



(51) International Patent Classification:
G06K 9/00 (2006.01)

(21) International Application Number:
PCT/US2011/056441

(22) International Filing Date:
14 October 2011 (14.10.2011)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
12/906,106 17 October 2010 (17.10.2010) US

(71) Applicant (for all designated States except US): **CANON KABUSHIKI KAISHA** [JP/JP]; 30-2, Shimomaruko, 3-chome, Ohta-ku, Tokyo 146-8501 (JP).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **DENNEY, Bradley** [US/US]; c/o Canon U.S.A., Inc., IP Division, 15975 Alton Parkway, Irvine, California 92618 (US). **KORAT-TIKARA-BALAN, Anoop** [IN/US]; c/o Canon U.S.A., Inc., IP Division, 15975 Alton Parkway, Irvine, California 92618 (US).

(74) Agents: **SEELEY, Bartholomew et al.**; c/o Canon U.S.A., Inc., IP Division, 15975 Alton Parkway, Irvine, California 92618 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

- with international search report (Art. 21(3))
- before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))

(54) Title: SYSTEMS AND METHODS FOR CLUSTER VALIDATION

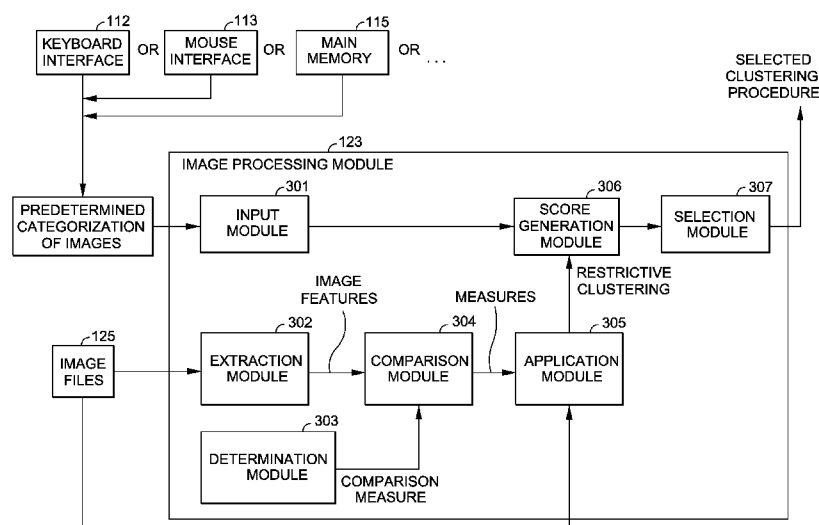


FIG. 3

(57) Abstract: Systems and methods for measuring similarity between a set of clusters generated by a clustering procedure and a set of object labels apply a clustering procedure to a set of objects to cluster the objects into a set of clusters, calculate a similarity index between the set of clusters and object labels determined from a predetermined categorization of objects, calculate an expected value of the similarity index, wherein the expected value of the similarity index is a value of the similarity index one would expect to obtain, on average, between a randomly generated second set of clusters and the object labels, and adjust the calculated similarity index by a penalty factor based at least in part on the expected value of the similarity index.

SYSTEMS AND METHODS FOR CLUSTER VALIDATION

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application is a continuation-in-part of U.S. Patent Application Serial No. 12/906,107, filed on October 17, 2010, which is hereby incorporated by reference in its entirety.

BACKGROUND

Field

[0002] The present disclosure relates to data clustering, and more particularly relates to organizing objects into clusters of related objects, such as organizing a group of images into one or more clusters of related images.

Background

[0003] In the field of digital imaging, a large set of images may be partitioned into smaller clusters of images. For example, a computing device may organize a large collection of images into a two or more clusters of images in which the images in a certain cluster are similar to each other. To that end, a number of different clustering procedures have been developed to automatically cluster images based on features of the images.

[0004] The performance of a clustering procedure may vary depending on the set of images being clustered. Thus, in some situations it may be better to choose one clustering procedure over another.

SUMMARY

[0005] Mutual Information (also referred to herein as “MI”) is an example of an information theoretical measure that has been used to measure similarity between two partitions or clustering results of the same data. The MI measures how closely results of two clustering procedures agree by measuring the mutual dependence of the two clustering results, but it lacks an interpretable baseline, which hinders its usability. There is an adjusted form of MI, referred to as the Adjusted Mutual Information (also referred to herein as “AMI”) that provides MI with an interpretable baseline by subtracting out the expected value of the MI. The value of AMI indicates how well the two partitions or clustering results agree when compared to the agreement obtained by chance.

[0006] AMI is based on a model of randomness where the size of the partitions or clusters is assumed to be known and fixed. However, when comparing the results obtained by clustering algorithms, it may be necessary to account for partitions or clusters that do not have a predetermined size, but instead a size that is determined by the partitioning or clustering algorithm.

[0007] Furthermore, a shortcoming of MI is that MI is a symmetric comparison, and thus only compares one clustering against another. MI can indicate a similarity between two clusterings, but does not account for which (if either) of the clusterings is “true” or correct. For example, some MI procedures result in the same symmetric MI score for either of the two clusterings, implying that the MI procedure simply compares the two clusters to another without regard for either of the clusters being “true”.

[0008] Also, the Rand index is a similarity measure for comparing two clusterings. The Rand index is a pair counting measure that is defined as the fraction of pairs that are either in the same set in both the partitions or are in different sets in both the partitions. However, the Rand index suffers from some of the same shortcomings as MI. In particular, the Rand index is a symmetric comparison, and does not account for a clustering being “true.”

[0009] Thus, while MI and the Rand index can indicate cluster similarity, they do not provide a user with an indication of a specific clustering procedure that would work best for a set of objects, e.g., which clustering procedure would best approximate a manual clustering performed by the user or cluster the objects into “true” clusters.

[0010] In one embodiment, a method for measuring similarity between a set of clusters generated by a clustering procedure and a corresponding set of object labels, wherein the clustering procedure is for grouping the set of objects, comprises applying a clustering procedure to a set of objects to cluster the objects into a first set of clusters, calculating a similarity index between the set of clusters and a set of object labels, calculating an expected value of the similarity index, wherein the expected value of the similarity index is a value of the similarity index one would expect to obtain, on average, between a randomly generated second set of clusters and the object labels, and adjusting the calculated similarity index by a penalty factor based at least in part on the expected value of the similarity index.

[0011] In one embodiment, a system for measuring similarity between a first set of clusters generated by a first clustering procedure and a set of object labels comprises a first computer-readable medium configured to store computer-executable instructions, and one or more processors configured to cause the system to retrieve a set of objects from one or more computer-readable media, apply a first clustering procedure to the set of objects to sort the objects into a first set of clusters, wherein the first set of clusters has a first number of clusters, retrieve a set of object labels from one or more computer-readable media, generate a similarity index between the first set of clusters and the set of object labels, generate an expected value of the similarity index, wherein the expected value of the similarity index indicates an average value of the similarity index between randomly generated sets of clusters and the set of object labels, wherein the randomly generated set of clusters has at most a same number of clusters as the first number of clusters, and adjust the calculated similarity index by a penalty factor that is based at least in part on the expected value of the similarity index.

[0012] In one embodiment, one or more computer-readable media store instructions that, when executed by one or more computing devices, cause the one or more computing devices to perform operations comprising retrieving a set of objects from one or more computer-readable media, generating a first set of clusters based on a first clustering procedure and the set of objects, calculating a similarity index between the first set of clusters and a set of object labels, calculating an expected value of the similarity index, wherein the expected value of the similarity index is a weighted average of possible values of a similarity index between a randomly generated set of clusters and the object labels, and modifying the calculated similarity index by a penalty factor that is based at least in part on the expected value of the similarity index.

[0013] This brief summary has been provided so that the nature of this disclosure may be understood quickly. A more complete understanding can be obtained by reference to the following detailed description and to the attached drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] FIG. 1 is a representative view of computing equipment relevant to one example embodiment.

[0015] FIG. 2 is a detailed block diagram depicting the internal architecture of the host computer shown in FIG. 1.

[0016] FIG. 3 is a representational view of an image processing module according to an example embodiment.

[0017] FIG. 4 is a flow diagram for explaining selection of a clustering procedure according to an example embodiment.

[0018] FIG. 5A is a view depicting a contingency table for explaining clustering according to an example embodiment.

[0019] FIG. 5B is a view for explaining mutual information.

[0020] FIG. 5C is a view for explaining elements of a cluster according to an example embodiment.

[0021] FIGs. 5D to 5F are diagrams depicting example results of clustering procedures according to an example embodiment.

[0022] FIG. 6 is a block diagram that illustrates an example embodiment of objects that are organized into different sets of clusters.

[0023] FIGs. 7A to 7C illustrate example embodiments of contingency tables.

[0024] FIG. 8 is a block diagram that illustrates an example embodiment of objects that are organized into different set of clusters.

[0025] FIGs. 9A and 9B illustrate example embodiments of contingency tables.

[0026] FIG. 10 is a block diagram that illustrates an example embodiment of a method for generating an expected index.

[0027] FIG. 11 illustrates an order of operations used to generate the expected index for two sets of clusters.

[0028] FIG. 12 illustrates example embodiments of contingency tables that show the independence of the column number from a shared index.

[0029] FIG. 13 is a block diagram that illustrates an example embodiment of a method for generating an expected index.

[0030] FIG. 14 is a block diagram that illustrates an example embodiment of a system for validating object clusters.

[0031] FIG. 15A is a block diagram that illustrates an example embodiment of a system for validating object clusters.

[0032] FIG. 15B is a block diagram that illustrates an example embodiment of a system for validating object clusters.

DESCRIPTION

[0033] FIG. 1 is a representative view of computing equipment relevant to one example embodiment. Computing equipment 40 includes host computer 41 which generally comprises a programmable general purpose personal computer (also referred to herein as a “PC”, though computer and PC may refer to any computing device, including, for example, a desktop computer, a server, a PDA, a laptop, a tablet, a phone) having an operating system such as Microsoft® Windows® or Apple® Mac OS® or LINUX, and which is programmed as described below so as to perform particular functions and in effect to become a special purpose computer when performing these functions. Computing equipment 40 includes color monitor 43 including display screen 42, keyboard 46 for entering text data and user commands, and pointing device 47. Pointing device 47 preferably comprises a mouse for pointing and for manipulating objects displayed on display screen 42.

[0034] Host computer 41 also includes computer-readable memory media such as computer hard disk 45 and DVD disk drive 44, which are constructed to store computer-readable information such as computer-executable instructions. DVD disk drive 44 provides a means whereby host computer 41 can access information, such as image data, computer-executable process steps, application programs, etc. stored on removable memory media. Other devices for accessing information stored on removable or remote media may also be provided.

[0035] Host computer 41 may acquire digital image data from other sources such as a digital video camera, a local area network, or the Internet via a network interface. Likewise, host computer 41 may interface with other color output devices, such as color output devices accessible over a network interface.

[0036] Display screen 42 displays a list of clustering procedures and a respective score for each procedure, along with a selection of the clustering procedure with the highest score. In that regard, while the below process will generally be described with respect to images for purposes of conciseness, it should be understood that other embodiments could also operate on other objects. For example, other embodiments could be directed to selecting a clustering procedure for clustering audio files, moving image files, documents, etc.

[0037] While FIG. 1 depicts host computer 41 as a personal computer, computing equipment for practicing aspects of the present disclosure can be implemented in a variety of embodiments, including, for example, a server, a laptop, a digital camera, mobile devices such as cell phones, ultra-mobile computers, netbooks, portable media players, or game consoles, among many others.

[0038] FIG. 2 is a detailed block diagram showing the internal architecture of host computer 41 of computing equipment 40. As shown in FIG. 2, host computer 41 includes central processing unit (CPU) 110 which interfaces with computer bus 114. Also interfacing with computer bus 114 are hard disk 45, network interface 111, random access memory (RAM) 115 for use as a main run-time transient memory, read only memory (ROM) 116, display interface 117 for monitor 43, keyboard interface 112 for keyboard 46, and mouse interface 113 for pointing device 47.

[0039] RAM 115 interfaces with computer bus 114 so as to provide information stored in RAM 115 to CPU 110 during execution of the instructions in software programs such as an operating system, application programs, image processing modules, and device drivers. More specifically, CPU 110 first loads computer-executable process steps from fixed disk 45, or another storage device into a region of RAM 115. CPU 110 can then execute the stored process steps from RAM 115 in order to execute the loaded computer-executable process steps. Data such as color images or other information can be stored in RAM 115, so that the data can be accessed by CPU 110 during the execution of computer-executable software programs, to the extent that such software programs have a need to access and/or modify the data. A module includes computer-readable

instructions that may be executed by one or more members (e.g., a CPU, direct mapped-memory) of one or more computing devices to cause the computing device(s) to perform certain operations, though for purposes of description a module may be described as performing the operations.

[0040] As also shown in FIG. 2, hard disk 45 contains computer-executable process steps for operating system 118, and application programs 119, such as graphic image management programs. Hard disk 45 also contains computer-executable process steps for device drivers for software interface to devices, such as input device drivers 120, output device drivers 121, and other device drivers 122. Clustering procedures 124 include a set of clustering procedures for clustering image files, audio files or other data. Image files 125, including color image files, and other files 126 are available for output to color output devices and for manipulation by application programs.

[0041] Image processing module 123 comprises computer-executable process steps, and generally comprises an input module, an extraction module, a determination module, a comparison module, an application module, a score generation module, and a selection module. Image processing module 123 inputs a set of images, and outputs a selection of a clustering procedure which best fits the set of images. More specifically, image processing module 123 comprises computer-executable process steps executed by a computer for causing the computer to perform a method for selecting a clustering procedure for grouping the set of images, as described more fully below.

[0042] The computer-executable process steps for image processing module 123 may be configured as a part of operating system 118, as part of an output device driver such as a printer driver, or as a stand-alone application program such as a color management system. They may also be configured as a plug-in or dynamic link library (DLL) to the operating system, device driver or application program. For example, image processing module 123 according to example embodiments may be incorporated in an output device driver for execution in a computing device, such as a printer driver, embedded in the firmware of an output device, such as a printer, in an input/output device such as a camera with a display, in a mobile output device (with or without an input camera) such

as a cell-phone or music player, or provided in a stand-alone image management application for use on a general purpose computer. It can be appreciated that the present disclosure is not limited to these embodiments and that the disclosed image processing module 123 may be used in other environments in which image clustering is used.

[0043] FIG. 3 illustrates the image processing module of FIG. 2 according to an example embodiment.

[0044] In particular, FIG. 3 illustrates an example architecture of image processing module 123 in which the sub-modules of image processing module 123 are included in fixed disk 45. Each of the sub-modules are computer-executable software code or process steps executable by a processor, such as CPU 110, and are stored on a computer-readable storage medium, such as fixed disk 45 or RAM 115. More or less modules may be used, and other architectures are possible.

[0045] As shown in FIG. 3, image processing module 123 includes an input module 301 for inputting color images and a predetermined categorization of the images, an extraction module 302 for extracting image features from each image in the set of images, and a determination module 303 for determining a comparison measure by which to compare respective features of the set of images. Comparison module 304 compares respective features between the images in the set of images based on the comparison measure, and outputs a group of measures representing the differences between features of respective images. Application module 305 applies plural clustering procedures to the set of images to cluster the images based in part on the calculated group of measures. Score generation module 306 generates a clustering quality score for each clustering procedure, based on the clusters created by the clustering procedure and the predetermined categorization of images. Selection module 307 selects a clustering procedure with a high clustering quality score. Each of these functions will be described more fully below.

[0046] FIG. 4 is a flow diagram for explaining selection of a clustering procedure for grouping a set of images.

[0047] Briefly, in FIG. 4, a clustering procedure for grouping a set of images is selected from amongst plural clustering procedures. A predetermined categorization of objects such as images is input, and image features are extracted from each image in the set of images. A comparison measure is determined, by which to compare respective features of the set of images. Respective features between the images in the set of images are compared, based on the comparison measure, and a group of measures representing the differences between features of respective images is output. The plural clustering procedures are applied to the set of images to cluster the images based in part on the calculated group of measures. A clustering quality score is generated for each clustering procedure, based on the clusters created by the clustering procedure and the predetermined categorization of images. The clustering procedure with a high clustering quality score is selected.

[0048] In more detail, in step 401, a set of images are input, along with a predetermined categorization of the images. In that regard, the predetermined categorization of images can be selected by a user, generated based on past categorizations of images, or generated using pre-labeled learning set of images, among other methods. For example, multiple user selections of categorizations could be aggregated and stored, or transmitted to computing equipment 40 over a network. Of course, the predetermined categorization could also be adjusted or modified over time, to keep up with changes in categorizations by users.

[0049] In step 402, image features are extracted from all of the input images. For example, colors, shapes, local features like SIFT bag of words, among many others can be extracted. In that regard, in many cases, clustering is not performed on the actual objects, but on features extracted from them. For example, a procedure for clustering images of cars does not usually operate in the pixel space of the images, but instead works with features, such as color or shape, extracted from the images. Thus, objects may be clustered based on features extracted from the objects.

[0050] In step 403, a comparison measure is determined, by which to compare respective features of the set of images. For example, the comparison measure could be a chi-

squared distance, a “histogram intersection” measure, Cosine distance, Tanimoto coefficient, Lp distances, Earth movers distance, or Hamming distance, among many others.

[0051] In step 404, respective features between the images in the set of images are compared based on the comparison measure, and a group of measures representing the differences between features of respective images is output. In particular, each image is compared against every other image in the set, and the output measures indicate how similar (or different) the images are according to the selected comparison measure. As an example, for a collection of images, the feature could be a SIFT bag of words vector, which is generated for every image. Each image feature can be compared to every other feature using a cosine similarity measure which would generate a similarity score for every image pair in our collection of images.

[0052] Other features may be used as well. For example, if the objects are documents, the features may be words or phrases in the documents. Also, other data associated with the objects, including metadata, may be used, for example date of creation, date of modification, author, location of creation, device that created the object, time of day of creation of the object, type of object, etc.

[0053] In step 405, the plural clustering procedures are applied to the set of images to cluster the images, based in part on the calculated group of measures. In that regard, nearly every clustering procedure uses at least one such measure in the clustering process. Thus, each clustering procedure is executed on the set of images, based in part on the calculated set of measures, to generate resultant clusters. In that regard, nearly every clustering procedure uses at least one such measure in the clustering process. Thus, each clustering procedure is executed on the set of images, based in part on the calculated set of measures, to generate resultant clusters. In that regard, it should be understood that each clustering procedure involves a choice of feature, comparison measure, and specific clustering process. Thus, this step contemplates that the same clustering procedure could be used multiple times but with differing parameters, to thereby produce different results. Additionally some clustering procedures involve randomization in their processing steps

and therefore can produce different results even with the same parameters. Thus this step can also contemplate the effectiveness of various versions of the same clustering procedure with the same parameters. For all practical purposes these can be considered as different clustering procedures to be evaluated.

[0054] In step 406, a clustering quality score is generated for each clustering procedure, based on the clusters created by the clustering procedure and the predetermined categorization of images.

[0055] For example, the clustering quality score can be generated for each clustering procedure by calculating a similarity index between the clustering procedure and the predetermined categorization of images, and adjusting by a penalty factor that is based on an expected index (also referred to herein as “EI”) from randomly assigning images to clusters of the clustering procedure. A similarity index is an information theoretical measure that measures similarity between two partitions or clustering results of the same data. Examples of similarity indexes include MI and the Rand index. However, it should be understood that there are many other similarity indexes for measuring similarity of clustering results and that the methods described here can be applied to these other methods as well.

[0056] In one example embodiment the clustering quality score (also referred to herein as “AI” or “AI*”) is generated according to

$$AI^* = \frac{\hat{I}(M) - E[\hat{I}(M)|a, C]}{K},$$

wherein the index generated from a contingency table M, $\hat{I}(M)$, equals the similarity index between the clustering procedure result and the predetermined categorization of images, $E[\hat{I}(M)|a, C]$ equals the expected index and is based on the number of clusters C generated by the clustering procedure, and on the predetermined categorization a, and wherein K is a normalization constant which depends only on the predetermined categorization a and the number of clusters C. Generation of AI* will be described more fully below with respect to FIGs. 5A to 5F.

[0057] In step 407, a clustering procedure with a high clustering quality score is selected.

[0058] In step 408, the selected clustering procedure is output. In that regard, the output clustering procedure may be displayed to the user as in FIG. 1, or may simply be stored or saved at computing equipment 41 for later use in grouping the same images or similar images. The selected clustering procedure could also be output over a network. Of course, many other methods of outputting the selected clustering procedure are possible.

[0059] Generation of the clustering quality score (i.e., AI^*) will now be described more fully with respect to FIGs. 5A to 5F. The following description will use MI by way of example, though other embodiments may use other similarity indexes.

[0060] Turning to FIG. 5A, let $S = \{S_1, S_2, \dots, S_N\}$ be a set of N objects, where each S_i belongs to one of R true categories, labeled $\mathcal{U} = \{U_1, U_2, \dots, U_R\}$. A clustering procedure produces a partition of these N objects into C clusters labeled $\mathcal{V} = \{V_1, V_2, \dots, V_C\}$.

[0061] The overlap between the true categories, and the clusters produced by a clustering procedure, can be summarized in the form of a contingency table M shown in FIG. 5A, where table element M_{ij} is the number of objects belonging to category U_i that was placed into cluster V_j . We also define a_i = row sums, and b_j = column sums. This is shown in FIG. 5A. Formally, the mutual information $I(X; Y)$ between discrete random variables X and Y is defined as

$$I(X; Y) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log \frac{p(x, y)}{p(x)p(y)},$$

where $\mathcal{X}$ and $\mathcal{Y}$ are the domains of X and Y respectively. $I(X; Y)$ is a symmetric measure that quantifies the information X and Y share. Entropy, denoted by $H(X)$, is a measure of uncertainty associated with a random variable, X . Formally,

$$H(X) = - \sum_{x \in X} p(x) \log p(x).$$

It can be verified that $I(X; Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$. Thus, MI is a measure of how much knowing one of the variables reduces uncertainty of the other. $I(X; Y)$ is upper-bounded by both $H(X)$ and $H(Y)$.

[0062] Using a statistical view, random variables $U \in \mathcal{U}$ can be used to represent the category, and $V \in \mathcal{V}$ to represent the cluster, that an object belongs to. Then after observing a contingency table M , the following frequentist estimates are generated:

$$\begin{aligned} \Pr(U = u_i) &= \frac{\sum_{j=1}^C M_{ij}}{N} = \frac{a_i}{N}, \\ \Pr(V = v_j) &= \frac{\sum_{i=1}^R M_{ij}}{N} = \frac{b_j}{N}, \text{ and} \\ \Pr(U = u_i, V = v_j) &= \frac{M_{ij}}{N}. \end{aligned}$$

The mutual information between U and V can be estimated as

$$\hat{I}(M) = \sum_{i=1}^R \sum_{j=1}^C \theta(M_{ij}, a_i, b_j), \text{ where } \theta(n, a, b) = \frac{n}{N} \log \frac{nN}{ab}.$$

[0063] When comparing two partitions, V and V' with C and C' clusters respectively, against a “true” partition U , if $C = C'$, the MI of each of the two partitions to the true partition, $I(U; V)$ and $I(U; V')$, is a fair measure for comparing these clustering procedures.

[0064] However, if $C \neq C'$ this might not be the case. For example, suppose there is a comparison of 3 partitionings, V_1, V_2 , and V_3 of a dataset consisting of two objects from one category and two objects from another. FIG. 5B shows the contingency tables of V_1, V_2 , and V_3 with respect to U , the “true” partition induced by the labels.

[0065] As shown in FIG. 5B, V_1 is the best possible clustering of the dataset whereas V_3 is a completely uninformative “clustering” since it placed each object in a separate “cluster”. But it turns out that the mutual information for all the 3 contingency tables in FIG. 5B are the same ($= H(U)$). Additionally, any random partition of N objects into N categories, although completely uninformative, achieves the highest possible Mutual Information score with respect to the true clustering.

[0066] Accordingly, a more informative measure should include a correction term to account for the similarity index (including the mutual information described above) that would be obtained by chance. That is, in order to evaluate a procedure that partitions the data into C clusters, the evaluation should take into account how much better this procedure does, on average, than a procedure that randomly partitions the same data into C clusters.

[0067] Therefore, an expected index (also referred to herein as “EI” or “EI*”) is calculated below. Then, EI* can be used as a baseline / penalty factor that can be subtracted from a similarity index (e.g., MI, Rand index) to obtain a more meaningful measure to validate a clustering procedure given a “true” clustering. The difference is typically normalized to lie within a range, and the resulting measure can be called the adjusted information, denoted by AI*.

[0068] To calculate the expected index, given N objects, with $a_i > 0$ objects belonging to categories U_i for $i = 1 \dots R$, it would be useful to compute the expectation of the similarity index estimate over all possible clusterings of these objects into exactly C clusters.

[0069] In that regard,

$$EI^* = E[\hat{I}(M)|a, c] = \sum_{M \in \mathcal{M}} \hat{I}(M) P(M|a, C),$$

where M is the set of all $R \times C$ contingency tables M , such that row i sums to a_i for $i = 1 \dots R$, and such that columns sums are non-zero, $P(M|a, C)$ is calculated as

$$P(M|a, C) = \frac{\mathcal{N}(M)}{\sum_{M \in \mathcal{M}} \mathcal{N}(M)},$$

where $\mathcal{N}(M)$ is the number of ways to cluster the given objects that result in the contingency table M that have the given row sums and non-zero column sums. Plugging in the above,

$$\begin{aligned} E[\hat{I}(M)|a, C] &= \sum_{M \in \mathcal{M}} \left\{ \left[\sum_{i=1}^R \sum_{j=1}^C \Theta(M_{ij}, a_i, b_j) \right] \left[\frac{\mathcal{N}(M)}{\sum_{M \in \mathcal{M}} \mathcal{N}(M)} \right] \right\} \\ &= \frac{1}{\sum_{M \in \mathcal{M}} \mathcal{N}(M)} \sum_{i=1}^R \sum_{j=1}^C \sum_{M \in \mathcal{M}} [\Theta(M_{ij}, a_i, b_j) \mathcal{N}(M)] \end{aligned}$$

[0070] The summation over M above can be replaced with a summation over all possible values for b_j and M_{ij} . There is not necessarily a need to sum over a_i since it is a fixed quantity.

[0071] Considering the range of values that b_j and M_{ij} can take, since there must be at least one element in each column of M , b_j has to be at least 1 and at most $N - (C - 1)$. Given b_j , M_{ij} can be at most $\min(a_i, b_j)$. Additionally, after filling the $[i, j]$ th cell, the j th column must be filled with $b_j - M_{ij}$ elements from a pool of $N - a_i$ elements. Therefore, M_{ij} should be at least $(a_i + b_j - N)^+$, which is $\max(0, a_i + b_j - N)$.

[0072] To replace the summation over M as mentioned above, $\mathcal{N}(M)$ should be replaced with $\mathcal{N}(M_{ij}, a_i, b_j|C)$, where $\mathcal{N}(n, a, b|C)$ is the number of ways to cluster the given objects into exactly C clusters such that there are n elements in a particular cell, and the number of elements in the corresponding row and column are a and b respectively. With

this transformation,

$$E[\hat{I}(M)|a, C] = \frac{1}{\sum_{M \in \mathcal{M}} \mathcal{N}(M)} \sum_{i=1}^R \sum_{j=1}^C \sum_{b=1}^{N-C+1} \sum_{n=\left(a_i+b-N\right)^+}^{\min(a,b)} \left[\Theta(n, a_i, b) \mathcal{N}(n, a_i, b | C) \right].$$

[0073] Since the categories of the objects are given, the denominator in the above equations is simply the number of ways to partition N distinguishable objects into C distinguishable non-empty bins, i.e.:

$$\sum_{M \in \mathcal{M}} \mathcal{N}(M) = S(N, C) \times C!,$$

where S denotes a Stirling number of the second kind.

[0074] Turning to FIG. 5C, $\mathcal{N}(n, a, b | C)$ can be calculated. As mentioned, this is the number of ways to cluster the given N objects into exactly C clusters so that a given cell contains n elements and there are a and b elements in its corresponding row and column respectively. Specifically,

$$\mathcal{N}(n, a, b | C) = \binom{a}{n} \binom{N-a}{b-n} S(N-b, C-1) (C-1)!.$$

In addition, substituting into the above, the terms inside the summation are independent of j and hence the summation over j can be removed and the whole expression multiplied by C . Thus,

$$E[\hat{I}(M)|a, C] = \sum_{i=1}^R \sum_{b=1}^{N-C+1} \sum_{n=\max(1, a_i+b-N)}^{\min(a_i, b)} \theta(n, a_i, b) \Psi(a_i, b, n | C)$$

where

$$\Psi(a, b, n | C) = \frac{\binom{a}{n} \binom{N-a}{b-n} S(N-b, C-1)}{S(N, C)}.$$

[0075] Once EI^* has been calculated, the adjusted index can be calculated as

$$AI^* = \frac{\hat{I}(M) - E[\hat{I}(M)|a, C]}{\mathcal{K}},$$

where $\mathcal{K}$ is a normalization constant which depends only on the predetermined categorization of images. Using one such choice for $\mathcal{K}$ we have

$$AI^* = \frac{\hat{I}(M) - E[\hat{I}(M)|a, C]}{H(U) - E[\hat{I}(M)|a, R]},$$

where the denominator $H(U) - E[\hat{I}(M)|a, R]$ is the information content of the predetermined categorization, a , minus the expected index we get given R clusters and the predetermined categorization a . In another embodiment, the adjusted index can be normalized using a divisor that may depend on the clustering result. For example,

$$AI^* = \frac{\hat{I}(M) - E[\hat{I}(M)|a, C]}{\mathcal{K} - E[\hat{I}(M)|a, C]}$$

where, for example, $\mathcal{K}$ could be $H(U)$.

[0076] FIGs. 5D to 5F are diagrams depicting example results of clustering procedures according to an example embodiment. Specifically, FIG. 5D illustrates AI^* based on different input features and clusters produced. Meanwhile, FIG. 5E illustrates AI^* based on different distance comparisons and clusters produced. FIG. 5F illustrates AI^* based on example clustering procedures and clusters produced.

[0077] By comparing a set of clustering procedures against a fixed “true” categorization of images to generate respective scores for each clustering procedure, it is ordinarily possible to automatically choose a clustering procedure which will group images in a manner best approximating a grouping performed manually by a user.

[0078] As mentioned above, while the above process has been described with respect to images for purposes of conciseness, it should be understood that other embodiments

could also operate on other objects. For example, other embodiments could be directed to selecting a clustering procedure for clustering audio files, moving image files, documents, etc.

[0079] FIG. 6 is a block diagram that illustrates an example embodiment of objects that are organized into different sets of clusters. The objects in a collection 600 are organized by a first clustering procedure 610 into clusters 611, 612, 613, 614, and 615. The objects in the collection 600 are also organized by object labels 620 into clusters 621, 622, and 623. The clusters 621–623 formed by applying the object labels 620 may be assumed to indicate a ground “truth” about the objects. For example, in the embodiment shown in FIG. 6, the clusters 621–623 indicate a shape of the objects in the respective clusters (triangle, square, circle). Thus, clusters 611–615 may be compared with clusters 621–623 to determine how accurately the first clustering procedure 610 assigns objects to respective clusters.

[0080] The number of objects and the number of clusters are shown as an example. In other embodiments the collection 600 may include more objects. Also, though the first clustering procedure 610 assigned the objects to five clusters, in other embodiments the objects may be assigned to more or less clusters (e.g., 2, 7, 9, 25, 34, X (where X is an arbitrary number of clusters)). The number of clusters may be constrained to be less than or equal to the total number of objects in the collection 600.

[0081] FIGs. 7A to 7C illustrate example embodiments of contingency tables. FIG. 7A shows an example embodiment of a contingency table 700A that has three rows and five columns and that compares clusters 611–615 with clusters 621–623. Thus, the table 700A includes three rows u_1 – u_3 , corresponding to the three clusters of clusters 621–623, and includes five columns v_1 – v_5 , corresponding to the five clusters of clusters 611–615. The total number of objects, 33, equals the total number of objects in the collection 600. The row marginal totals 730 indicate the total number of objects in the respective clusters 621–623 associated with the rows u_1 – u_3 . The column marginal totals b_1 – b_5 indicate the total number of objects in the respective clusters 611–615 associated with the columns

v_1-v_5 . In table 700A, the column marginal totals v_1-v_5 and the cell totals $M_{11}-M_{35}$ have not been determined.

[0082] FIG. 7B shows an example embodiment of a contingency table 700B that has three rows 710B and five columns 740B and that compares clusters 611–615 with clusters 621–623. The table 700B shown in FIG. 7B has values that correspond to clusters 611–615 and clusters 621–623 (the clusters that are generated by applying the object labels and that indicate the ground “truth” of the objects). Thus, the rows marginal totals 730B indicate the total number of objects in the cluster associated with the respective row 710B. Since cluster 623 contains 12 circles, and the marginal total 730B of the row 710B associated with cluster 623 (indicated by a circle) is 12. Likewise, cluster 621 contains 11 triangles, and the row 710B associated with cluster 621 (indicated by a triangle) has a marginal total 730B of 11. Also, cluster 622 contains 10 squares, and the row 710B associated with cluster 622 (indicated by a square) has a marginal total 730B of 10.

[0083] Furthermore, the column marginal totals 740B indicate the total number of objects in the respective cluster associated with the column. Thus, the marginal total of column v_1 is 3 because 3 is the total number of objects in cluster 611, the marginal total of column v_2 is 7 because 7 is the total number of objects in cluster 612, the marginal total of column v_3 is 10 because 10 is the total number of objects in cluster 613, the marginal total of column v_4 is 9 because 9 is the total number of objects in cluster 614, and the marginal total of column v_5 is 4 because 4 is the total number of objects in cluster 615.

[0084] The contingency table 700B indicates the sharing of objects between clusters 611–615 and clusters 621–623. For example, cell 751B indicates that there are 2 objects shared by cluster 612 and cluster 621. Also for example, cell 753B indicates that there are 5 objects shared by cluster 614 and cluster 623.

[0085] FIG. 7C shows an example embodiment of a contingency table 700C that has three rows and five columns and that indicates objects shared between five randomly generated clusters based on the objects in the collection 600 and clusters 621–623 (the

clusters that are associated with the object labels and that indicate the ground “truth” about the objects). The columns v_1-v_5 are associated with the randomly generated clusters. Randomly generated sets of clusters may be formed by assigning objects in a collection to clusters without performing any analysis (e.g., feature extraction) on the objects. Though the clusters associated with the columns v_1-v_5 are randomly generated, the randomly generated clusters still share objects with clusters 621–623. For example, cell 751C indicates that the randomly generated cluster associated with column v_5 shares 2 objects with cluster 621. Therefore, even clusters generated by a clustering procedure that randomly assigns objects to clusters will still have some similarity to clusters generated based on object labels, and hence, the ground “truth” of the objects.

[0086] FIG. 8 is a block diagram that illustrates an example embodiment of objects that are organized into different sets of clusters. The objects in a collection 800 are organized by a second clustering procedure 820 into clusters 821, 822, and 823. The objects in the collection 800 are also organized based on object labels 810 into clusters 811, 812, 813, and 814. The clusters 811–814 formed based on the object labels 810 may be assumed to indicate a ground “truth” about the objects. For example, in the embodiment shown in FIG. 8, the clusters 811–814 indicate a shape of the objects in the respective clusters (triangle, square, circle, star). Thus, clusters 811–814 may be compared with clusters 821–823 to determine how accurately the second clustering procedure 820 assigns objects to respective clusters relative to the ground “truth.”

[0087] FIGs. 9A and 9B illustrate example embodiments of contingency tables. FIG. 9A shows an example embodiment of a contingency table 900A that has four rows 910A and three columns v_1-v_5 and that compares clusters 811–814 with clusters 821–823. The table 900A shown in FIG. 9A has values that correspond to clusters 811–814 and clusters 821–823 (the clusters that are generated by applying the object labels and that indicate the ground “truth” of the objects). Thus, the rows marginal totals 930A indicate the total number of objects in the cluster that is associated with the respective row. Since cluster 811 contains 10 triangles, and the marginal total of the row associated with cluster 811 (indicated by a triangle) is 10. Likewise, cluster 812 contains 10 circles, and the row

associated with cluster 812 (indicated by a circle) has a marginal total of 10. Also, cluster 813 contains 8 squares, and the row associated with cluster 813 (indicated by a square) has a marginal total of 8. Finally, cluster 814 contains 7 stars, and the row associated with cluster 814 (indicated by a star) has a marginal total of 7.

[0088] Furthermore, the column marginal totals 940A indicate the total number of objects in the respective cluster that is associated with the column. Thus, the marginal total of column v_1 is 15 because 15 is the total number of objects in cluster 821, the marginal total of column v_2 is 8 because 8 is the total number of objects in cluster 822, and the marginal total of column v_3 is 12 because 12 is the total number of objects in cluster 823.

[0089] The contingency table 900A indicates objects shared between clusters 811–814 and clusters 821–823. For example, cell 951A indicates that there are 2 objects shared by cluster 811 and cluster 822. Also for example, cell 953A indicates that there are 3 objects shared by cluster 814 and cluster 821.

[0090] FIG. 9B shows an example embodiment of a contingency table 900B that has four rows and three columns and that indicates objects shared between three randomly generated clusters based on the objects in the collection 800 and clusters 811–814 (the clusters that are associated with the object labels and that indicate the ground “truth” about the objects). The columns v_1 – v_3 are associated with the randomly generated clusters. Though the clusters associated with the columns v_1 – v_3 are randomly generated, the randomly generated clusters still share objects with clusters 811–814. For example, cell 951A indicates that the randomly generated cluster associated with column v_2 has 6 objects in common with cluster 813. Like FIG. 7C, this illustrates that even clusters generated by a clustering procedure that randomly assigns objects to clusters will still have some similarity to clusters generated based on object labels, and hence, the ground “truth” of the objects. Thus, merely determining the objects shared by the clusters created by applying a clustering procedure to a collection of objects and the clusters created based on object labels may not provide an accurate measure of the effectiveness of the clustering procedure, because the object sharing may be based on chance rather

than an inherent ability of the clustering procedure to discover the object labels. Rather, a baseline for the comparison is desirable in order to evaluate the performance of a clustering procedure's partitioning of objects into clusters relative to a random partitioning of the objects to clusters. A baseline may be used as a point of reference to evaluate a clustering procedure.

[0091] FIG. 10 is a block diagram that illustrates an example embodiment of a method for generating an expected index. An expected index (also referred to herein as "EI") may be used as a baseline when evaluating the performance of a clustering procedure. Before the flow in FIG. 10 begins, the number of clusters generated by the object labels is determined (this value is referred to as "R"). Also, the number of clusters generated by the clustering procedure that is to be evaluated is determined (this value is referred to as "C"). Though the clusters generated using the object labels are associated with respective rows and the clusters generated using the clustering procedure are associated with respective columns in some embodiments, in other embodiments the clusters generated using the object labels are associated with the columns and the clusters generated using the clustering procedure are associated with the rows. Additionally, other embodiments of this method and the other methods described herein may omit blocks, add blocks, change the order of the blocks, combine blocks, and/or divide blocks into separate blocks. Additionally, one or more components of the systems and devices described herein may implement the method shown in FIG. 10 and the other methods described herein. Finally, though it may be helpful to reference a contingency table to understand the method shown in FIG. 10, generation of a contingency table is not required to perform the method. Thus, for example, a computing device performing the method would not need (though it may) to devote resources (e.g., processor time, memory) to generating and storing a contingency table.

[0092] Flow starts in block 1000, where the EI is set to 0 and a counter i is set to 1. The counter i may indicate a cluster (which may be illustrated by an associated row) formed by applying object labels to a collection of objects. Flow then proceeds to block 1005, where it is determined if all values of i have been considered. All values of i may be

found to not have been considered if $i \leq R$ or may be found to have been considered if $i > R$. If in block 1005 it is determined that all values of i have been considered, then flow proceeds to block 1020, where the EI is returned. However, if $R \geq 1$, then all values of i will not have been considered the first time flow proceeds to block 1005, and thus if at least one cluster is generated using the object labels then the result of the determination the first time flow proceeds to block 1005 will be “no.” If in block 1005 it is determined that all values of i have not been considered, then flow proceeds to block 1010.

[0093] In block 1010, the counter j , which may correspond to a cluster (which may be illustrated by an associated column) formed by a clustering procedure, is reset. For example, j may be set to 1. Flow then proceeds to block 1015, where the value of the next a_i is set to the number of objects in one of the clusters formed using the object labels. The i count indicates the particular cluster, and, thus as i increases, the number of objects in each of the clusters formed using the object labels will be considered. If a reference to a contingency table is made, the i count refers to a row of the table, and a_i is the marginal value of the row (which indicates the number of objects in the corresponding cluster). Therefore, using the example of the clusters shown in FIG. 6 that were formed using the object labels, as i increases during iterations of the flow, i will assume the values of 1, 2, and 3, and when $i = 1$ then a_1 is set to 11, when $i = 2$ then a_2 is set to 10, and when $i = 3$ then a_3 is set to 12.

[0094] From block 1015, flow then proceeds to block 1025, where it is determined if all values of j have been considered (e.g., determine if all C clusters have been considered). All values of j may be found to not have been considered if $j \leq C$ or may be found to have been considered if $j > C$. If all values of j have been considered, then flow proceeds back to block 1005. If all values of j have not been considered, then flow proceeds to block 1030. In block 1030, the counter b , which may correspond to a number of objects in a cluster (which may be illustrated by the marginal value of the associated column), is reset. For example, b may be set to 1, though in embodiments where clusters are not assumed to be non-empty b may be set to 0.

[0095] Flow then proceeds to block 1035, where it is determined if all values of b have been considered. In some embodiments, it is assumed that no clusters are empty. Therefore, if C clusters are generated by a clustering procedure (illustrated by C columns in a contingency table), then each of the C clusters must have at least one object. Therefore, the maximum number of objects that can be in any single cluster is $N - C + 1$, and the minimum number of objects in any single cluster is 1. Accordingly, all values of b may be determined to have not been considered if $b \leq N - C + 1$, or may be determined to have been considered if $b > N - C + 1$. If all values of b have been considered, then flow proceeds to block 1025. If all values of b have not been considered, then flow proceeds to block 1040. In embodiments that allow empty clusters, in block 1030 b would be set to 0 and in block 1035 all values of b up to and including N would be considered.

[0096] In block 1040, n , a counter associated with a number of objects, is reset, and the shared weighted total (also referred to herein as “SWT”), which indicates the aggregate weighted similarity index, is reset. SWT may be set to 0. For n , the minimum number of shared objects is the maximum of 1 and $a_i + b - N$, assuming that all R clusters and all C clusters are non-empty. Thus, n may be set to the maximum of 1 and $a_i + b - N$. Flow then proceeds to block 1045. Note that in some embodiments n could be set to the maximum of 0 and $a_i + b - N$ (which may be indicated by the notation “ $(a_i + b - N)^+$ ”). Though in the embodiment shown in FIG. 10 the $n = 0$ case may not have any contributive effect to the SWT, in some embodiments the $n = 0$ case contributes to the SWT, and therefore n may be set to 0 or the maximum of 0 and $a_i + b - N$.

[0097] In block 1045, it is determined if all values of n given a_i and b have been considered. The value of n is related to a_i and b because the number of objects shared by two clusters cannot exceed the total number of objects in either of the clusters. Thus, the maximum value of n is the minimum of a_i and b . Therefore, if $n > \min(a_i, b)$ all values of n may be determined to have been considered. Or, if $n \leq \min(a_i, b)$, all values of n may be determined to have not been considered. If all values of n have been considered, then

flow proceeds to block 1070 (discussed in para. [0102]). If all values of n have not been considered, then flow proceeds to block 1050.

[0098] In block 1050, the shared index for the given a_i , b , and n is calculated. θ may be used to denote a function used to calculate the shared index, thus the shared index may be denoted by $\theta(n, a_i, b)$. Also, in some embodiments, which calculate the shared index based on Mutual Information,

$$\theta(n, a_i, b) = \frac{n}{N} \log \frac{nN}{a_i b}.$$

Additionally, in some embodiments, the shared index may be calculated based on the Rand index:

$$\theta(n, a_i, b) = \frac{\frac{N^2}{RC} - \frac{a_i^2}{C} - \frac{b^2}{R} + 2n^2}{N(N-1)}.$$

After the shared index is calculated, flow proceeds to block 1055.

[0099] In block 1055, the probability of n given a_i , b , and N is calculated. This may be denoted by $P_n(n | a_i, b, N)$. In some embodiments, P_n is given by the density of the hypergeometric distribution. Therefore, in these embodiments,

$$P_n(n | a_i, b, N) = \frac{\binom{a_i}{n} \binom{N - a_i}{b - n}}{\binom{N}{b}}.$$

Thus, P_n may indicate the probability of sharing n objects when a first cluster has a_i objects (illustrated by the marginal value of an associated row), when a second cluster has b objects (illustrated by the marginal value of an associated column), and the total number of objects in the collection is N .

[0100] Flow then proceeds to block 1060, where the weighted value of the shared index is calculated. The probability of n given a_i , b , and N may be used to weigh the shared

index θ . Therefore, in some embodiments the weighted value of the shared index is $\theta(n, a_i, b)P_n(n|a_i, b, N)$. Flow then moves to block 1065.

[0101] In block 1065, the weighted value of the shared index is added to the SWT. Thus, in a particular iteration of the flow, $SWT += \theta(n, a_i, b)P_n(n|a_i, b, N)$. After block 1065, n is incremented (e.g., $n = n + 1$) and flow then proceeds back to block 1045. Therefore, after all values of n are considered in block 1045, $SWT = \sum_n \theta(n, a_i, b)P_n(n|a_i, b, N)$.

[0102] If in block 1045 all values of n given a_i and b are found to have been considered, then flow proceeds to block 1070. In block 1070, the marginal weighted value (also referred to herein as “MWV”) is generated based on the SWT and the probability of b , which may be denoted by $P_b(b)$. The marginal weighted value may be generated according to $MWV = SWT * P_b(b)$. Depending on the embodiment, $P_b(b)$ may be generated based on b given N and C and may be denoted by $P_b(b; N, C)$. In some embodiments,

$$P_b(b; N, C) = \frac{1}{C} \binom{N}{b} \frac{\{N-b\}}{\{C-1\}},$$

where $\{N\}_C$ is a Stirling number of the second kind, sometimes denoted at $S(N, C)$.

[0103] The flow then proceeds to block 1075, where the MWV is added to the EI. Thus, for any given iteration of the flow through block 1075, $EI += MWV$. Next, b is incremented (e.g., $b = b + 1$) and flow returns to block 1035. If all values of b are found to have been considered in block 1035, then j is incremented (e.g., $j = j + 1$) and flow returns to block 1025. If all values of j are found to have been considered in block 1025, then i is incremented (e.g., $i = i + 1$) and flow returns to block 1005. If in block 1005 all values of i are found to have been considered, then flow proceeds to block 1020, where the EI is returned. The EI generated by the flow after all values of i, j, b , and n have been considered may be given by the following equation:

$$EI = \sum_{i=1}^R \sum_{j=1}^C \left[\sum_{b=1}^{N-C+1} \left\{ \sum_{n=\max(1, a_i+b-N)}^{\min(a_i, b)} \theta(n, a_i, b) P_n(n|a_i, b, N) \right\} P_b(b; N, C) \right].$$

In some embodiments, which are based on mutual information,

$$EI = \sum_{i=1}^R \sum_{j=1}^C \left[\sum_{b=1}^{N-C+1} \frac{1}{C} \binom{N}{b} \frac{\{N-b\}}{\{C-1\}} \times \left\{ \sum_{n=(a_i+b-N)^+}^{\min(a_i, b)} \frac{n}{N} \log \frac{nN}{a_i b} \frac{\binom{a_i}{n} \binom{N-a_i}{b-n}}{\binom{N}{b}} \right\} \right].$$

Also, in some embodiments, which are based on the Rand index,

$$EI = \left(1 - \frac{\sum_{i=1}^R (a_i)_2}{(N)_2} \right) \left(1 - \frac{\{N-1\}}{\{C\}} \right) + \frac{\sum_{i=1}^R (a_i)_2}{(N)_2} \frac{\{N-1\}}{\{C\}}.$$

[0104] FIG. 11 illustrates a flow used to generate the expected index for two sets of clusters. FIG. 11 illustrates this flow by referencing a contingency table. First, as is shown in table 1100, flow proceeds row by row. A specific row being evaluated is denoted by i , and a_i denotes the row marginal value of row i (a_i may indicate the number of objects in a cluster generated using the object labels, which is assumed to be a ground “truth”). Next, within each row and for a given i (and thus a_i), as is shown in table 1110, flow proceeds column by column across all columns. A specific column being evaluated is denoted by j . In other embodiments, flow may first proceed column by column then row by row.

[0105] Within column j , as is shown in table 1120, all column marginal values, denoted by b_j , are considered. Finally, for each column marginal value b_j , all values of n_{ij} are considered, as is shown in table 1130. For all cell values of n_{ij} a weighted value of the shared index is calculated. Thus, the flow systematically calculates the weighted value of the shared index for all values of a cell given all marginal values of a column, for all marginal values of a column given a row (which indicates an associated row marginal value), and for all rows of a table.

[0106] FIG. 12 illustrates example embodiments of contingency tables that show the independence of the column number from a shared index. In one embodiment, the shared index θ for n objects is

$$\theta(n, a_i, b) = \frac{n}{N} \log \frac{nN}{a_i b}.$$

However, the shared index may be generated in other ways, for example by

$$\theta(n, a_i, b) = \frac{\frac{N^2}{RC} - \frac{a_i^2}{C} - \frac{b^2}{R} + 2n^2}{N(N-1)}.$$

[0107] Therefore, the shared index depends on the number of shared objects n , the row marginal value 1215, and the column marginal value 1217. However, the shared index given a particular value of b is not dependent on the column number j . Therefore, the shared index of cell 1213 is the same as the shared index of cell 1223, even though cell 1213 is in column v_2 and cell 1223 is in column v_3 , because the row marginal value 1225 and the column marginal value 1227 of cell 1223 are the same as the row marginal value 1215 and the column marginal value 1217 of cell 1213, respectively. Likewise, though cell 1233 is in column v_1 and cell 1243 is in column v_C , both cells have the same shared index as cell 1213 and cell 1223 because the row marginal values 1235, 1245 and the column marginal values 1237, 1247 are the same as row marginal values 1215, 1225 and column marginal values 1217, 1227, respectively.

[0108] Not only is the shared index independent of the column number, but columns v_1 – v_C have the same range of marginal values b , since b lies within the range of 1 and $N - C + 1$ for each column (or in the range of 0 to N , in some embodiments). Thus, since the shared index does not depend on column number j , and since the range of column marginal values b given N and C is the same for all columns, calculating the shared index for all the cells in a row (the number of cells in a row is C) for all values of b given N and C duplicates the calculation C times. Thus, for a given n , a_i and b the shared index needs to be calculated only once.

[0109] However, the probability of a column marginal value b having a given value is relevant to the weighted value of the shared index. But the probability of b given N and C (denoted by $P_b(b; N, C)$) does not depend on column number and, hence, is the same for each column. Therefore, when calculating the probability of a given value of b , it is only necessary to calculate the probability that any column marginal value has the given value of b . Thus, it is not necessary to calculate a probability of b for each value of b in each column (e.g., calculate $P_b(b_j; N, C)$ for j from 1 to C) if the probability that any column has the marginal value of b given N and C is known. In some embodiments, the probability that any one column has the marginal value of b given N and C is given by the following distribution (also referred to herein as the “Stirling distribution” since it includes Stirling numbers of the second kind or as the “cluster cardinality distribution”):

$$P_b(b; N, C) = \frac{1}{C} \binom{N}{b} \frac{\{N-b\}}{\{C-1\}}.$$

Thus, when the distribution P_b is summed for all columns C we are given the sum

$$S_b(b) = \sum_{j=1}^C \frac{1}{C} \binom{N}{b} \frac{\{N-b\}}{\{C-1\}} = \binom{N}{b} \frac{\{N-b\}}{\{C-1\}}.$$

Therefore, since the shared index, P_n , and S_b do not depend on j , the above sum of P_b can be substituted for $\sum_{j=1}^C P_b(b; N, C)$ and the summation over j can be removed from the equation

$$EI = \sum_{i=1}^R \sum_{j=1}^C \left[\sum_{b=1}^{N-C+1} \left\{ \sum_{n=\max(1, a_i+b-N)}^{\min(a_i, b)} \theta(n, a_i, b) P_n(n|a_i, b, N) \right\} P_b(b; N, C) \right],$$

which yields

$$EI = \sum_{i=1}^R \sum_{b=1}^{N-C+1} \left\{ \sum_{n=\max(1, a_i+b-N)}^{\min(a_i, b)} \theta(n, a_i, b) P_n(n|a_i, b, N) \right\} \binom{N}{b} \frac{\{N-b\}}{\{C-1\}}.$$

However, in embodiments where

$$P_n(n | a_i, b, N) = \frac{\binom{a_i}{n} \binom{N - a_i}{b - n}}{\binom{N}{b}},$$

then

$$EI = \sum_{i=1}^R \sum_{b=1}^{N-C+1} \left\{ \sum_{n=\max(1, a_i+b-N)}^{\min(a_i, b)} \theta(n, a_i, b) \frac{\binom{a_i}{n} \binom{N - a_i}{b - n}}{\binom{N}{b}} \right\} \binom{N}{b} \frac{\{N - b\}}{\{C\}}.$$

This may be simplified to

$$EI = \sum_{i=1}^R \sum_{b=1}^{N-C+1} \sum_{n=\max(1, a_i+b-N)}^{\min(a_i, b)} \theta(n, a_i, b) \frac{\binom{a_i}{n} \binom{N - a_i}{b - n} \{N - b\}}{\{C\}}.$$

Therefore, this simplification may be used to calculate the EI.

[0110] FIG. 13 is a block diagram that illustrates an example embodiment of a method for generating an expected index. Before the flow in FIG. 13 begins, the number of clusters generated using the object labels is determined (this value is referred to as “R”). Also, the number of clustered generated by the clustering procedure that is to be evaluated is determined (this value is referred to as “C”). Though the clusters generated using the object labels are associated with rows and the clusters generated using the clustering procedure are associated with columns in some embodiments, in other embodiments the clusters generated using the object labels are associated with columns and the clusters generated using the clustering procedure are associated with rows. Additionally, other embodiments of this method may omit blocks, add blocks, change the order of the blocks, combine blocks, and/or divide blocks into separate blocks. Additionally, one or more components of the systems and devices described herein may implement the method shown in FIG. 13. Finally, though it may be helpful to reference a contingency table to understand the method shown in FIG. 13, generation of a contingency table is not required to perform the method. Thus, for example, a computing

device performing the method would not need (though it may) to devote resources (e.g., processor time, memory) to generating and storing a contingency table.

[0111] Flow starts in block 1300, where the EI is set to 0 and a counter i is set to 1. The counter i may indicate a cluster (which may be illustrated by an associated row) formed by applying object labels to a collection of objects. Flow then proceeds to block 1305, where it is determined if all values of i have been considered. All values of i may be found to not have been considered if $i \leq R$ or may be found to have been considered if $i > R$. If in block 1305 is determined that all values of i have been considered, then flow proceeds to block 1310, where the EI is returned. However, if $R \geq 1$, then all values of i will not have been considered the first time flow proceeds to block 1305, and thus if at least one cluster is generated using the object labels then the result of the determination the first time flow proceeds to block 1305 will be “no.” If in block 1305 is determined that all values of i have not been considered, then flow proceeds to block 1315.

[0112] In block 1315, the counter b , which may correspond to a number of objects in a cluster (which may be illustrated by the associated column marginal value), is reset. For example, b may be set to 1. However, in some embodiments b may be 0.

[0113] Flow then proceeds to block 1320, where it is determined if all values of b have been considered. In some embodiments, it is assumed that no clusters are empty. Therefore, if C clusters are generated by a clustering procedure (illustrated by C columns in a contingency table), then each of the C clusters must have at least one object. Therefore, the maximum number of objects that can be in any single cluster is $N - C + 1$, and the minimum number of objects in any single cluster is 1. Accordingly, all values of b may be determined to have not been considered if $b \leq N - C + 1$, or may be determined to have been considered if $b > N - C + 1$. If all values of b have been considered, then flow proceeds to block 1305. If all values of b have not been considered, then flow proceeds to block 1325. In embodiments that allow empty clusters, in block 1315 b would be set to 0 and in block 1320 all values of b up to and including N would be considered.

[0114] In block 1325, n , a counter associated with a number of objects, is reset, and the shared weighted total (also referred to herein as “SWT”), which indicates the aggregate weighted shared index, is reset. SWT may be set to 0. For n , the minimum number of objects in a cell is the maximum of 1 and $a_i + b - N$, assuming that all R clusters and all C clusters are non-empty. Thus, n may be set to the maximum of 1 and $a_i + b - N$. Flow then proceeds to block 1330. Note that in some embodiments n could be set to the maximum of 0 and $a_i + b - N$. In the embodiment shown in Fig. 13, the $n = 0$ case may not have any contributive effect to the SWT, and thus it can be effectively omitted, though in other embodiments the $n = 0$ case may have a contributive effect.

[0115] In block 1330, it is determined if all values of n given a_i and b have been considered. The value of n is related to a_i and b because the number of objects shared by two clusters cannot exceed the total number of objects in either of the clusters. Thus, the maximum value of n is the minimum of a_i and b . Therefore, if $n > \min(a_i, b)$ all values of n may be determined to have been considered. Or, if $n \leq \min(a_i, b)$, all values of n may be determined to have not been considered. If all values of n have been considered, then flow proceeds to block 1355 (discussed in para. [0120]). If all values of n have not been considered, then flow proceeds to block 1335.

[0116] In block 1335, the shared index given a_i , b , and n is calculated, denoted by $\theta(n, a_i, b)$. After the shared index is calculated, flow proceeds to block 1340.

[0117] In block 1340, the probability of n given a_i , b , and N is calculated. As discussed above, this may be denoted by $P_n(n | a_i, b, N)$.

[0118] Flow then proceeds to block 1345, where the weighted value of the shared index is calculated. The probability of n given a_i , b , and N may be used to weigh the shared index θ . Therefore, in some embodiments the weighted value of the shared index is $\theta(n, a_i, b)P_n(n | a_i, b, N)$. Flow then moves to block 1350.

[0119] In block 1350, the weighted value of the shared index is added to the SWT. Thus, in a particular iteration of the flow, $SWT += \theta(n, a_i, b)P_n(n | a_i, b, N)$. After block 1350,

n is incremented (e.g., $n = n + 1$) and flow then proceeds back to block 1330. Therefore, after all values of n are considered in block 1330, $SWT = \sum_n \theta(n, a_i, b) P_n(n | a_i, b, N)$.

[0120] If in block 1330 all values of n given a_i and b are found to have been considered, flow proceeds to block 1355. In block 1355, the MWV is generated based on the SWT and the sum over all columns of the probability of a column marginal value of b , which may be denoted by $S_b(b)$. The marginal weighted value may be generated according to $MWV = SWT * S_b(b)$. In some embodiments,

$$S_b(b) = \binom{N}{b} \frac{\{N - b\}}{\{C - 1\}}.$$

Because P_n will eventually be multiplied by S_b , in block 1340 P_n may be calculated according to

$$P_n^*(n | a_i, b, N) = \binom{a_i}{n} \binom{N - a_i}{b - n},$$

and in block 1355 P_b may be calculated according to

$$P_b^*(b) = \frac{\{N - b\}}{\{C - 1\}}.$$

[0121] The flow then proceeds to block 1360, where the MWV is added to the EI. Thus, for any given iteration of the flow through block 1360, $EI += MWV$. Next, b is incremented (e.g., $b = b + 1$) and flow returns to block 1320. If all values of b are found to have been considered in block 1320, then i is incremented (e.g., $i = i + 1$) and flow returns to block 1305. If in block 1305 all values of i are found to have been considered, then flow proceeds to block 1310, where the EI is returned. The EI generated by the flow after all values of i , b , and n have been considered may be given by the following equation:

$$EI = \sum_{i=1}^R \sum_{b=1}^{N-C+1} \sum_{n=\max(1, a_i+b-N)}^{\min(a_i, b)} \theta(n, a_i, b) \frac{\binom{a_i}{n} \binom{N-a_i}{b-n} \{C-1\}}{\{N\}}.$$

[0122] However, calculating a Stirling number of the second kind can be very computationally expensive, but approximations may be used which avoid the calculation of Stirling numbers. For example, when n is large, the following approximation may be used:

$$\frac{\{n\}}{\{k\}} \approx \frac{k^n}{k!} \text{ as } n \rightarrow \infty.$$

Other approximations exist, such as Temme's approximation, for example.

[0123] Using the approximation given above for large n , the probability mass function of the cluster cardinality distribution approaches that of the binomial distribution:

$$P_b(b|N, C) = \frac{1}{C} \binom{N}{b} \frac{\{C-1\}}{\{N\}} \approx \binom{N}{b} \left(\frac{1}{C}\right)^b \left(1 - \frac{1}{C}\right)^{N-b}.$$

Moreover, the approximation of $P_b(b|N, C)$ allows empty clusters. Thus, embodiments that use the approximation are examples of embodiments that allow empty clusters, the resulting EI may not be an approximation, and the resulting EI may only be an approximation if all clusters are constrained to be non-empty. Additionally, for cases when N is much larger than C , the empty cluster case has a very small probability of occurring randomly, i.e., $P(0)$ is close to zero as is $P(b)$ for $b > N - C + 1$, and thus the $b = 0$ and $b > N - C + 1$ cases can sometimes be effectively ignored.

[0124] Using this approximation, in some embodiments EI may be calculated as follows:

$$EI(a, C) = \sum_{i=1}^R \sum_{j=1}^C \sum_{b=1}^{N-C+1} \left\{ \binom{N}{b} \left(\frac{1}{C}\right)^b \left(1 - \frac{1}{C}\right)^{N-b} \sum_{n=(a_i+b-N)^+}^{\min(a_i, b)} \frac{\binom{b}{n} \binom{N-b}{a_i-n}}{\binom{N}{a_i}} \frac{n}{N} \log \frac{nN}{a_i b} \right\}.$$

Since these embodiments are based on mutual information, the expected index may be referred to as “Expected Mutual Information,” or “EMI.”

[0125] Additionally, in some embodiments the EI may be calculated as follows:

$$EI(a, C) = \left(1 - \frac{\sum_{i=1}^R (a_i)_2}{(N)_2}\right) \left(1 - \frac{1}{C}\right) + \frac{\sum_{i=1}^R (a_i)_2}{C(N)_2}.$$

Since these embodiments are based on the Rand index, the expected index may be referred to as “Expected Rand Index,” or “ERI.”

[0126] FIG. 14 is a block diagram that illustrates an example embodiment of a system 1400 for validating object clusters. The system includes a cluster validation device 1410 and an object storage device 1420, both of which are computers (e.g., a desktop computer, a server, a PDA, a laptop, a tablet, a phone). The cluster validation device includes one or more processors (CPUs) 1411, I/O interfaces 1412, and storage 1413. The CPUs 1411 includes one or more central processing units (e.g., microprocessors) and are configured to read and perform computer-executable instructions, such as instructions stored in the modules. Note that the computer-executable instructions may include those for the performance of the methods described herein. The I/O interfaces 1412 provide communication interfaces to input and output devices, which may include a keyboard, a display, a mouse, a printing device, a touch screen, a light pen, an optical storage device, a scanner, a microphone, a camera, a drive, and a network (either wired or wireless).

[0127] Storage/RAM 1413 includes one or more computer readable and/or writable media, and may include, for example, a magnetic disk (e.g., a floppy disk, a hard disk), an optical disc (e.g., a CD, a DVD, a Blu-ray), a magneto-optical disk, a magnetic tape, semiconductor memory (e.g., a non-volatile memory card, flash memory, a solid state drive, SRAM, DRAM), an EPROM, an EEPROM, etc. Storage/RAM 1413 may store computer-readable data and/or instructions. The members of the cluster validation device 1410 communicate via a bus.

[0128] The cluster validation device 1410 also includes a clustering procedure module 1414, an object labels module 1415, and a cluster validation module 1416. The clustering procedure module 1414 includes computer-executable instructions that may be executed by the cluster validation device 1410 to cause the cluster validation device 1410 to perform a clustering procedure to cluster objects (e.g., extract features from objects, assign the objects to respective clusters). The object labels module 1415 includes computer-executable data and/or instructions that indicate labels for respective objects, and the labels may be used to cluster the objects, for example by the cluster validation device 1410. The cluster validation module 1416 includes computer-executable instructions that may be executed to cause the cluster validation device to compare clusters created by a clustering procedure to the object labels, for example by determining an expected index (EI) or an adjusted index (AI) of the clusters created by the clustering procedure and the object labels. The cluster validation module 1416 may be executed by the cluster validation device to cause the cluster validation device to perform the methods described herein.

[0129] The object storage device 1420 includes a CPU 1422, storage/RAM 1423, and I/O interfaces 1424. The object storage device also includes object storage 1421. Object storage 1421 includes a computer-readable medium that stores objects (e.g., images, video, documents) thereon. The members of the object storage device 1420 communicate via a bus. The cluster validation device 1410 may retrieve objects from the object storage 1421 on the object storage device 1420 via a network 1430, which may include one or more wired or wireless networks.

[0130] FIG. 15A is a block diagram that illustrates an example embodiment of a system 1500A for validating object clusters. The system includes a cluster validation device 1510, an object storage device 1520, and a clustering device 1540. The cluster validation device 1510 includes a CPU 1511, I/O interfaces 1512, a cluster validation module 1513, and storage/RAM 1514. The object storage device 1520 includes a CPU 1522, I/O interfaces 1524, object storage 1521, and storage/RAM 1523. The clustering device 1540 includes a CPU 1541, I/O interfaces 1542, an object labels module 1543, and a clustering

procedure module 1544. The members of each of the devices communicate via a respective bus. In the embodiment shown in FIG. 15A, the clustering procedure device 1540 generates clusters using the object labels 1543 and/or the clustering procedure module 1544. The clustering procedure device 1540 communicates with the object storage device 1520 via a network 1530 to access the objects in the object storage 1521. Thus, in this embodiment, different devices may store the objects, cluster the objects, and validate the clusters.

[0131] FIG. 15B is a block diagram that illustrates an example embodiment of a system 1500B for validating object clusters. The system includes a cluster validation device 1550 that includes a CPU 1551, I/O interfaces 1552, object storage 1553, a clustering procedure module 1554, an object labels module 1555, a cluster validation module 1556, and RAM/storage (not shown). The members of the cluster validation device 1550 communicate via a bus. Therefore, in the embodiment shown in FIG. 15B, one computing device stores the objects, clusters the objects, and validates the clusters. However, other embodiments may organize the components differently than the example embodiments shown in FIG. 14, FIG. 15A, and FIG. 15B.

[0132] The above described devices, systems, and methods can be achieved by supplying one or more storage media having stored thereon computer-executable instructions for realizing the above described operations to one or more computing devices that are configured to read the computer-executable instructions stored in the one or more storage media and execute them. In this case, the systems and/or devices perform the operations of the above-described embodiments when executing the computer-executable instructions read from the one or more storage media. Also, an operating system on the one or more systems and/or devices may implement the operations of the above described embodiments. Thus, the computer-executable instructions and/or the one or more storage media storing the computer-executable instructions therein constitute an embodiment.

[0133] Any applicable computer-readable storage medium (e.g., a magnetic disk (including a floppy disk, a hard disk), an optical disc (including a CD, a DVD, a Blu-ray disc), a magneto-optical disk, a magnetic tape, and a solid state memory (including flash

memory, DRAM, SRAM, a solid state drive)) can be employed as a storage medium for the computer-executable instructions. The computer-executable instructions may be written to a computer-readable storage medium provided on a function-extension board inserted into the device or on a function-extension unit connected to the device, and a CPU provided on the function-extension board or unit may implement the operations of the above-described embodiments.

[0134] This disclosure has provided a detailed description with respect to particular representative embodiments. It is understood that the scope of the appended claims is not limited to the above-described embodiments and that various changes and modifications may be made without departing from the scope of the claims.

WHAT IS CLAIMED IS:

1. A method for measuring similarity between a set of clusters generated by a clustering procedure and a corresponding set of object labels, wherein the clustering procedure is for grouping the set of objects, the method comprising:

applying a clustering procedure to a set of objects to cluster the objects into a first set of clusters;

calculating a similarity index between the set of clusters and a set of object labels;

calculating an expected value of the similarity index, wherein the expected value of the similarity index is a value of the similarity index one would expect to obtain, on average, between a randomly generated second set of clusters and the object labels; and

adjusting the calculated similarity index by a penalty factor based at least in part on the expected value of the similarity index.

2. The method according to Claim 1, wherein every possible way to cluster the objects into the randomly generated second set of clusters is assumed to be equally likely.

3. The method according to Claim 2, wherein a probability of a randomly generated cluster in the second set of clusters having a certain number of objects is given by the distribution

$$P(b) = \frac{1}{C} \binom{N}{b} \frac{\{N-b\}}{\{C-1\}},$$

wherein b is the certain number of objects in the cluster, N is a total number of objects in the set of objects, C is a total number of clusters in the set of clusters, and $\{n\}$ is a Stirling number of the second kind.

4. The method according to Claim 2, wherein a probability of a randomly generated cluster in the second set of clusters having a certain number of objects is given by or approximated by a binomial distribution with $p = 1/C$,

$$P(b) = \binom{N}{b} \left(\frac{1}{C}\right)^b \left(\frac{C-1}{C}\right)^{N-b},$$

wherein C is the total number of clusters in the set of clusters, b is the certain number of objects in the cluster, and N is the total number of objects in the set of objects.

5. The method according to Claim 1, wherein the similarity index is calculated using mutual information between a set of clusters and the object labels.

6. The method according to Claim 5, wherein an expected value of the mutual information is calculated according to

$$EMI(a, C) = \sum_{i=1}^R \sum_{b=1}^{N-C+1} \sum_{n=\max(1, a_i+b-N)}^{\min(a_i, b)} \frac{n}{N} \log \frac{nN}{a_i b} \frac{\binom{a_i}{n} \binom{N-a_i}{b-n} \{C-1\}^{N-b}}{\{N\}^C},$$

wherein a_i is a number of objects belonging to an i -th label for each i from 1 to R , R is a total number of object labels, C is a total number of clusters in the set of clusters, and N is a total number of objects.

7. The method according to Claim 1, wherein the similarity index is calculated using a Rand Index between the first set of clusters and the object labels.

8. The method according to Claim 7, wherein the expected value of the similarity index is calculated according to

$$EI(a, C) = \left(1 - \frac{\sum_{i=1}^R (a_i)_2}{(N)_2}\right) \left(1 - \frac{\{C-1\}}{\{N\}}\right) + \frac{\sum_{i=1}^R (a_i)_2 \{C-1\}}{(N)_2 \{N\}},$$

wherein a_i is a number of objects belonging to an i -th label for each i from 1 to R , N is a total number of objects in the set of objects, R is a number of object labels, C is a number of clusters in the set of clusters, and $(x)_n$ is the falling factorial of x .

9. The method according to Claim 1, wherein the predetermined categorization of objects is selected by a user.
10. The method according to Claim 1, wherein the predetermined categorization of objects is generated based on past categorizations of objects.
11. The method according to Claim 1, wherein the predetermined categorization of objects is generated using a pre-labeled learning set of objects.
12. The method according to Claim 1, wherein applying the clustering procedures comprises:
- extracting object features from each object in the set of objects;
 - determining a comparison measure by which to compare respective features of the objects in the set of objects;
 - comparing respective features of the objects in the set of objects based on the comparison measure to determine differences between respective features of the objects; and
 - outputting a group of measures representing the differences between respective features of the objects,
- wherein the first clustering procedure is applied to the set of objects based at least in part on the group of measures.
13. The system for measuring similarity between a first set of clusters generated by a first clustering procedure and a set of object labels, the system comprising:
- a first computer-readable medium configured to store computer-executable instructions; and
 - one or more processors configured to cause the system to
 - retrieve a set of objects from one or more computer-readable media,

apply a first clustering procedure to the set of objects to sort the objects into a first set of clusters, wherein the first set of clusters has a first number of clusters,

retrieve a set of object labels from one or more computer-readable media,

generate a similarity index between the first set of clusters and the set of object labels,

generate an expected value of the similarity index, wherein the expected value of the similarity index indicates an average value of the similarity index between randomly generated sets of clusters and the set of object labels, wherein the randomly generated sets of clusters have at most a same number of clusters as the first number of clusters, and

adjust the calculated similarity index by a penalty factor that is based at least in part on the expected value of the similarity index.

14. The system according to Claim 13, wherein the randomly generated sets of clusters are different from one another.

15. The system according to Claim 13, wherein

in the aggregate, the average value of the similarity index between the randomly generated sets of clusters and the set of object labels is the average of the similarity index between every possible arrangement of the objects into a number of clusters equal to the first number of clusters and the set of object labels.

16. The system according to Claim 15, wherein the one or more processors are further configured to cause the system to generate the expected value of the similarity index according to

$$EI(a, C) = \sum_{i=1}^R \sum_{b=1}^{N-C+1} \sum_{n=\max(1, a_i+b-N)}^{\min(a_i, b)} \frac{n}{N} \log \frac{nN}{a_i b} \frac{\binom{a_i}{n} \binom{N-a_i}{b-n} \{C-1\}^{N-b}}{\{C\}^N},$$

wherein a_i is a number of objects belonging to an i -th label for each i from 1 to R , R is a total number of object labels, C is the first number of clusters, and N is a total number of objects.

17. One or more computer-readable media storing instructions that, when executed by one or more computing devices, cause the one or more computing devices to perform operations comprising:

- retrieving a set of objects from one or more computer-readable media;
- generating a first set of clusters based on a first clustering procedure and the set of objects;
- calculating a similarity index between the first set of clusters and a set of object labels;
- calculating an expected value of the similarity index, wherein the expected value of the similarity index is a weighted average of possible values of a similarity index between a randomly generated set of clusters and the object labels; and
- modifying the calculated similarity index by a penalty factor that is based at least in part on the expected value of the similarity index.

18. The one or more computer-readable media of Claim 17, wherein the expected value is a weighted average of the similarity index between the randomly generated set of clusters and the object labels.

19. The one or more computer-readable media of Claim 17, wherein the weighted average assumes all possible randomly generated sets of clusters are equally likely.

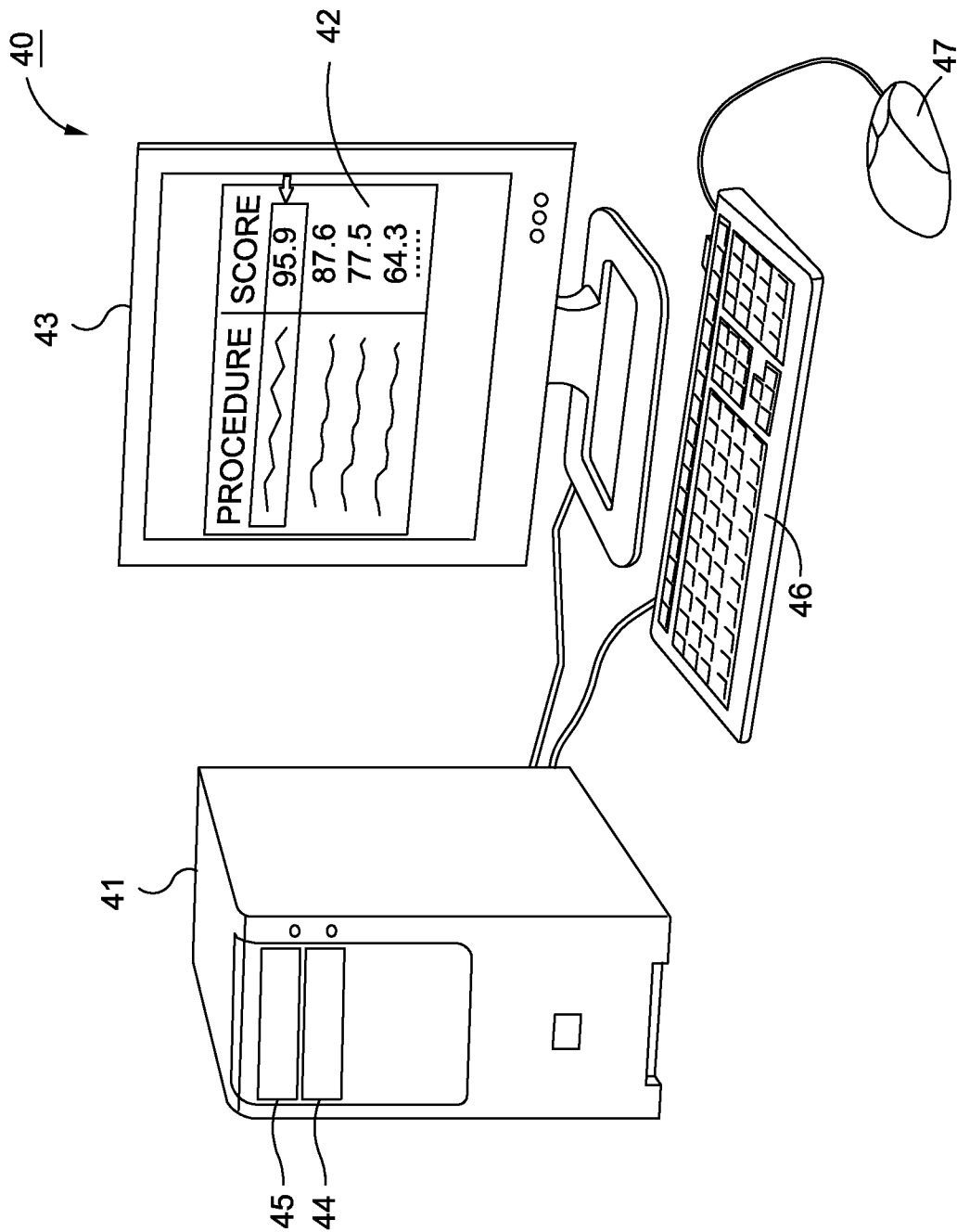


FIG. 1

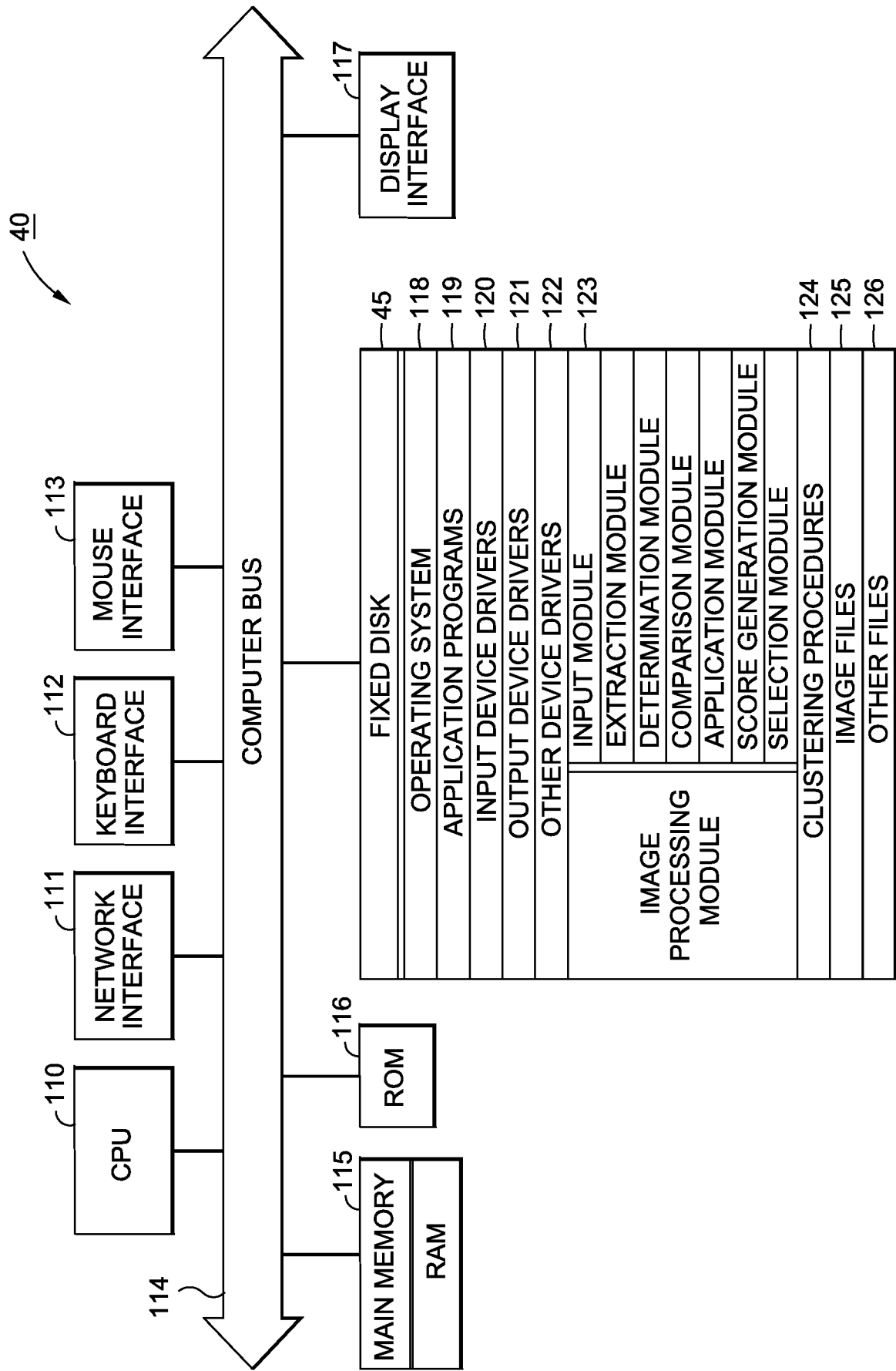


FIG. 2

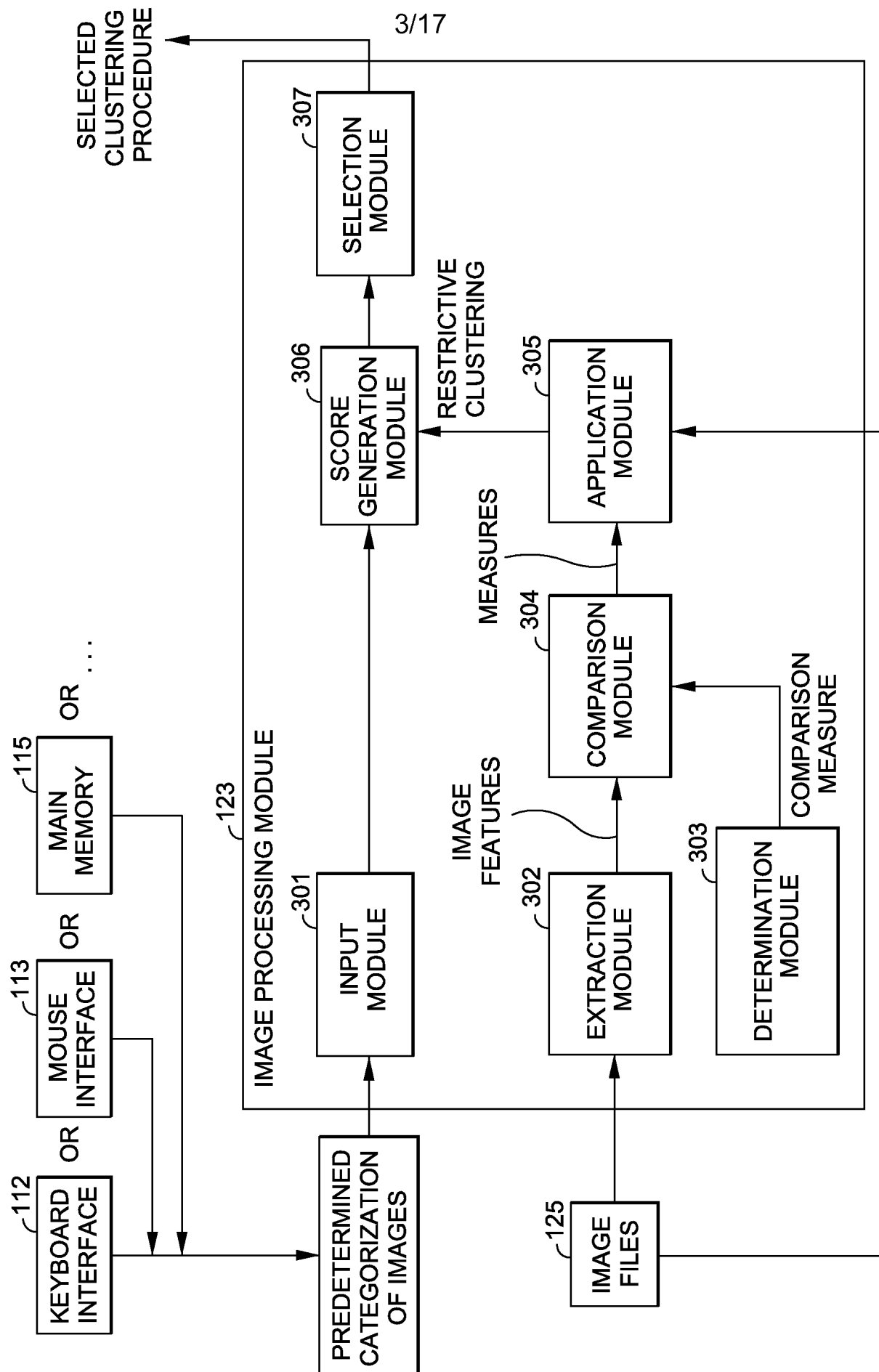
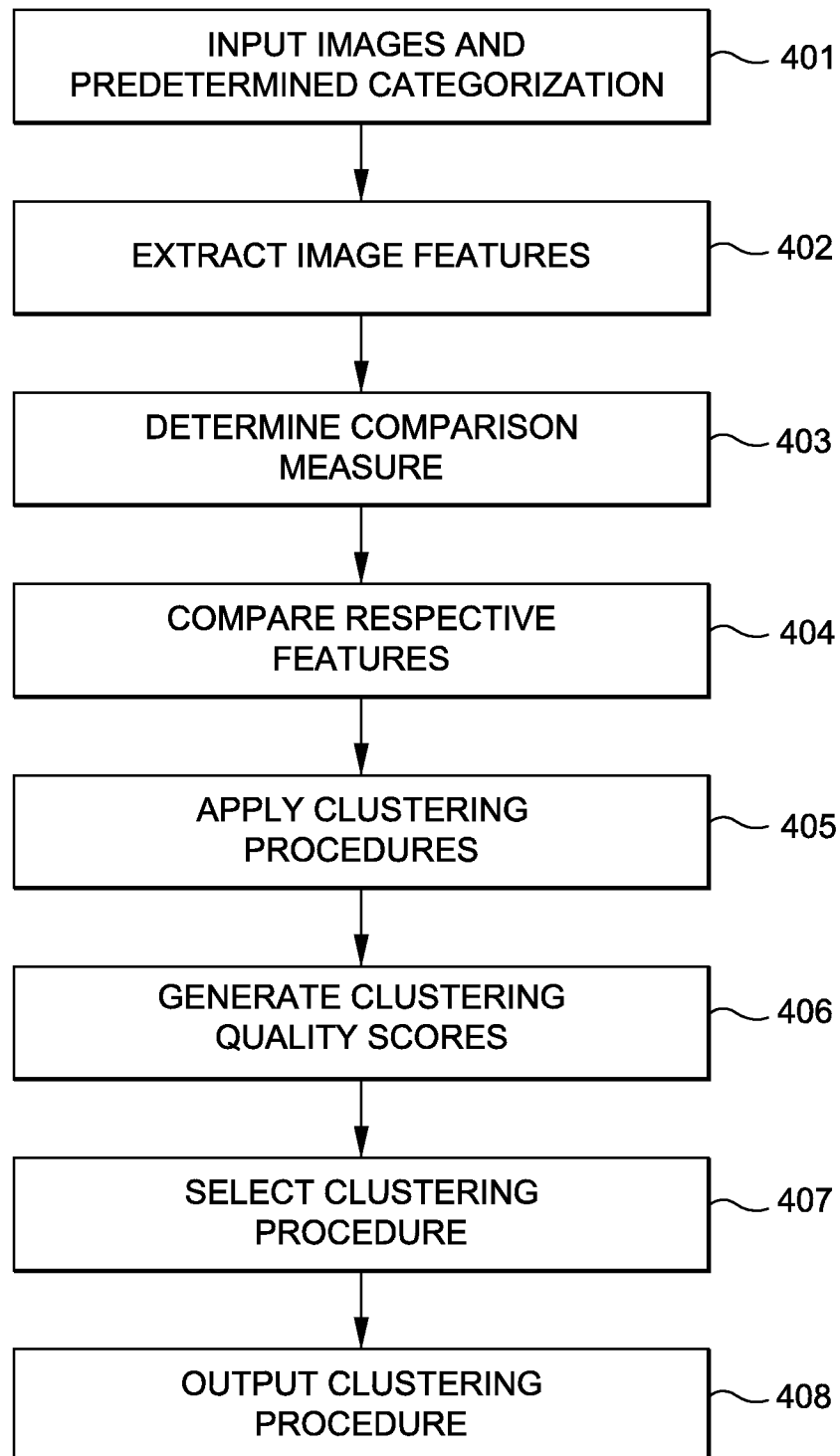


FIG. 3

4/17

**FIG. 4**

v/ν	v_1	v_2	...	v_c	Sums
u_1	M_{11}	M_{12}	...	M_{1c}	a_1
u_2	M_{21}	M_{22}	...	M_{2c}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
u_R	M_{R1}	M_{R2}	...	M_{Rc}	a_R
Sums	b_1	b_2	...	b_c	N

FIG. 5A

2	0
0	2

(a) U vs. V1

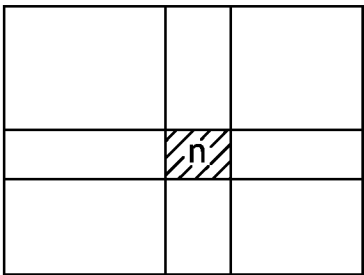
2	0	0
0	1	1

(b) U vs. V2

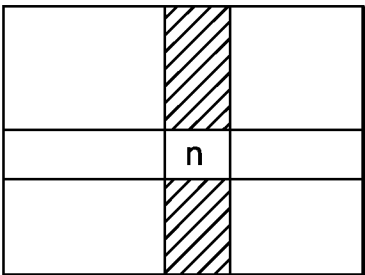
1	1	0	0
0	0	1	1

(c) U vs. V3

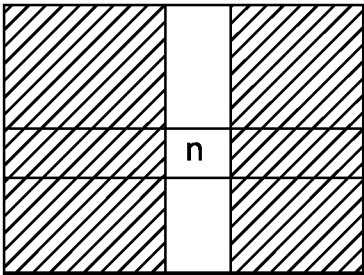
FIG. 5B



(a)



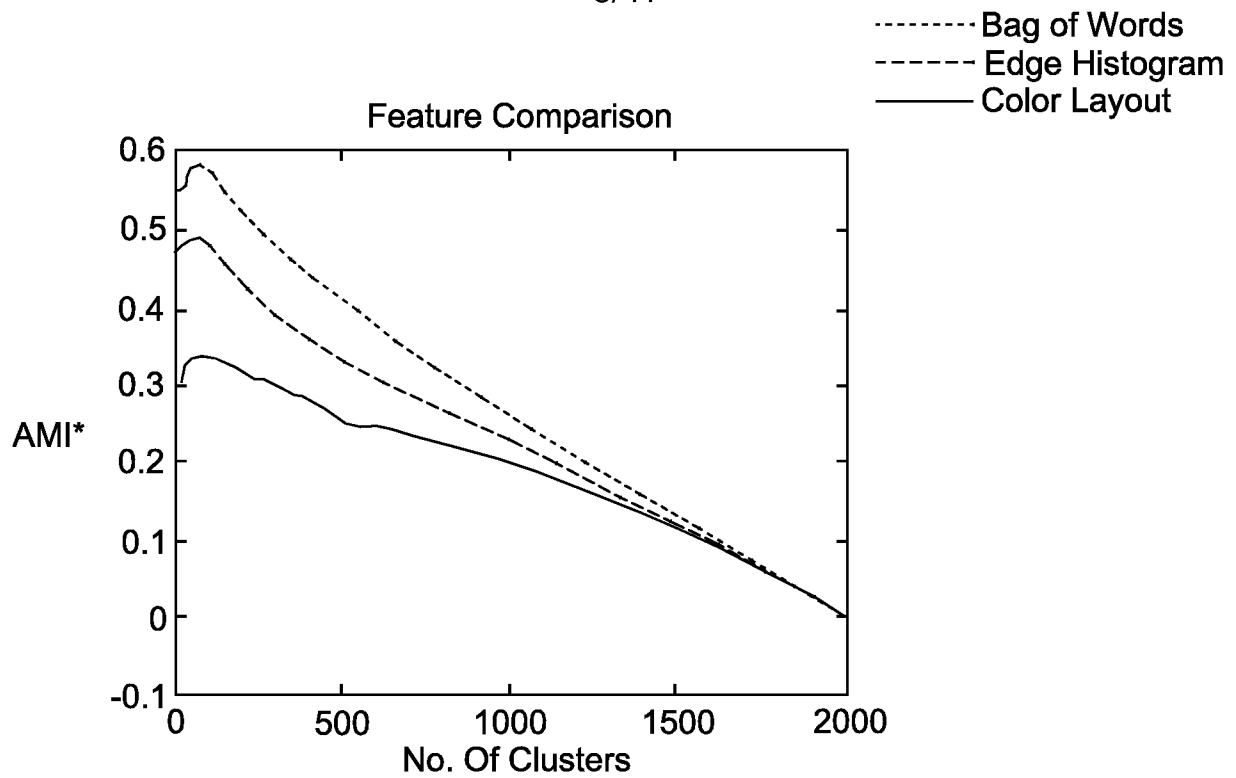
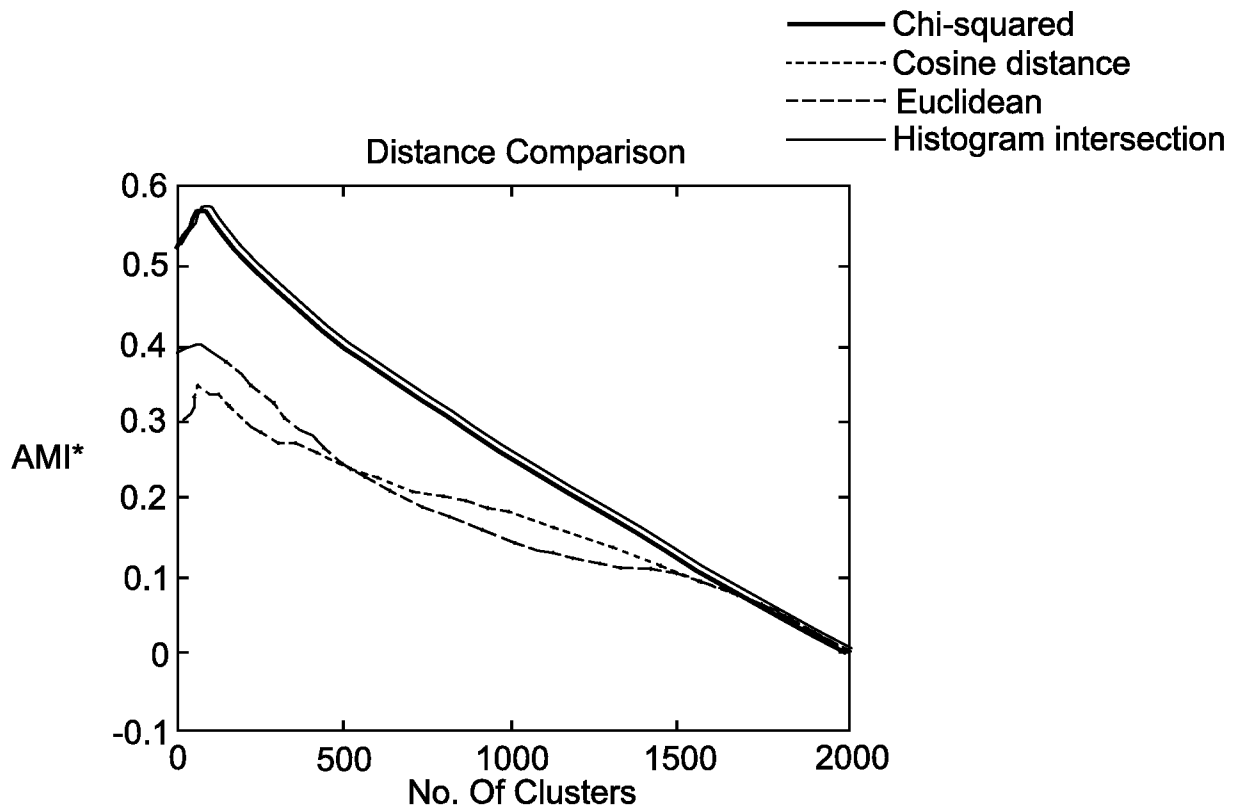
(b)



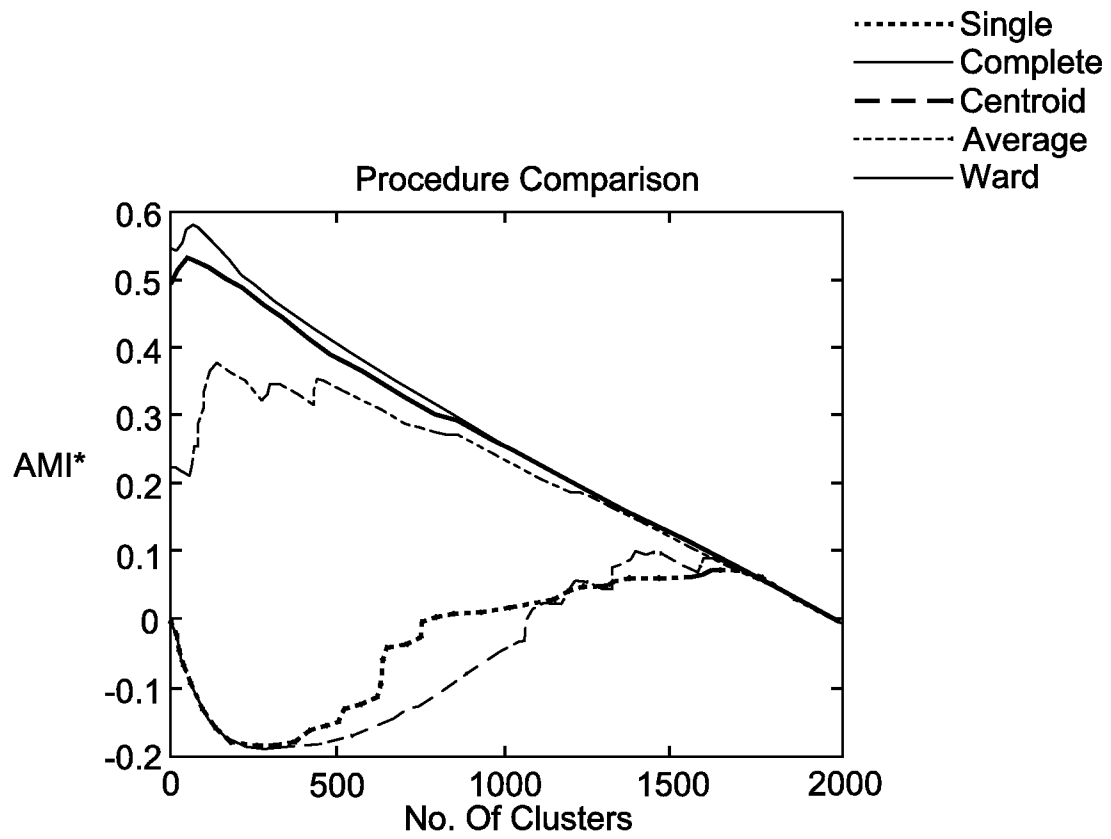
(c)

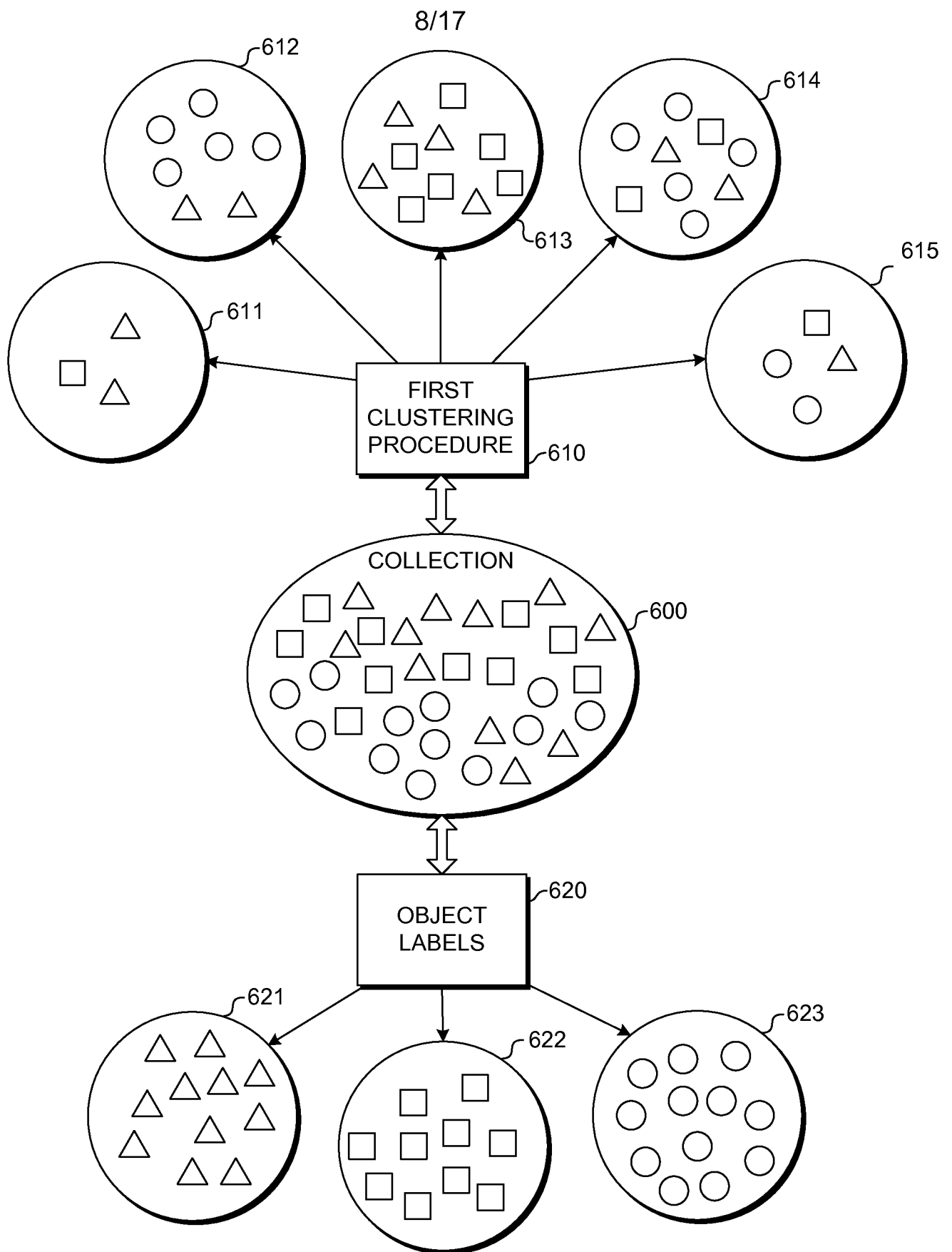
FIG. 5C

6/17

**FIG. 5D****FIG. 5E**

7/17

**FIG. 5F**

**FIG. 6**

700A

9/17

U / V	V ₁	V ₂	V ₃	V ₄	V ₅	Sums
u ₁	M ₁₁	M ₁₂	M ₁₃	M ₁₄	M ₁₅	11
u ₂	M ₂₁	M ₂₂	M ₂₃	M ₂₄	M ₂₅	10
u ₃	M ₃₁	M ₃₂	M ₃₃	M ₃₄	M ₃₅	12
Sums	b ₁	b ₂	b ₃	b ₄	b ₅	33

730A

FIG. 7A

700B

U / V	V ₁	V ₂	V ₃	V ₄	V ₅	Sums
△	2	2	4	2	1	11
□	1	0	6	2	1	10
○	0	5	0	5	2	12
Sums	3	7	10	9	4	33

710B

751B

753B

730B

740B

FIG. 7B

700C

(Randomly Generated)

U / V	V ₁	V ₂	V ₃	V ₄	V ₅	Sums
△	1	2	2	4	2	11
□	1	3	4	2	0	10
○	1	2	4	3	2	12
Sums	3	7	10	9	4	33

751C

FIG. 7C

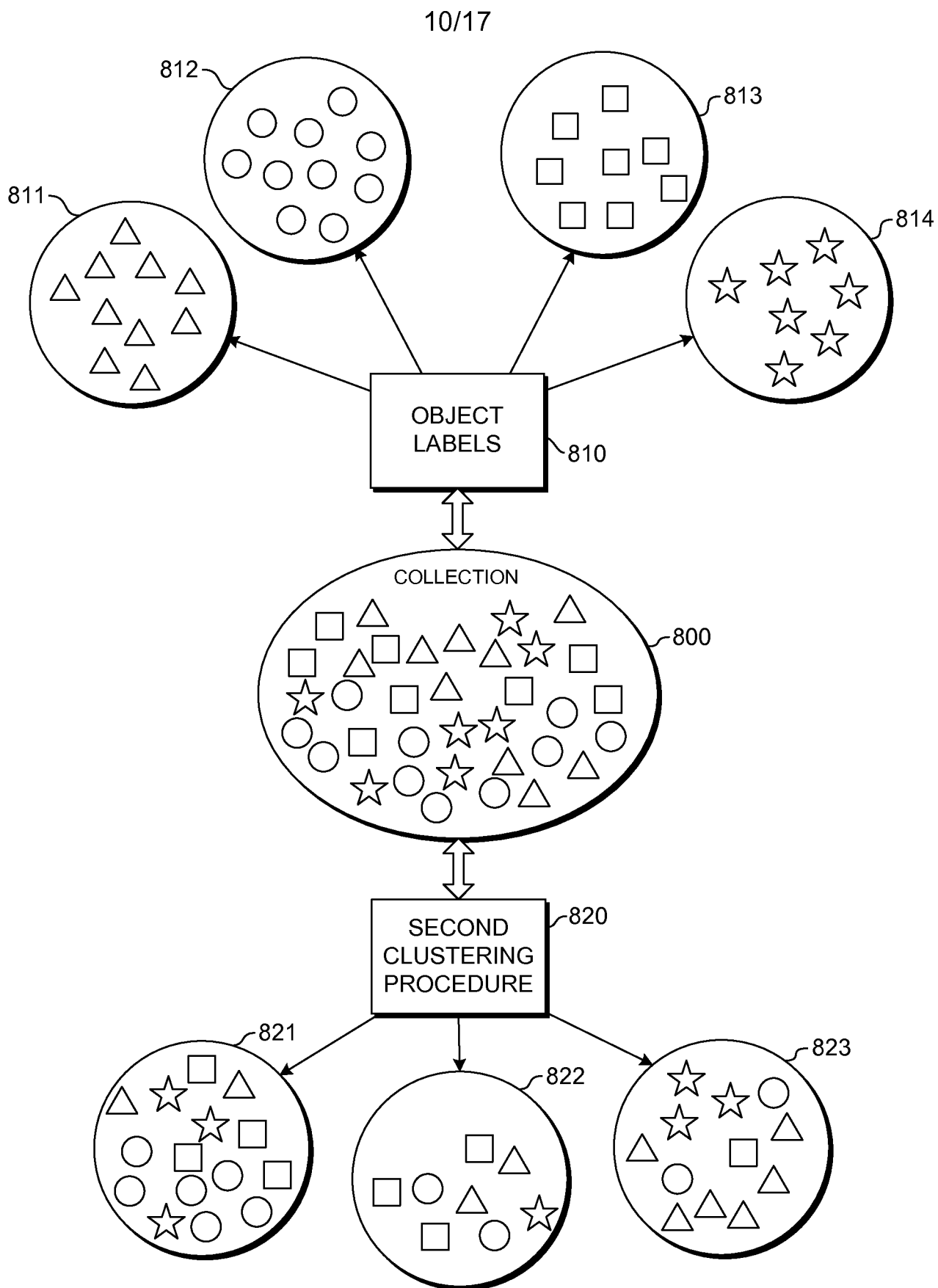


FIG. 8

11/17

900A

U / V	V ₁	V ₂	V ₃	Sums
△	2	2	6	10
○	6	2	2	10
□	4	3	1	8
☆	3	1	3	7
Sums	15	8	12	35

951A

910A

930A

953A

940A

FIG. 9A

900B

(Randomly Generated)

U / V	V ₁	V ₂	V ₃	Sums
△	3	3	4	10
○	3	4	3	10
□	0	6	2	8
☆	5	1	1	7
Sums	11	14	10	35

951A

FIG. 9B

12/17

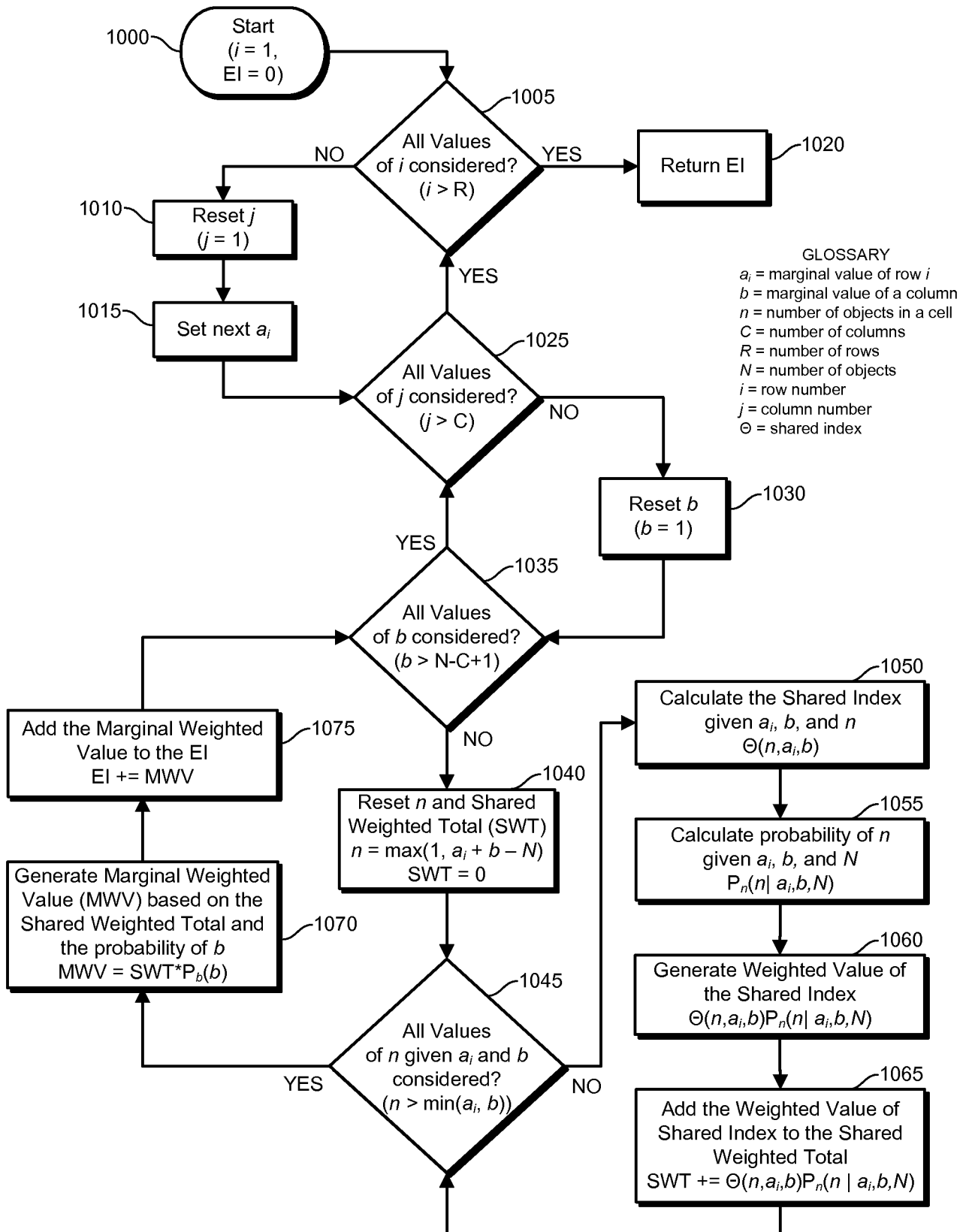


FIG. 10

13/17

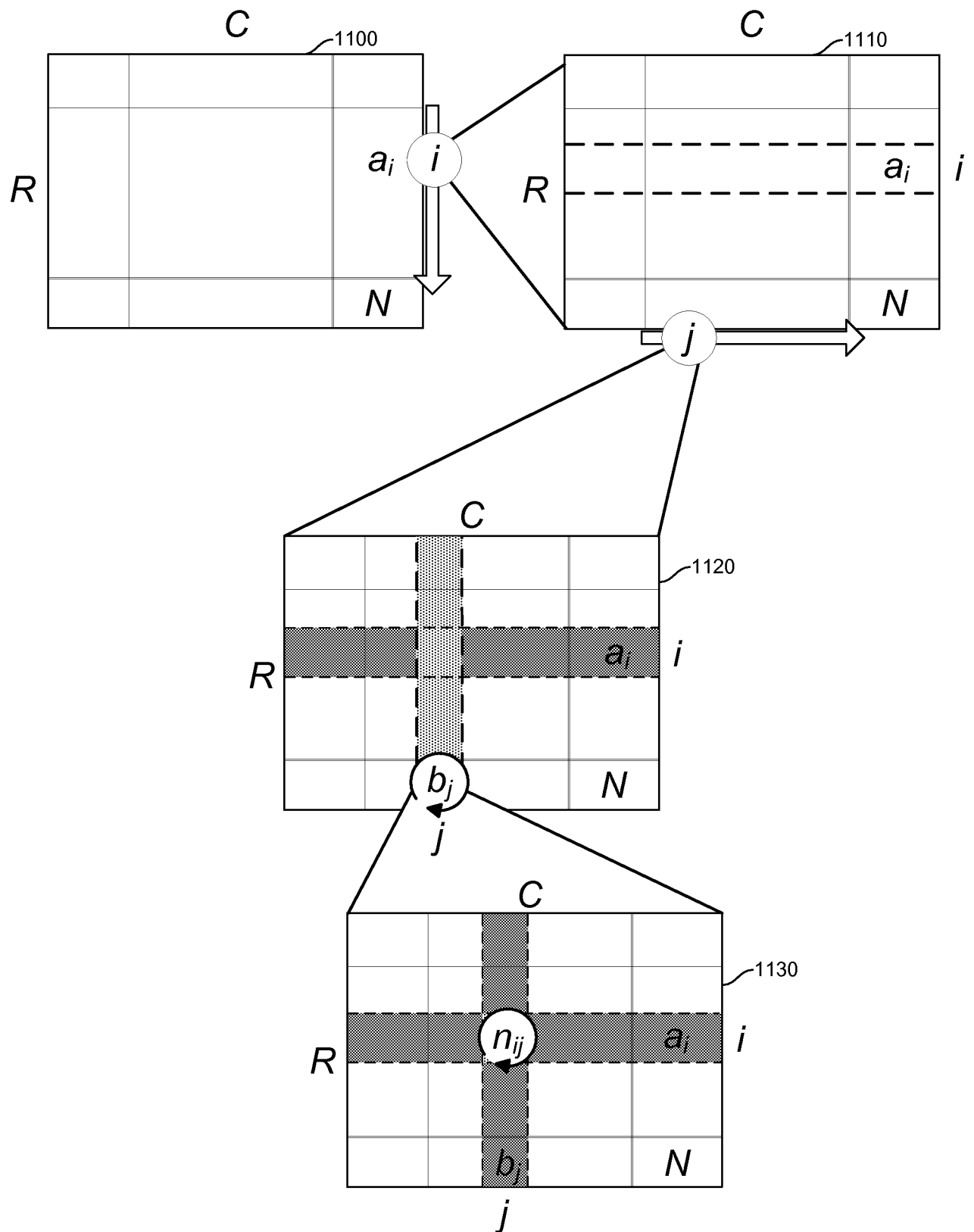


FIG. 11

14/17

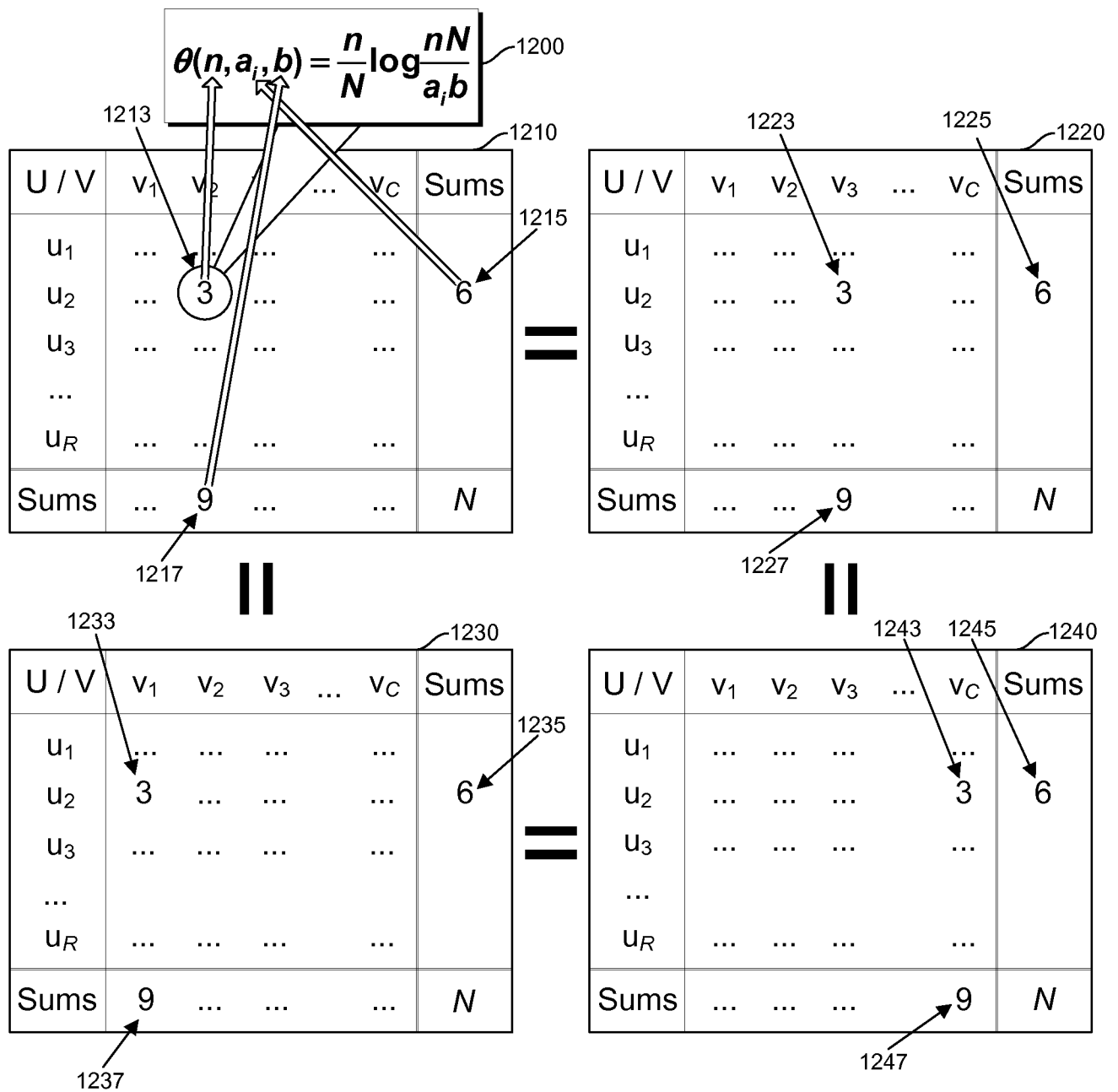


FIG. 12

15/17

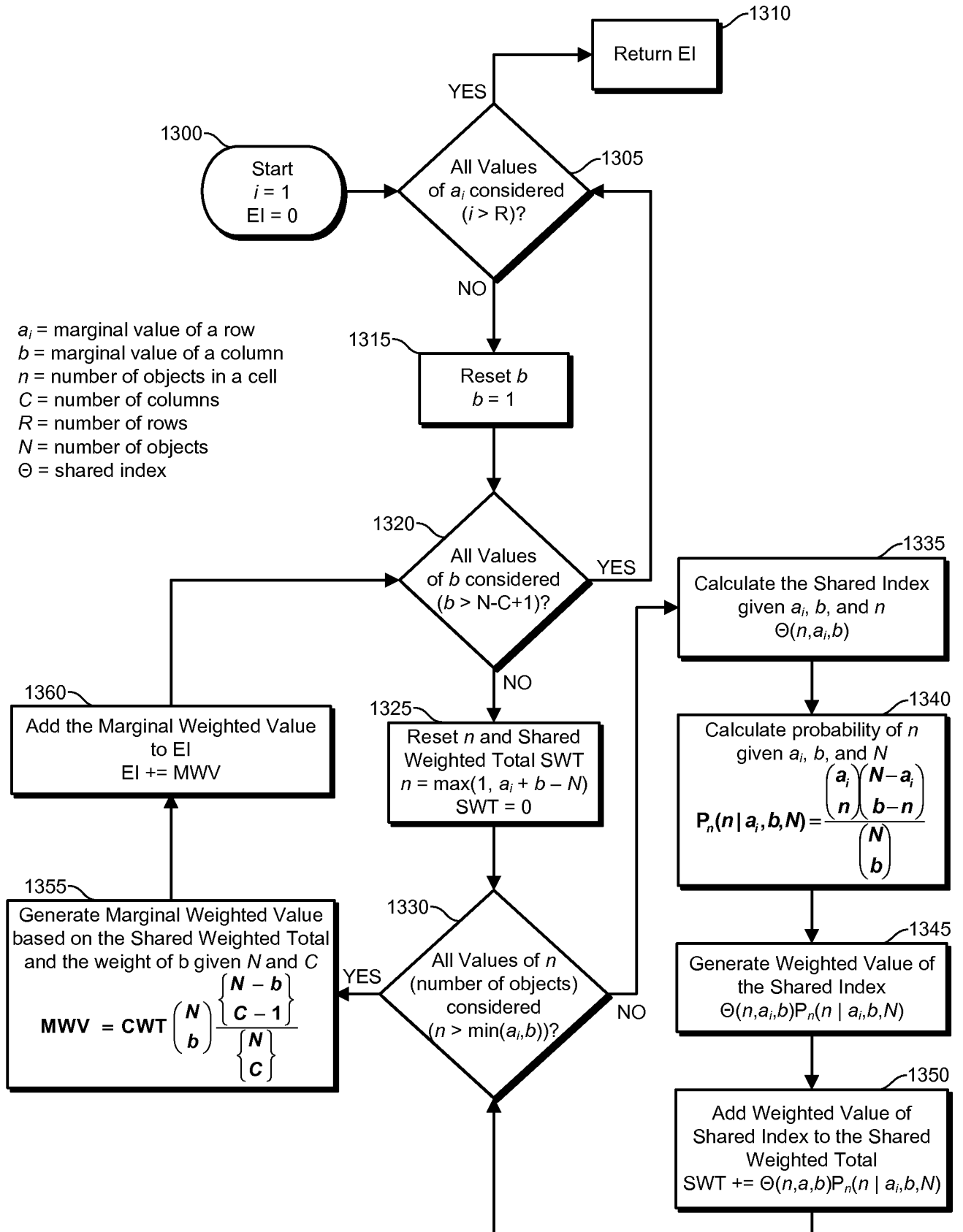
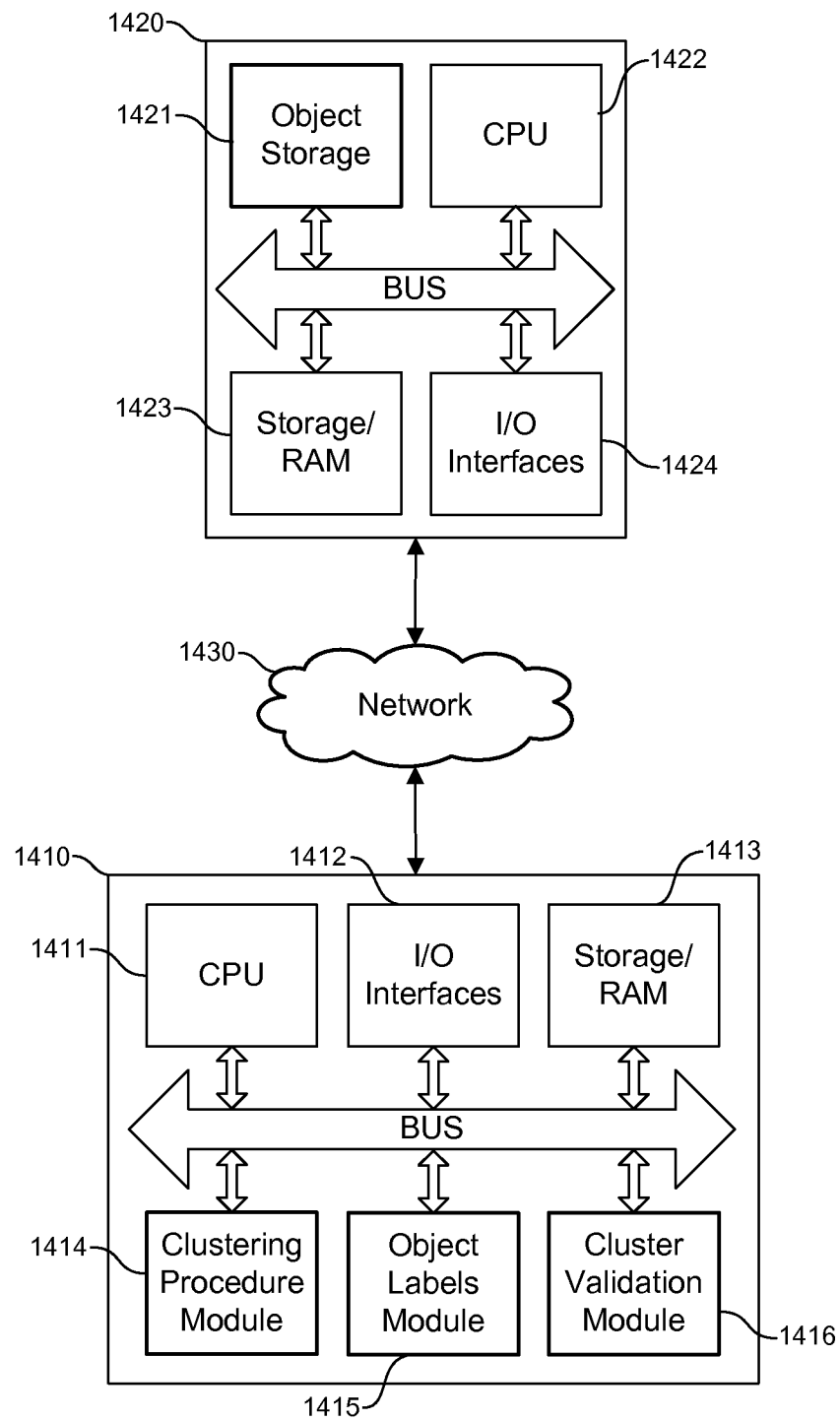
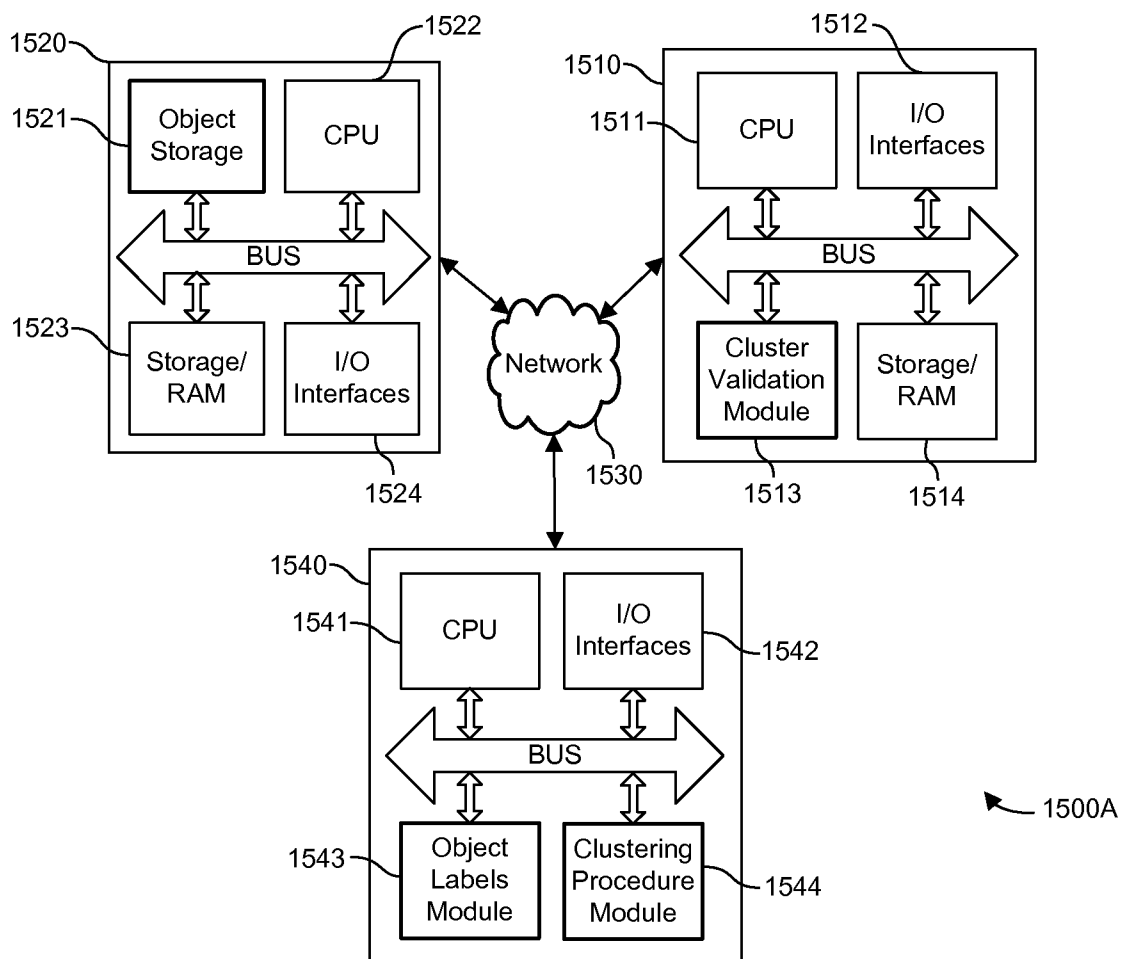
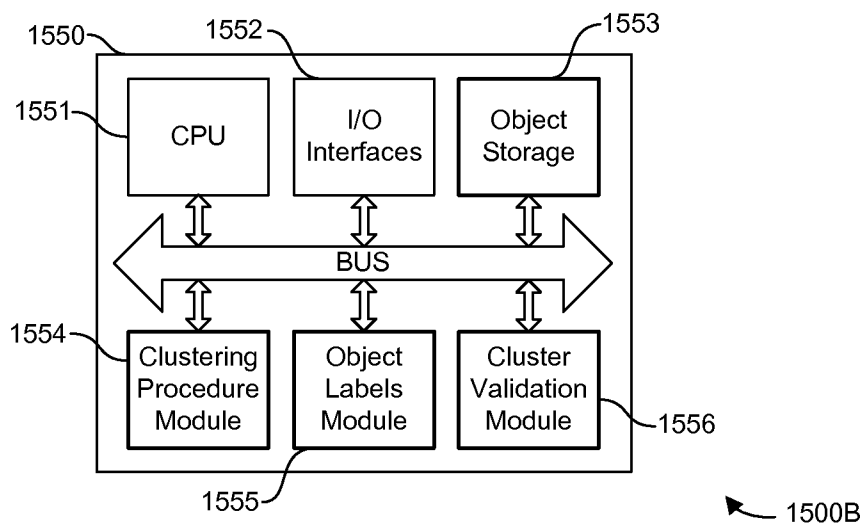


FIG. 13

16/17

**FIG. 14**

17/17

**FIG. 15A****FIG. 15B**

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US 11/56441

A. CLASSIFICATION OF SUBJECT MATTER

IPC(8) - G06K 9/00 (2012.01)

USPC - 382/170

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

UPC: 382/170; IPC: G06K 9/00 (2012.01)

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
382/170, 192, 194; 707/999.006, E17.024; G06K 9/00, G06K5

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

PubWest (PGPB, USPT, EPAB, JPAB); Google (Patents, Scholar)

Search Terms: Cluster\$5, group\$4, cateror\$6, character\$4, array, set, Measure\$, calculate\$5, determin\$4, survey\$3, evaluat\$3, Similiar\$5, mutual, alike, like, Index\$5, rat\$5, Object\$2, image\$2, basis, Label\$2, tag\$5, identify\$4, class\$5, Expect\$3, average, random\$4

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	Kuncheva, L.I., Hadjitodorov, S.T., "Using Diversity in Cluster Ensembles", 2004 IEEE International Conference, Systems, Man and Cybernetics, October 10-13, 2004 [Online], retrieved on 28 February 2012, retrieved from the Internet, <http://pages.bangor.ac.uk/~mas00a/papers/lkSMC04.pdf 10-13 2004>, abstract, pg 1214-1216	1-12 -----
Y		13-19
Y	US 2010/0092073 A1 (Prokhorov) 15 April 2010 (15.04.2010), entire document, especially abstract, para [0022],[0044].	13-19
Y	US 2007/0061319 A1 (Bergholz) 15 March 2007 (15.03.2007) (para [0034]-[0040],[0042], [0048]-[0052])	1-19
Y	US 2010/0153318 A1 (Branavan et al.) 17 June 2010 (17.06.2010) (para [0041]-[0044],[0087]-[0089],[0100]-[0102],[0159])	1-19
A	US 5,438,676 A (Schwanke) 01 August 1995 (01.08.1995), entire document	1-19
A/P	US 2011/0134128 A1 (Hu et al.) 09 June 2011 (09.06.2011), entire document	1-19
L	CITATION: Kuncheva, L.I., Hadjitodorov, S.T., "Using Diversity in Cluster Ensembles", 2004 IEEE International Conference, Systems, Man and Cybernetics, October 10-13, 2004 [Online], retrieved on 28 February 2012, retrieved from the Internet, <http://pages.bangor.ac.uk/~mas00a/papers/lkSMC04.pdf 10-13 2004>	1-19

☐ Further documents are listed in the continuation of Box C.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

28 February 2012 (28.02.2012)

Date of mailing of the international search report

08 MAR 2012

Name and mailing address of the ISA/US

Mail Stop PCT, Attn: ISA/US, Commissioner for Patents
P.O. Box 1450, Alexandria, Virginia 22313-1450

Facsimile No. 571-273-3201

Authorized officer:

Lee W. Young

PCT Helpdesk: 571-272-4300
PCT OSP: 571-272-7774