accessible effectively, there is a need for new annotations to make that document accessible in an effective way.

What is considered effective depends on the context in which the annotations are used or applied, for example, depending on the client applications software using the annotations. Although specific requirements and limitations of client applications are not known, assumptions about the general requirements of such applications can be made. For example, it is not effective for annotations to be shared by more photos than can be presented on a screen or printed on one page. Usability tests may determine what number of documents can share an annotation in an effective way, for example measuring how many images can be presented on a screen legibly.

In one example, the effectiveness measures if the number of documents with the same category-value pair a has a manageable size m. It is determined by the computer system 200 as follows:

$$effectiveness(a,D)=\begin{cases}\left(\frac{m}{|D_a|}\right)^{-q} & \text{if } D_a \le |m| \\ \left(\frac{m}{|D_a|}\right)^{k} & \text{if } |D_a| > m\end{cases} \qquad (3)$$

In this example, q is a constant greater than or equal to 0, which is used for controlling the rate at which effectiveness increases when there are less than m documents associated with category-value pair a. q is usually set to a value smaller than or equal to 1, and in one example, q is set to 0.001, so that more emphasis is placed on the number of documents being manageable.

Factor k is a constant greater than or equal to 1, for controlling the rate at which effectiveness decreases once D_a's size is greater than m and in one example is set to 2.

An example of effectiveness measurements is shown in Figure 4. In this example, the number of documents with annotation a ($|D_a|$) 400 is plotted against the effectiveness 410. In the example, m is set to 8 corresponding to the peak in the graphs 420. in this example, first

graph 430 uses settings for q=0.001 and k=2, as shown at 450. Second graph 440 uses alternative settings q=1 and k=1, as shown at 460.

The efficiency component measures whether annotations have been added without waste. In this example, the efficiency measures whether a document is annotated with the minimal number of category-value pairs. It is computed as follows:

$$efficiency(D) = \begin{cases} 0 & if \mid A_D \mid = 0 \\ \dfrac{\sum_{d \in D, \mid A_d \mid > 0} \mid A_d \mid^{-s}}{\sum_{d \in D} \min(1, \mid A_d \mid)} & otherwise \end{cases} \quad (4)$$

where s is a predefined constant greater than 0, and usually smaller than or equal to 1, which is used for controlling the rate at which efficiency decreases for annotating a document with more than one category-value pair. In one example s is set to 2.

Annotation coverage can be defined using two or more of the four components extent, discriminative power, effectiveness and efficiency, in different ways giving different emphasis to different components in accordance with the requirements of the applications.

In this example, annotation coverage is defined as follows:

$$AnnotationCoverage(D) = \frac{\sum_{d \in D, \mid A_d \mid > 0} \max_{a \in A_d}(effectiveness(a, D))}{\mid D \mid} \cdot efficiency(D) \quad (5)$$

Here the extent is represented by the sum over all documents and the division by the size of the collection. By integrating the components, redundancy of terms is avoided. As the effectiveness and efficiency measures used in this example have already taken into account discriminative power and extent respectively, separate terms for discriminative power and extent are not required in this example. The maximum function is applied to select the category-value pair associated with a document that results in the highest effectiveness.

Typically the annotation coverage is 0 when no documents are annotated and 1 when each document is annotated with exactly one category-value pair and for each category-value pair

in the collection $|D_a|=|D|/m$, that is: each category-value pair a is associated with exactly m documents.

As it may be hard to define a precise value for m, a distribution based on m may be used instead of m as a single value. The expected effectiveness is then computed as follows:

$$\text{expected effectiveness(a, D)} = \sum_{i=1}^{|D_a|} p(m=i) \cdot \textit{effectiveness}(a, D \mid m=i) \qquad (6)$$

where p is the probability that a value i is the maximum manageable size. For example, the probability may be based on a Poisson probability distribution.

The four components of annotation coverage can be combined in a variety of ways without departing from the essence of the current invention.

At step 310 a next cluster C is selected to determine the annotation gain for that respective cluster. The annotation gain can include both system and user aspects, as mentioned above.

The system aspect of annotation gain requires the clusters to be scored to determine which cluster, if it were to be annotated, will result in the greatest improvement in annotation coverage. Predicting the change in annotation coverage is trivial if each added category-value pair is new to the document collection. However, in practice it is more likely that some category-value pairs are reused for several clusters. Therefore, for computing the expected change in annotation coverage, the possibility that an existing category-value pair is reused needs to be taken into account.

This is achieved by calculating an annotation impact that compares the current annotation coverage and the expected annotation coverage after the annotation of the cluster C to be scored. In this example, in computing the expected annotation coverage as a result of annotating a cluster, the collection is not actually annotated with that value. The computation can be seen as running a "what-if" scenario.

In this example, an annotation impact is based on *current annotation effect (CAE)* and/or *added annotation effect* (*AAE*) measures, as follows:

$$\textit{AnnotationImpact}(C,D) = \textit{CAE}(C,D) + \textit{AAE}(C,D) \qquad (7)$$

The CAE measures the effect of annotating the cluster with a category-value pair already associated with some documents in the collection. The AAE measures the effect of annotating a cluster with a category-value pair that has not already been associated to any document in the collection.

Both CAE and AAE are computed by comparing the documents in cluster C to groups of documents D_a assigned a category-value pair a, for all possible category-value pairs.

This allows the annotation impact to be computed based on the correlation between features of the cluster to be scored and the current category-value pairs in the collection, which can be determined using various approaches.

In one example, current category-pairs values are examined against the documents in the cluster. The correlation is then based on the likelihood that a current category-value pair applies to documents in the cluster to be scored. This approach is suitable when it is possible to analyze the content of a document to assess whether an annotation applies to a document. For example, the likelihood may be based on the number of times a previously entered category-value pair occurs in the text documents in the cluster to be scored. In an example of a stolen goods database, the characteristics of an image of art work may be analysed to compute the probability that it belongs to a particular class and style, such as impressionist painting or renaissance sculpture.

However, it is not always possible to assess whether an annotation applies to a document without knowledge about the context of the collection and about the semantics of the annotation. Therefore, in the current example, the likelihood that a previously entered category-value pair a will be applied to the cluster C is computed by comparing certain quantifiable features of documents D_a previously annotated with the category-value pair a to documents in cluster C, to determine a level of similarity. For example, two sets of documents may be compared by calculating the highest, lowest or average similarity of pairs of documents.

The CAE uses the similarity between the cluster to be scored and a set D_a to determine the likelihood that the cluster to be scored will have category-value pair a. The more similar the two sets are, the more likely it is that the cluster to be scored will be annotated with a.

The AAE uses the similarity between the cluster to be scored and the one D_a set that is most similar to the cluster to be scored to determine the likelihood that a novel category-value pair will be used. If the cluster to be scored is not similar to any of the D_a sets, it is likely that the cluster will be annotated with a category-value pair that is not yet in A_D. When no documents have previously been annotated, all clusters to be scored have a likelihood of 1, that is: the annotation will be novel.

Accordingly, at step 315, a next group of documents D_a is selected, with a similarity metric σ between the group of documents D_a and the cluster C being determined at step 320. This can be determined in any one of a number of manners and may therefore examine the content of the documents, other document properties such as the time at which the documents were created, or the like. For example, in a photo collection, photos may be compared by using feature similarity metrics such as colour histogram intersections.

In this example, the similarity metric σ has a value between 0 (documents have nothing in common according to the features employed) and 1 (the sets of documents are the same according to the features employed). As a result, the value of the similarity metric σ can be used as an indication of the probability of the cluster C and documents D_a having the same category-value pair a.

At step 325 a current annotation effect *CAE* representative of the effect of the cluster C being annotated with the annotation a to form a cluster C_a is determined. At step 330 it is determined if all annotated documents D_a have been considered and if not the process returns to step 315 to select a next group of documents have a different category-pair value a.

This is repeated until the cluster C has been compared to groups of documents D_a for all of the category-pair values a, such that the *CAE* is given by:

$$CAE(C,D) = \sum_{a \in A_D} \sigma(C, D_a) \cdot (AnnotationCoverage(D \mid C_a, a \in A_D)) - AnnotationCoverage(D)) \quad (8)$$

At this point the process moves on to step 335 with the computer system 200 operating to determine the documents D_a for which the similarity metric σ has the highest value.

Once this has been determined the computer system 200 determines an added annotation effect *AAE* representative of the effect of the cluster *C* being annotated with a new annotation *b*. In this case, the *AAE* is given by:

$$AAE(C,D) = (1 - \max_{a \in A_D} \sigma(C, D_a)) \cdot (AnnotationCoverage(D \mid C_b, b \notin A_D) - AnnotationCoverage(D)) \quad (9)$$

The first computation of annotation coverage handles the case where *C* is annotated with a novel category-value pair, *b*, while the second computation of AnnotationCoverage handles the case where *C* is not yet annotated with *b*.

In the above calculations it is possible that the resulting annotation impact will have a negative value. In one example, a limit value of 0 is applied to the annotation impact so that adding a category-value pair never has a negative impact. However, in this example, measures may be negative, as it indicates that annotation of the cluster may result in the category-value pair losing its effectiveness and discriminative power.

The algorithm used by the computer system 200 is determining *CAE* and *AAE* will now described in more detail, with reference to Figure 5. In this example, A_D is the set of all annotations found in the collection 510. For each category-value pair *a* of A_D 515, 525, a group of documents D_a is created at 530, comprising all documents that have previously been annotated with annotation *a*. The cluster *C* to be scored, input at 500 is compared with each group D_a at 535 employing the similarity metric σ which takes as input two sets of documents and gives a similarity score between 0 and 1 as output.

In the example of colour photo images, σ is the intersection of normalized three dimensional RGB colour histogram similarity metric for images with 8 bins in each of the three dimensions. If some documents in *C* have already been annotated, σ may take those into account, e.g. returning 1 if a document has category-value pair *a*. If category-value pairs are similar but not the same, σ may return a value of document similarity weighted by a measure of the category-value pair similarity.

In this example, the *CAE* measure is computed by increasing the sum at 505 with the multiplication of similarity score $\sigma(C, D_a)$ (representing the probability that *C* has category-value pair *a*) with the expected change in annotation coverage at 545. The sum is increased

for each a in A_D. Added to the sum is the computation for the AAE measure: 1 minus the highest similarity at 505 for $\sigma(C, G_a)$ found for an category value-pair a in A at 540 (representing the probability that the category-value pair for C cannot be found in A) multiplied by the expected change in annotation coverage at 520.

Figure 6 illustrates a further example for calculating the annotation impact.

In this example, a document collection 600 contains 40 documents 610. Six documents are in a first cluster to be scored C_l 620. Some documents in the collection have been annotated previously, viz. $D_{A=\{x,y\}}$ comprising 3 documents with category-value pair "A={x,y}" 630 and $D_{B=\{u,v\}}$ comprising 6 documents with category-value pair "B={u,v}" 640.

First, the computer system 200 computes a similarity metric σ=0.6 of C_l with $D_{A=\{x,y\}}$ at 650. Then the difference between annotation coverage after and before annotating with "A={x,y}" in the preferred embodiment is computed. The increase of the sum at 545 then is 0.062. This means there is a relatively low probability that the cluster to be scored has category-value pair "A={x,y}", but that even though there are documents already annotated with "A={x,y}", category-value pair of C_l with "A={x,y}" will still contribute to the annotation coverage.

Next, the method computes a similarity metric σ=0.9 of C_l with $D_{B=\{u,v\}}$ at 650. After computing expected change in annotation coverage, the increase of the sum at 545 is computed as -0.015. This means there is a relatively high probability that C_l has category-value pair "B={u,v}", but that there are many documents already annotated with "B={u,v}" so that annotating C_l with "B={u,v}" will actually reduce effectiveness.

Next, the probability of a new category-value pair is based on the highest similarity at 505 of C_l to the annotated groups 630, 640 and it is multiplied by the expected change in annotation coverage for a novel category-value pair: (1 – max(0.6,0.9)) * 0.15 = 0.015.

This means it is not very likely that a new category-value pair will be entered, but taking into account the size of C_l, all documents would contribute maximally to annotation coverage of the collection if it were to be annotated.

The impact of a new category-value pair weighed by the probability of a new category-value pair then is added to the sum, so that the resulting annotation impact at 520 is 0.062 for the first cluster C_1 620 to be scored.

Similarly, the annotation impact for the 5 documents in the second cluster to be scored C_2 625 may be computed. This results in a value of 0.109. Even though C_2 625 contains less documents than C_1 620, it does get a higher score because it is expected that the category-value pairs for C_2 625 are more useful (that is, more novel and more effective) than for C_1 620.

Whilst the annotation impact alone can be used to determine the annotation again, additional factors may also be calculated.

In the current example, the annotation impact is based on annotations propagating to all documents in a cluster. However, particularly in cases where the user provides an annotation after viewing only the cluster summary, the propagation of the annotation to unseen documents in the cluster may not work well. In those cases, the user may elect to have the annotations propagated to part of a cluster only.

Therefore, in this example, the expected change in annotation coverage is not only based on the annotation impact but is also based on the propagation confidence measure, which measures whether documents in the cluster are expected to have the same or similar category-value pairs.

When the input clusters are user defined, it is expected that the documents in a cluster have at least one meaningful category-value pair in common. The propagation confidence can therefore be set to a maximum value of 1. However, in a more practical scenario where clusters are computed automatically, the annotation system needs to cope with cluster errors.

If the clustering method employed computes and returns a confidence score for each found cluster, the confidence score can be used as the value for the propagation confidence measure, normalized if necessary. This may be obtained from document or cluster metadata, depending on the implementation.

If the clustering method does not compute a confidence score, or if it is a third party method that does not make the information available, or if the confidence score provided by the clustering method is deemed unreliable, a value for the propagation confidence measure needs to be computed.

In one example, the propagation confidence score is at least partially based on the similarity between documents in the cluster. Accordingly, at step 350 the computer system 200 determines a similarity score $\sigma(d_1, d_2)$ for each document pair d_1, d_2 in the cluster C.

For example, the similarity score $\sigma(d_1, d_2)$ may be based on colour histogram intersection of photos in a cluster. In another example, the similarity may be based on the appearance of the same face in photos in a cluster. In yet another example, the similarity may be based on the number of words all text documents have in common. If all documents are highly similar to one another, there is a high probability that they have a category-value pair in common. If the documents are not that similar to one another, there is a low probability that they have a category-value pair in common.

The similarity score $\sigma(d_1, d_2)$ is then used to determine a propagation confidence score indicative of the likelihood of an annotation being applied successfully to all documents in the cluster. In this example, the propagation confidence score is computed at step 350 by taking the average of the similarity scores of all possible pairs of documents in the cluster:

$$\text{propagation confidence } (C) = \frac{\sum_{\substack{d_1 \in C,\, d_2 \in C \\ d_1 \neq d_2}} \sigma(d_1, d_2)}{|C| \cdot (|C| - 1)} \qquad (10)$$

In an alternative example, the propagation confidence score is computed by taking the lowest similarity score found amongst the similarity scores of all possible pairs of documents in the cluster.

The propagation confidence score may be affected by user interaction. For example, if a user modifies the automatically generated clusters because of an error in the clustering or because the user interprets the documents differently, the modification increases the propagation confidence as user modifications are expected to increase the quality of the clustering.

For example, the propagation confidence may be set to the maximum, such as 1, when it is expected that the user changes everything to be correct. In another example, the propagation confidence may be adjusted to reflect the case where the user improved the clustering, but did not yet make all necessary changes. For example, if document d_I is moved from cluster A to cluster B, the similarity score between documents already in B and the new member d_I is set to 1, thereby increasing the average of the similarity scores for B and thereby increasing the propagation confidence score.

Once the propagation confidence is calculated, at step 360 the computer system 200 determines the expected change in annotation coverage using the annotation impact and the propagation confidence. In one example, the expected change in annotation coverage is the propagation confidence score multiplied by the annotation impact. In a second example, the propagation confidence score and the annotation impact are normalized first, for example based on the value ranges.

As described before, the annotation gain may not only be based on a system aspect, but on a user interest aspect as well. In this example, user interest for a cluster is based on monitoring user interaction with documents in the cluster. The monitoring may be done in the context of an application that the annotation recommendation scoring method is part of, and/or it may be done by one or more other applications such as a viewer, browser or editor of the documents. The monitoring may be done real-time, or it may be done via analysis of user interaction logs. Examples of monitored user interaction with a document are: editing, viewing, printing, e-mailing, linking to, and publishing on a Web site.

Accordingly, at step 365 a user interest score E is optionally determined for the cluster C to determine the likelihood of the user being interested in the results of annotating the cluster C. This process will be described in more detail below.

The user interest score is then used together with the annotation coverage change to determine an annotation gain for the cluster C at step 370. In this example, a value for the annotation gain measure based on both the system aspect and the user interest is computed by the weighted sum of the expected change in annotation coverage and the score for user interest.

At step 375 it is determined if all clusters have been considered and if not the process returns to step 310 to select a next cluster.

The clusters for which annotation gains are computed may be part of a hierarchy, in which case the annotation gain can be computed for each individual cluster and sub-cluster.

For example, if cluster *A* contains sub-clusters *B* and *C*, sub-cluster *B* contains documents *d1* and *d2*, and sub-cluster *C* documents *d3* and *d4*, then three annotation gains are computed: for *A* (based on documents *d1*, *d2*, *d3*, and *d4*), for *B* (based on documents *d1* and *d2*), and for *C* (based on documents *d3* and *d4*).

In an alternative example, the annotation gain for a parent cluster may be based on the annotation gain of the child clusters, for example for computational speed-up. In the example, the annotation gain for cluster *A* may be the average, minimum or maximum of the annotation gains for clusters *B* and *C*.

Once all clusters have been considered at step 380, the computer system 200 selects one or more clusters *C* based on the calculated annotation gain, for example, by selecting the cluster *C* with the highest annotation gain, or the clusters having an annotation gain exceeding a threshold. The computer system uses this to generate an annotation recommendation indicative of the selected cluster *C* at step 385, which is displayed to the user at step 390.

In general, the representation will typically show the clusters, any cluster structure, such as relationships between clusters in a cluster hierarchy above. The representation may use cluster summaries, as described above, and typically includes an indication of any existing annotations. An example representation is shown in Figures 7A and 7B.

In this example, the GUI 700 displays a number of documents 701 arranged as thumbnails within sub-clusters 702A, 702B, 702C, which are in turn provided within a parent cluster 703. A number of documents 704 are also provided directly within the parent cluster 703 as shown. In the event that the contents of a cluster are presented as a summary by showing only some of the documents 701, and hiding other documents within the cluster, this is represented by a show/all button 705 presented beside the cluster summary. Hidden documents can be viewed by maximising the cluster, either by clicking the show/all button 705, or by using a minimise/maximise control 706 displayed on each cluster.

Figure 7B shows a view of the GUI 700 in which the cluster 701B is maximised. In this example, when a cluster is opened, it extends in a vertical direction, using as much space as required to show all documents at a specified thumbnail size. A cluster may be minimised by clicking on the minimise/maximise control 706. This again displays only the cluster summary documents 701 and the summary button 705, hiding all other documents.

It will be appreciated that the GUI 700 visually represents the relationship between clusters 703 and their sub-clusters 702 based on the relative arrangement. This allows users to alter the clustering of documents if this is not suitable by moving documents between clusters, or creating new clusters into which they may move documents. When new clusters are added, these are sorted to their appropriate position in the collection. For clusters of digital images, this can be achieved by sorting in time order based on the shooting date of the first image.

An annotation recommendation is provided shown generally as an icon at 708. In this example, if the cluster 702C is recommended as the preferred cluster for annotation. If the cluster 702C is displayed on the GUI, then the annotation recommendation can be shown in the form of a star, or other icon, presented within the cluster 702C. In the event that the recommended cluster is not presented on the GUI, for example if it has been displaced due to the expansion of another cluster 702B, as shown in Figure 7B, then the annotation recommendation 708 includes an arrow indicating the direction of the recommended cluster 702C. This allows the user to scroll the GUI 700 in the direction of the arrow, for example, by using a scroll bar 709, until the recommended cluster is displayed.

The user can then elect to provide an annotation via a suitable mechanism, such as selecting a menu option, selecting the recommendation icon 708, or double-clicking on the cluster to be annotated.

At step 395 the user provides an annotation using a suitable input mechanism, with the annotation being duplicated or linked to each of the documents in the cluster *C*. The process can then return to step 305 to allow further annotations to be received.

To encourage the user to make more annotations, feedback can also be provided on the representations. This provides users of an indication of the effect of previous user actions,

giving the user the sense that the user is making progress, thereby maintaining enthusiasm to annotate more documents.

In this example, an annotation level of the sub-clusters 702 and the parent cluster 703, is represented by a progress bar 707, as shown, although additional or alternative representations, such as numerical or percentage values, can be used. The annotation level represents a current level of annotation coverage provided by the cluster, and accordingly, as annotations are provided, the computer system 200 updates the progress bars 707, to reflect the new annotation coverage.

The annotation coverage may be determined in any one of a number of ways, and can depend for example on factors such as the cluster's contribution to the overall annotation coverage, or the like.

In one example, the annotation coverage of a cluster C in the context of a collection of documents D using a modified version of equation (5), as follows:

$$AnnotationCoverage(C,D) = \frac{\sum_{d \in C, |A_d|>0} \max_{a \in A_d}(effectiveness(a,D))}{|C|} \cdot efficiency(C) \qquad (11)$$

Alternative examples for cluster annotation coverage are similar to the examples for collection annotation coverage described above.

Apart from communicating to the user which clusters are best annotated to maximize annotation gain, this also helps the user to communicate which category-value pairs are best used for annotating the cluster.

For example, in a user interface where the user can select category-value pairs that previously have been associated with other clusters and documents, the list of category-value pairs may be ranked so that the category-value pair resulting in the highest annotation gain when applied to the selected cluster is presented first.

In this example, the computations employed in the *CAE* measure is reused to score annotations for each category-value pair a in A_D given a selected set of documents C in the context of document collection D:

$$annotation\_score(a, C, D) = AnnotationCoverage(D \mid C_a, a \in A_D) - AnnotationCoverage(D) \quad (12)$$

Similarly, the score for annotating the selected cluster with a novel category-value pair can be computed.

Figure 8 is an example of a process for determining a user interest for documents in a cluster.

In this example, the user interest is determined by weighing the amount of interactions (such as the number of times documents in the cluster have been viewed) with a measure of the importance of those interactions (such as the number of seconds the document was viewed).

For example, when a user spends one second on viewing a document, it may indicate that the user evaluated the document quickly but that it is not an important document to the user, e.g. a bad image. In another example, a user views a document for ten seconds and edits it, indicating that the document is important to the user and of great interest.

To achieve this, at step 800 a number of user interaction types I_x are selected. This can then be achieved for example by allowing a user to manually select various options from a drop down list, define new interaction types, or alternatively these can be predefined in the software provided on the computer system 200. Example interactions include viewing (I_1), editing (I_2) and printing (I_3) documents.

At step 810 an evaluation importance function E_x is determined for each interaction type. This is used to define the relative importance of the different types of interaction.

At step 820 the computer system 200 monitors user interactions with the documents this will typically be achieved by maintaining a record of user interaction with each document in the document collection. Thus for example, this can be in the form of a database that records information at least partially indicative of the user interactions, such as the number of interactions for each interaction type, for each document. Alternatively, details of the interaction can also be recorded.

For example, if the document was viewed but less than a number of seconds determined by a viewing threshold, say 2, E_1 may return 0.0. If the document was not viewed at all, E_1 may return 0.5. And if the document was viewed for a number of seconds greater than or equal to

the viewing threshold, E_1 may return 1.0. For example, E_2 and E_3 may return 1.0 if the user interaction was performed and 0.0 otherwise.

At step 830 the computer system 200 determines the user interest E_x for documents in the cluster C based on the number of interactions that have occurred. In this example, the user interest is defined as the sum of the values returned by E_x for all documents in the cluster, divided by the number of documents in the cluster. The terms E_x are weighted, where the weights represent the importance of user interaction of type x in general and how much a user is willing to spend on a document.

The weights may be factory settings based on heuristics, or they may be set by the user. For example, an edit interaction costs the user more in effort than simply viewing a document, and printing even costs the user in ink and paper. Then the weight for I_1 (viewing) may be set to 0.2, the weight for I_2 (editing) to 0.4, and the weight for I_3 (printing) to 0.6.

At step 840 the computer system 200 may also optionally determine a user interest E_x for similar documents in other clusters. This is used to assess the users interest in documents in the cluster by considering not only those documents but also other documents of a similar nature. This may be applied to allow the computer system 200 to determine E_x for a limited number of other documents, such as the top 10 most similar documents, all documents having a similarity metric σ higher than a threshold such as 0.9, or the like.

In a further variation, the user interest can also be based on the quality of content of the documents, as shown at step 850. This is performed on the basis that the user is more interested in high quality documents (eg. nice photos or well-written texts) than in bad quality documents. Examples of measuring the quality of a document are analysis of edges in a photo (where weak edges indicate out of focus or motion blur) or analysis of the number of spelling and grammar errors and use of passive voice in a text document.

In a further example, user interest is based on a user profile specified by the user and/or automatically constructed by the system. For example, the user create a user profile stating that he or she is interested only in video segments that last no more than 30 seconds. In another example, the system creates a user profile by observing that the user never watches video segments that last more than 30 seconds.

The foregoing embodiment describes a method of acquiring user annotations for databases of digital image documents. The term document includes any electronic document such as text, audio and video documents, CAD designs of buildings and molecules, models of DNA sequences, DNA sequence listings or the like, and in particular includes any multimedia content items, such as photos, video segments, audio segments, or the like.

However, the techniques are also applicable to applications such as Integrated Development Environments (IDE), which allow a programmer to write, compile, edit, test and/or debug source code. Often programmers do not have the time to write comments for all pieces of code. It is important that they focus their commenting efforts on those pieces of code that are most in need of further description.

In this case, the documents consist of declarations of classes, variables, methods, etc. The annotations are text comments that the programmer associated with these declarations to explain and document their purpose. A source code comment may consist of a few lines of text associated with one of the comment fields such as @version, @author, @param, @return, @exception, etc. commonly used for documenting Application Programming Interfaces (APIs). Such source code comments can be equated to category-value pairs of annotations.

During the editing of the source code, an IDE application may recommend the programmer to enter comments for those parts of the source code that appear to be key components of the software. The annotation gain is based on both a measure of the user interest and a measure of the expected change in annotation coverage.

In this scenario, the user interest can be a function of the time the programmer spent on editing, updating and debugging a piece of code while annotation coverage can be computed based on the proportion of source documents that has been annotated (a.k.a extent) and/or on a measure of the number of pieces of source code an annotation is associated with (a.k.a. discriminative power), and/or on the portion of declarations of a source document that has non-empty comment fields (a.k.a efficiency).

In this case, when computing the annotation coverage, the comment of a declaration may also be weighted according to a measure of the importance and/or complexity of the associated

code. The importance measure may be based on the number of times the code is referenced or called while the complexity measure may be based on the size of the code and the number of conditional statements it contains.

Throughout the above description the term cluster is understood to refer to any group of one or more documents. Thus, the proposed methods may be applied to single documents, as it is the same as populating each cluster with one document. In this case, the propagation confidence would always be 1 and other measures would be computed as described above.

The term processing system is understood to encompass the computer system 200, as well as any other suitable processing system, such as a set-top box, PDA, mobile phone, or the like.

The foregoing describes only some embodiments of the present invention, and modifications and/or changes can be made thereto without departing from the scope and spirit of the invention, the embodiments being illustrative and not restrictive.

In the context of this specification, the word "comprising" means "including principally but not necessarily solely" or "having" or "including", and not "consisting only of". Variations of the word "comprising", such as "comprise" and "comprises" have correspondingly varied meanings.

2005242219 12 Dec 2005

THE CLAIMS DEFINING THE INVENTION ARE AS FOLLOWS:

1) A method for use in annotating documents provided in a collection, the method comprising, in a processing system:
   a) determining, for a number of document clusters, an annotation gain, the annotation gain being at least partially indicative of the effect of annotating at least one document associated with the respective cluster;
   b) determining a recommendation indicative of at least one cluster in accordance with the determined annotation gains; and,
   c) providing an indication of the recommendation.

2) A method according to claim 1, wherein the method comprises, in the processing system, determining the annotation gain for a cluster using at least one of:
   a) an expected change in annotation coverage obtained by annotating the cluster; and,
   b) a user interest indicative of user interactions with at least one of:
      i) documents in the cluster; and,
      ii) documents similar to documents in the cluster.

3) A method according to claim 2, wherein the method comprises, in the processing system, determining the annotation coverage using at least two of:
   a) a document annotation extent;
   b) an annotation discriminative power;
   c) an annotation effectiveness; and,
   d) an annotation efficiency.

4) A method according to claim 1, wherein the method comprises, in the processing system:
   a) determining an annotation impact for a respective cluster by:
      i) determining a current annotation effect indicative of the effect of annotating the cluster with existing annotations associated with other documents; and,
      ii) determining an added annotation effect indicative of the effect of annotating the cluster with a new annotation; and,
   b) using the annotation impact in determining the annotation gain.

5) A method according to claim 4, wherein the method comprises, in the processing system:
   a) for at least one group of documents annotated with an existing annotation:
      i) determining a similarity metric between documents in the cluster and the documents in the group;

12 Dec 2005 2005242219

ii) determining an expected annotation coverage obtained if the documents in the cluster are annotated with the existing annotation; and,

b) determining the current annotation coverage of the document collection; and,

c) determining the current annotation effect using the similarity metric, the current annotation coverage and the expected annotation coverage.

6) A method according to claim 4, wherein the method comprises, in a processing system:

a) determining a similarity metric between documents in the cluster and a most similar group of previously annotated documents;

b) determining an expected annotation coverage obtained if the documents in the cluster are annotated with a new annotation; and,

c) determining the current annotation coverage of the document collection; and,

d) determining the added annotation effect using the similarity metric, the current annotation coverage and the expected annotation coverage.

7) A method according to claim 1, wherein the method comprises, in the processing system:

a) determining a confidence score at least partially based on a similarity score determined for each pair of documents in the cluster; and,

b) using the confidence score in determining the annotation gain.

8) A method according to claim 7, wherein the method comprises, in the processing system, determining the confidence score using at least one of:

a) an indication of user interactions with the documents in the cluster; and,

b) metadata associated with the documents in the cluster.

9) A method according to claim 1, wherein the method comprises, in the processing system:

a) monitoring user interactions with documents in the cluster; and,

b) determining a user interest score based using the monitored user interactions.

10) A method according to claim 9, wherein the method comprises, in the processing system:

a) determining a number of interaction types;

b) for each interaction type, determining:

i) a number of user interactions; and,

ii) a relative importance; and,

c) using the determined number of user interactions and the relative importance to determine the user interest score.

11) A method according to claim 10, wherein the user interaction types comprise at least one of:

a) viewing a document;

b) printing a document;

c) editing a document;

d) e-mailing a document;

e) publishing a document; and,

f) linking to a document.

12) A method according to claim 9, wherein the method comprises, in the processing system, determining a user interest score using at least one of:

a) a user profile specified by the user; and,

b) a user profile automatically constructed by the processing system.

13) A method according to claim 1, wherein the method comprises, in the processing system, determining the annotation gain following at least one of:

a) annotation of at least one document;

b) a change in user interest for at least one document;

c) a change in cluster membership, including at least one of:

i) adding a document to a cluster;

ii) removing a document from a cluster; and,

iii) moving a document from one cluster to another cluster;

d) a change in configuration of clusters, including at least one of:

i) adding a cluster;

ii) removing a cluster; and

iii) combining clusters; and,

e) adding a document to the collection; and,

f) removing a document from the collection.

14) A method according to claim 1, wherein the method comprises, in the processing system:

a) determining a cluster summary for at least one cluster;

b) requesting user annotations for the at least one cluster summary;

c) receiving user annotations via an input; and,

d) applying the user annotations to documents in the cluster represented by the cluster summary.

12 Dec 2005 2005242219

15) A method according to claim 14, wherein the method comprises, in the processing system, applying the user annotations by at least one of:

a) duplicating the user annotations;

b) referencing the user annotations; and,

c) storing at least a portion of the user annotations as metadata associated with the documents.

16) A method according to claim 1, wherein the method comprises, in the processing system, determining clusters using at least one of:

a) a clustering algorithm; and,

b) user input commands received via an input.

17) A method according to claim 1, wherein the method comprises, using annotations comprising at least one of:

a) at least one category-value pair;

b) a confidence score;

c) information used for calculating a confidence score; and

d) an indicator indicative of the means by which an category-value pair was obtained.

18) A method according to claim 1, wherein the document is at least one of:

a) an image; and,

b) a video segment.

19) Apparatus for use in annotating documents provided in a collection, the apparatus including a processing system for:

a) determining, for a number of document clusters, an annotation gain, the annotation gain being at least partially indicative of the effect of annotating at least one document associated with the respective cluster;

b) determining a recommendation indicative of at least one cluster in accordance with the determined annotation gains; and,

c) providing an indication of the recommendation.

20) Apparatus according to claim 19, wherein the processing system comprises at least one of:

a) a store for storing at least one of:

i) documents;

ii) annotations;

iii) references to annotations;

iv) a cluster configuration;

v) cluster summaries;

vi) information at least partially indicative of a confidence score;

vii) information at least partially indicative of user interactions;

b) a display for displaying at least one of:

i) documents;

ii) document clusters;

iii) annotations;

iv) a cluster summary; and,

v) the recommendation; and,

c) an input for receiving at least one of:

i) annotations; and,

ii) user input.

21) A computer program product for use in annotating documents provided in a collection, the computer program product being formed from computer executable code, which when executed on a suitable processing system is for:

a) determining, for a number of document clusters, an annotation gain, the annotation gain being at least partially indicative of the effect of annotating at least one document associated with the respective cluster;

b) determining a recommendation indicative of at least one cluster in accordance with the determined annotation gains; and,

c) providing an indication of the recommendation.

DATED this TWELFTH Day of DECEMBER 2005

CANON INFORMATION SYSTEMS RESEARCH AUSTRALIA PTY LTD

Patent Attorneys for the Applicant

DAVIES COLLISON CAVE

2005242219 12 Dec 2005

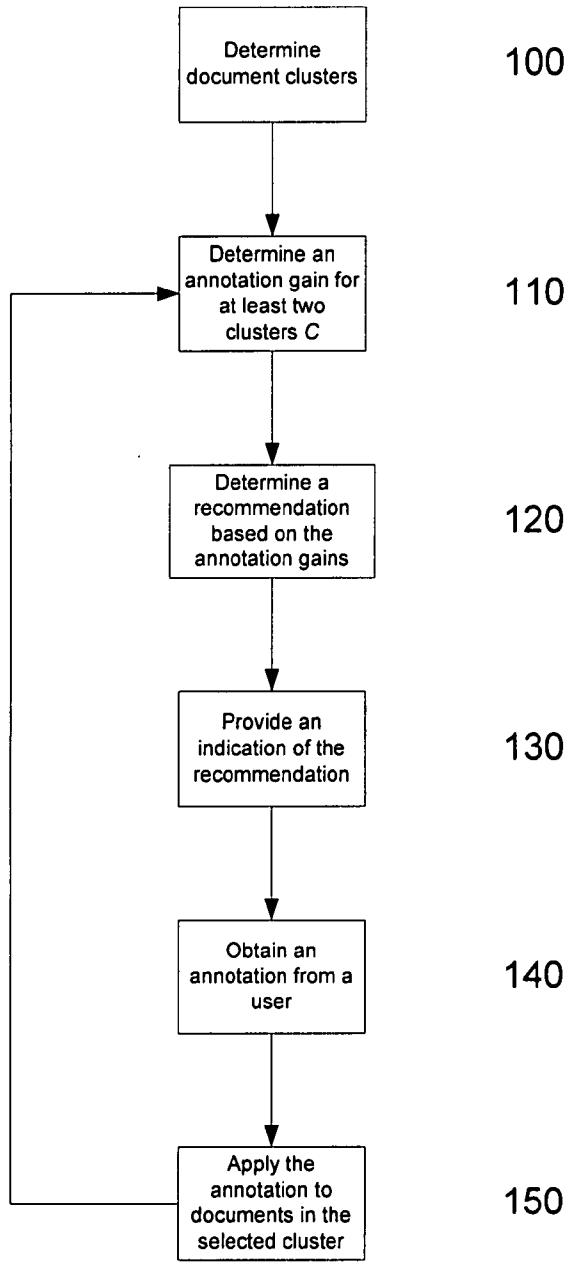

Fig. 1

2005242219 12 Dec 2005

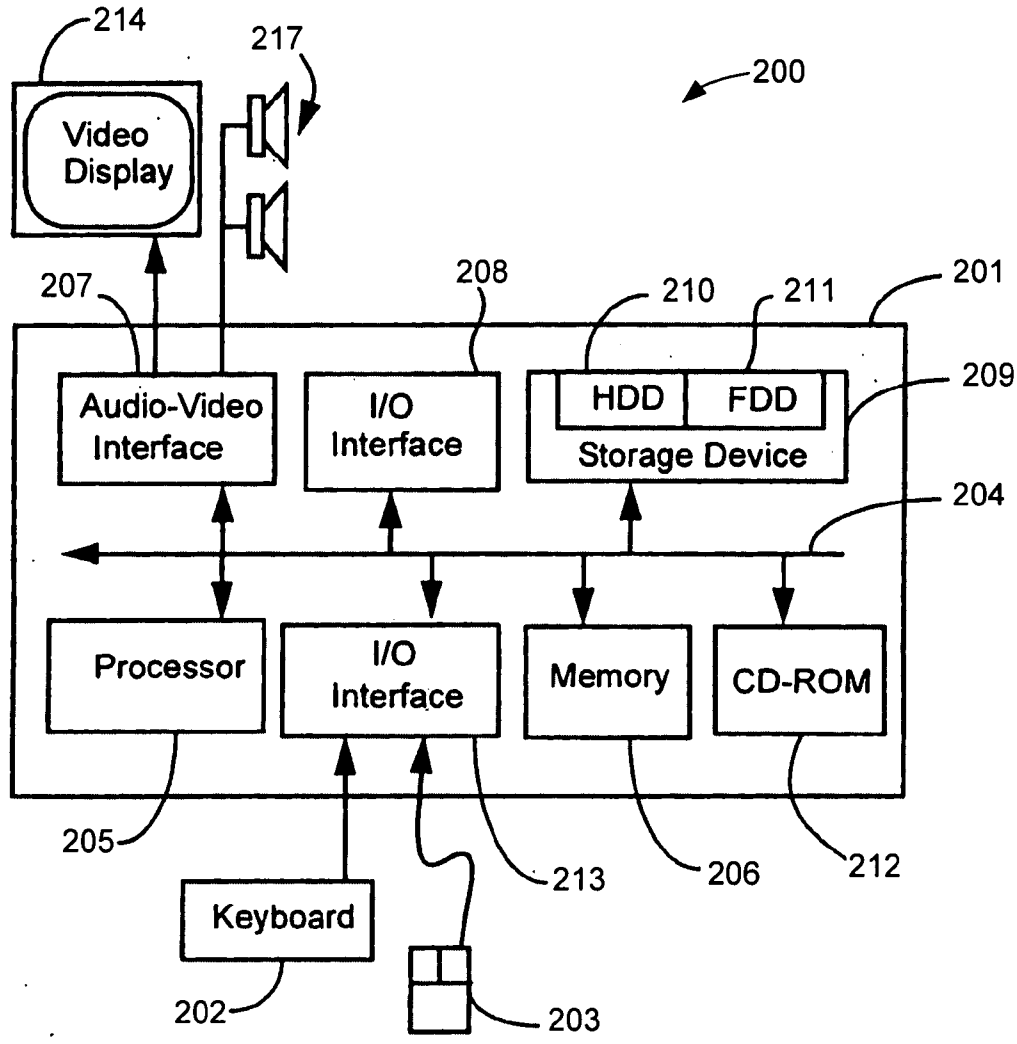

Fig. 2

2005242219 12 Dec 2005

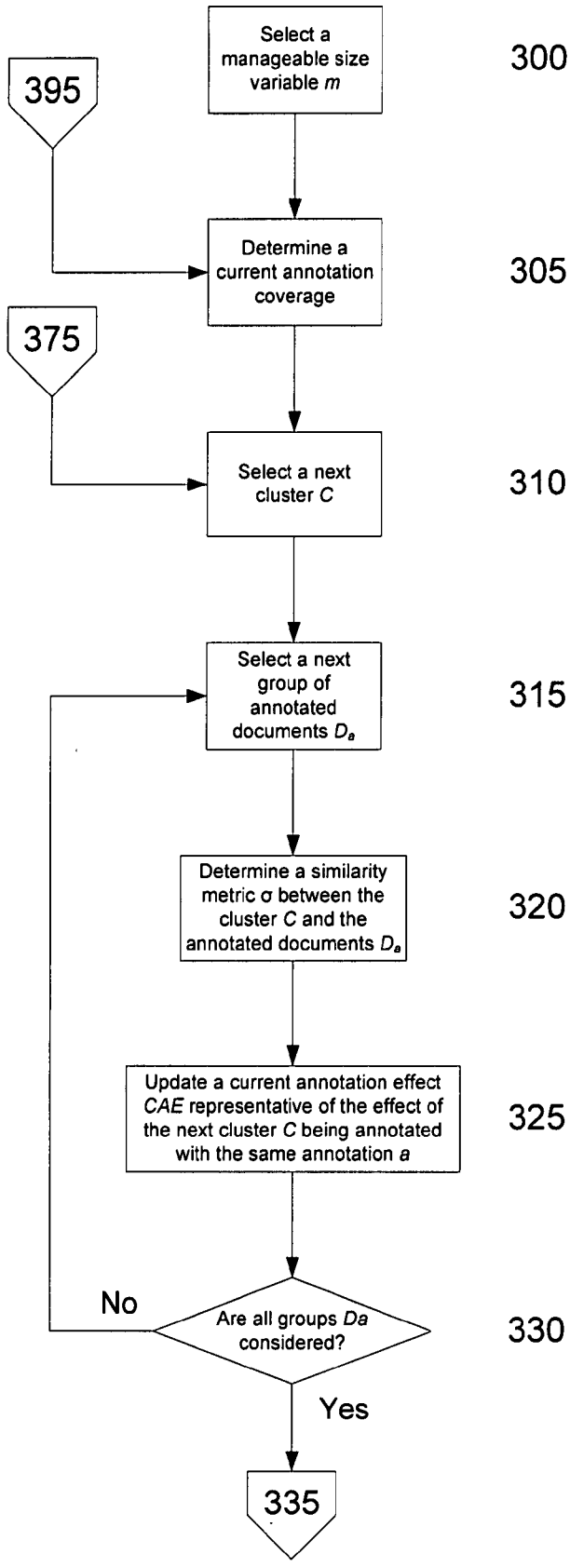

Fig. 3A

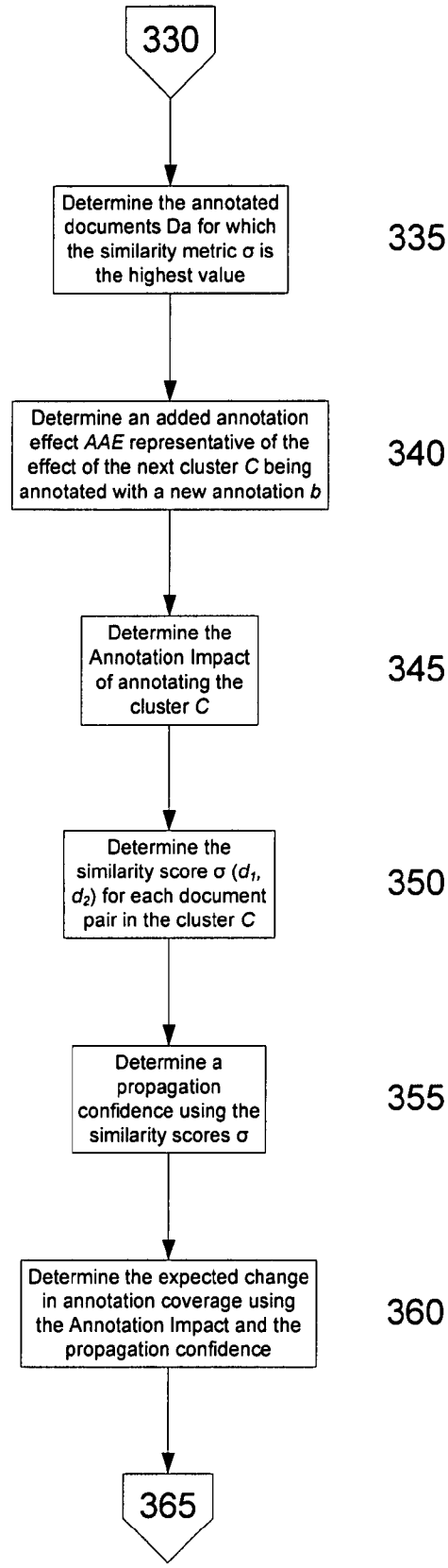
330
Determine the annotated documents Da for which the similarity metric σ is the highest value
335
Determine an added annotation effect AAE representative of the effect of the next cluster C being annotated with a new annotation b
340
Determine the Annotation Impact of annotating the cluster C
345
Determine the similarity score σ (d1, d2) for each document pair in the cluster C
350
Determine a propagation confidence using the similarity scores σ
355
Determine the expected change in annotation coverage using the Annotation Impact and the propagation confidence
360
365

Fig. 3B

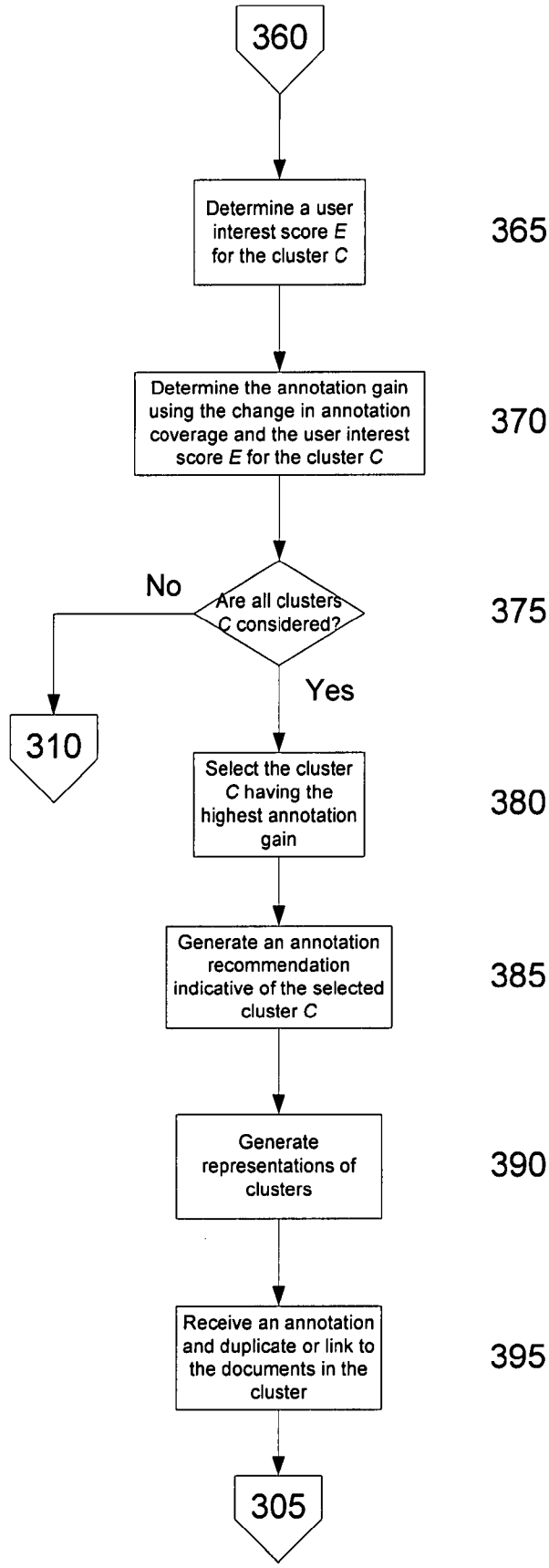
360
Determine a user interest score E for the cluster C
365
Determine the annotation gain using the change in annotation coverage and the user interest score E for the cluster C
370
No
Are all clusters C considered?
375
Yes
310
Select the cluster C having the highest annotation gain
380
Generate an annotation recommendation indicative of the selected cluster C
385
Generate representations of clusters
390
Receive an annotation and duplicate or link to the documents in the cluster
395
305

Fig. 3C

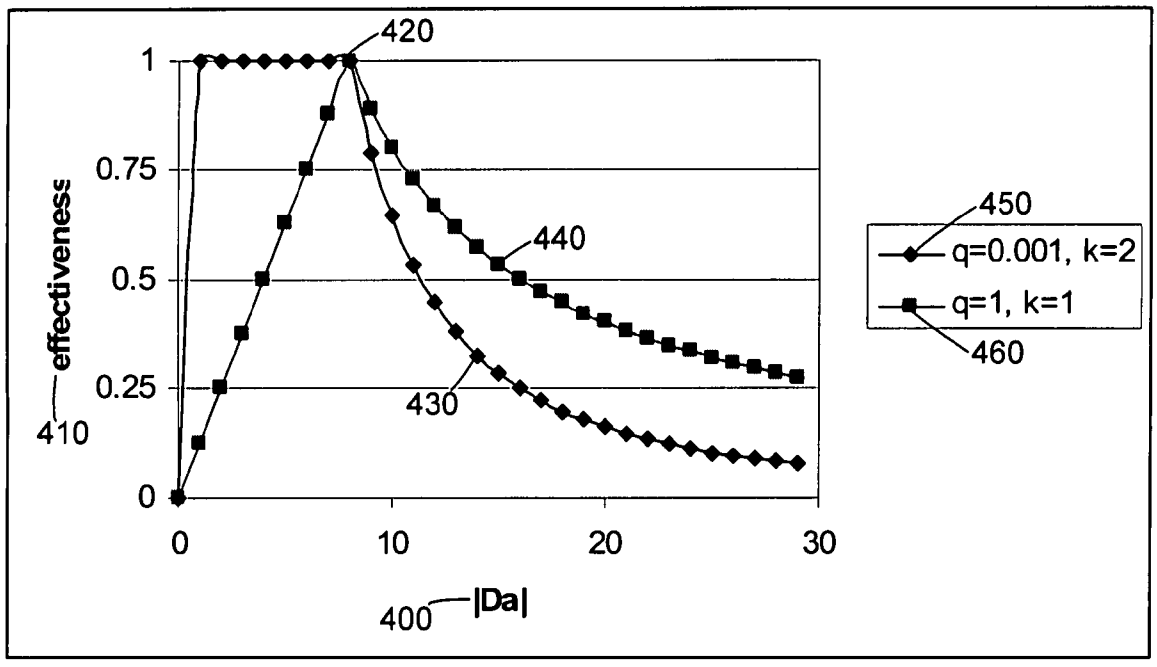

Fig. 4

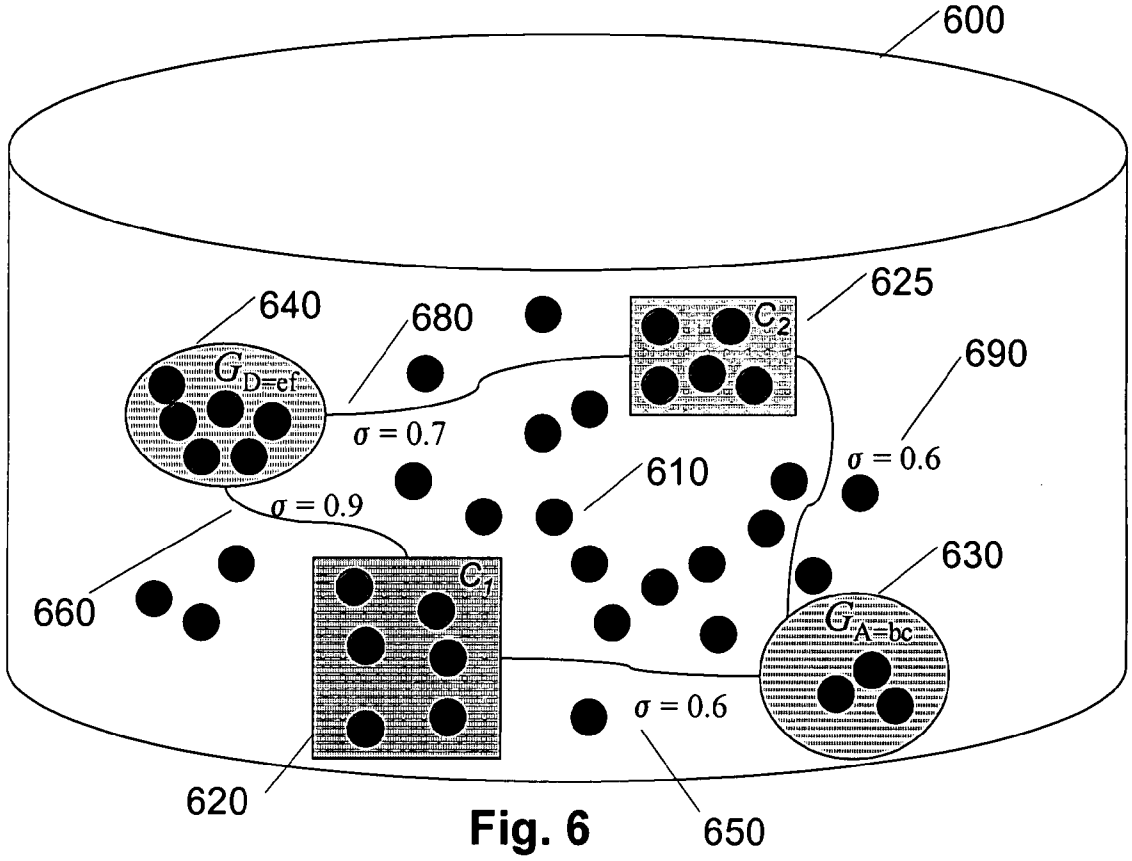

Fig. 6

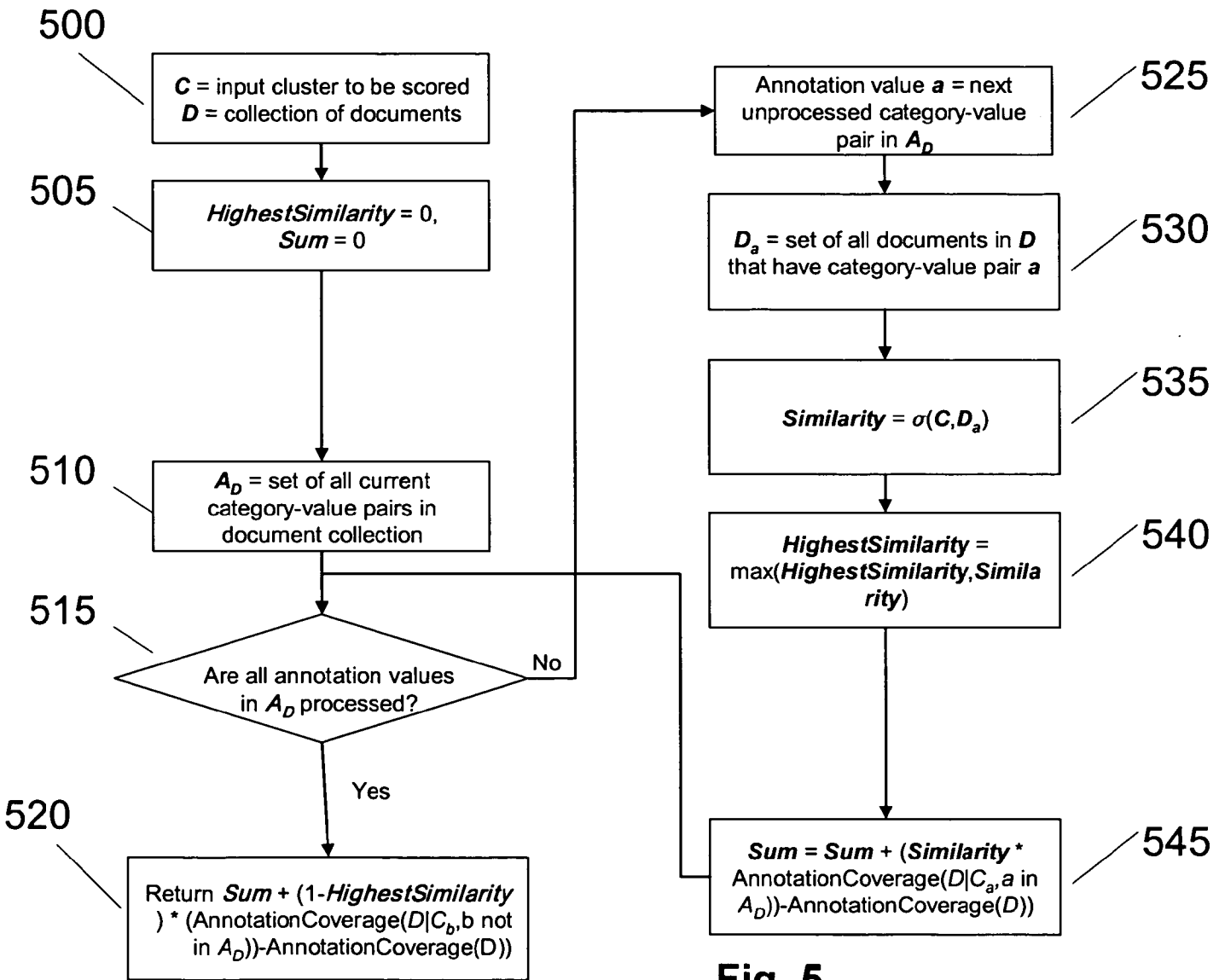
500
C = input cluster to be scored
D = collection of documents
505
HighestSimilarity = 0,
Sum = 0
510
A_D = set of all current category-value pairs in document collection
515
Are all annotation values in A_D processed?
No
Yes
520
Return Sum + (1-HighestSimilarity) * (AnnotationCoverage($D|C_b$,b not in A_D))-AnnotationCoverage(D))
525
Annotation value a = next unprocessed category-value pair in A_D
530
D_a = set of all documents in D that have category-value pair a
535
Similarity = $\sigma(C,D_a)$
540
HighestSimilarity = max(HighestSimilarity,Similarity)
545
Sum = Sum + (Similarity * AnnotationCoverage($D|C_a$,a in A_D))-AnnotationCoverage(D))

Fig. 5

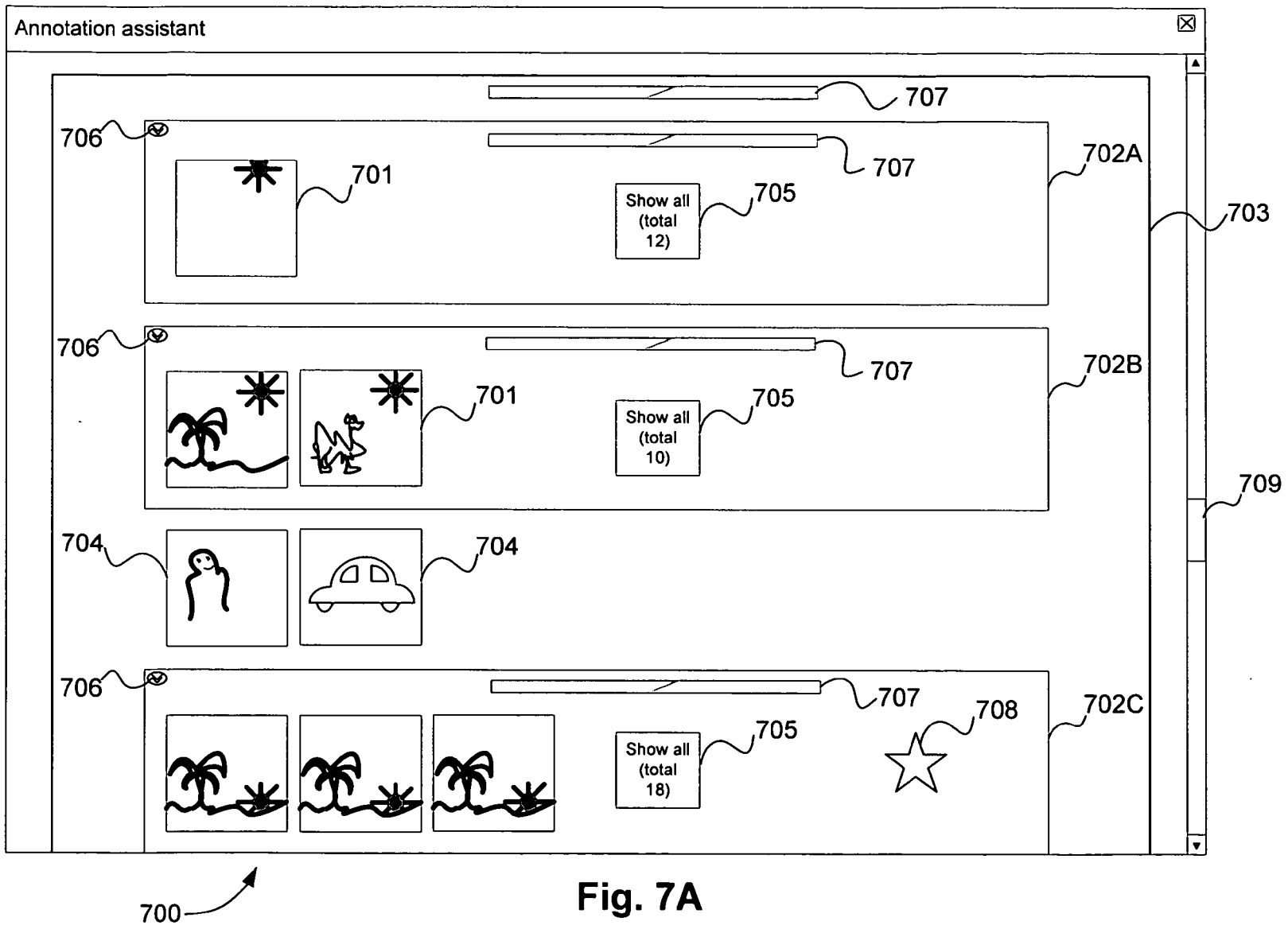
Annotation assistant
707
706
707
701
Show all (total 12)
705
702A
703
706
707
701
Show all (total 10)
705
702B
704
704
706
707
708
Show all (total 18)
705
702C
709
700

Fig. 7A

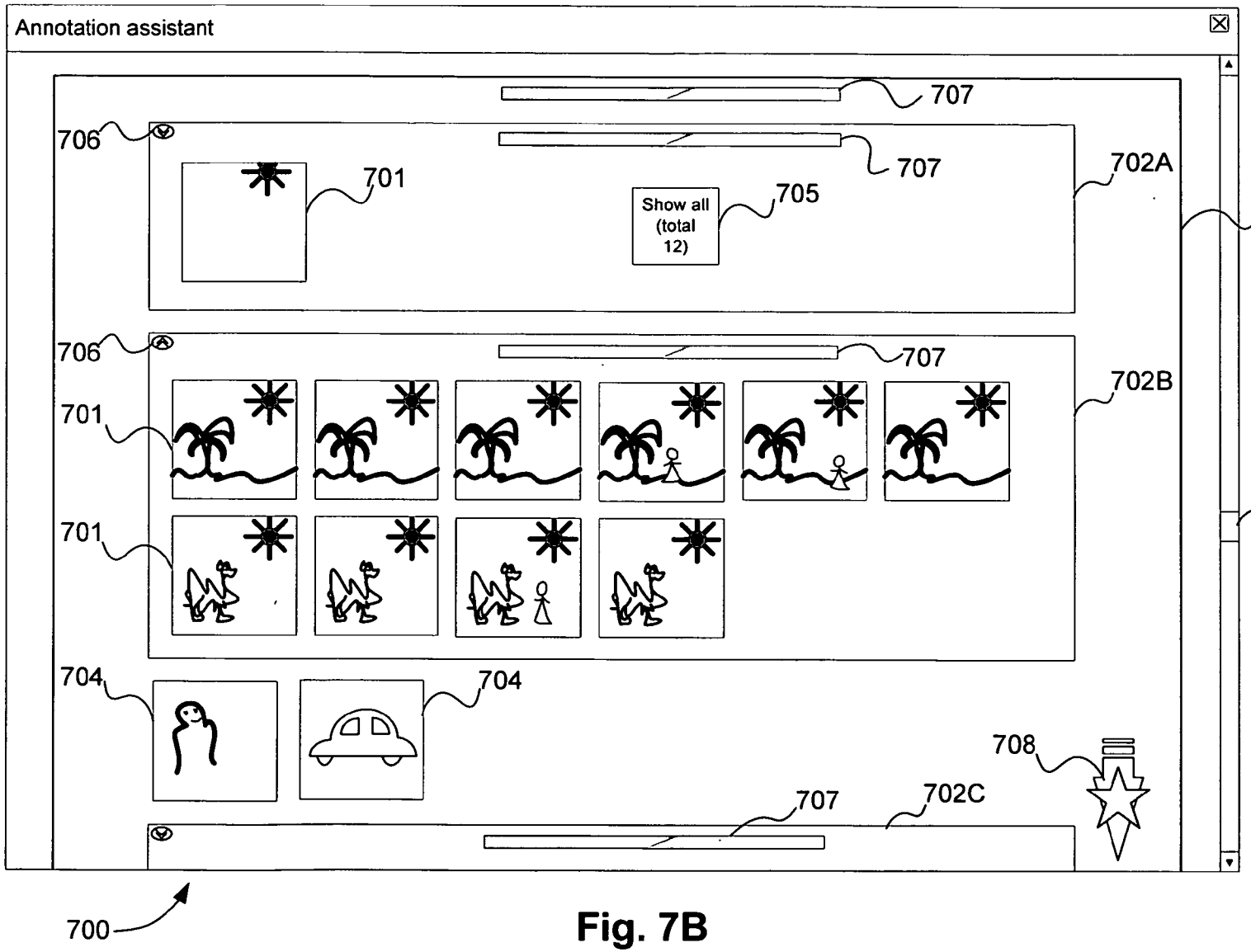
Annotation assistant
707
706
701
707
Show all (total 12)
705
702A
703
706
707
702B
701
701
709
704
704
708
707
702C
700

Fig. 7B

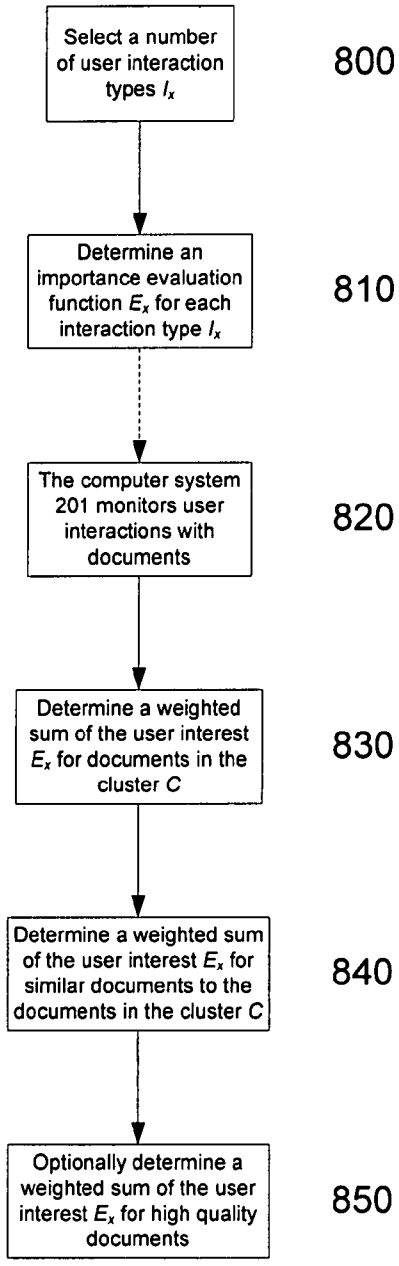
Select a number of user interaction types I_x
800
Determine an importance evaluation function E_x for each interaction type I_x
810
The computer system 201 monitors user interactions with documents
820
Determine a weighted sum of the user interest E_x for documents in the cluster C
830
Determine a weighted sum of the user interest E_x for similar documents to the documents in the cluster C
840
Optionally determine a weighted sum of the user interest E_x for high quality documents
850

Fig. 8