(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
1 December 2016 (01.12.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/190851 A1

(51) International Patent Classification:
G06F 19/26 (2011.01) *G06F 17/30* (2006.01)
G06F 19/00 (2011.01)

(21) International Application Number:
PCT/US2015/032402

(22) International Filing Date:
26 May 2015 (26.05.2015)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **WILLIAM MARSH RICE UNIVERSITY** [US/US]; 6100 Main Street, Houston, Texas 77005 (US).

(72) Inventors: **BISBERG, Alex**; 6100 Main Street, MS142, Room 613, Houston, Texas 77030 (US). **HU, Chenyue**; 6500 Main Street, MS142, Houston, Texas 77030 (US). **QUTUB, Amina A.**; 6100 Main Street, MS142, Houston, Texas 77005 (US).

(74) Agent: **LENART, Edward J.**; KENNEDY LENART SPRAGGINS LLP, 8601 Ranch Road 2222, STE. 1-225, Austin, Texas 78730 (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, BN, 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, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, 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, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, 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, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

(54) Title: PRESENTING A DIGITAL ORGANISM WITHIN A DIGITAL MAP

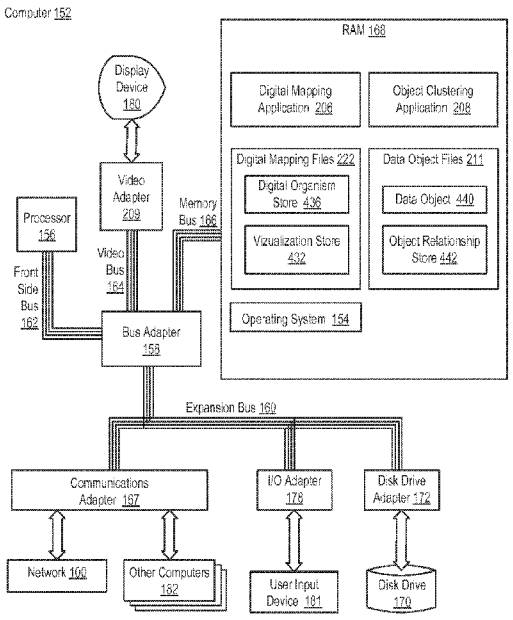


FIG. 1

(57) Abstract: Presenting a digital organism within a digital map, including: receiving a plurality of digital organism objects associated with a digital organism, each digital organism object comprises a biological layer indicator and a structure indicator; and rendering a portion of the digital organism within the digital map based at least on a first digital organism object, the first digital organism object being one of the plurality of digital organism objects.

WO 2016/190851 A1

PRESENTING A DIGITAL ORGANISM WITHIN A DIGITAL MAP

Inventors: Alex Bisberg
Chenyue Hu
Amina A. Qutub

STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR DEVELOPMENT

The invention was made with United States Government support under Grant No. 1150645 awarded by the National Science Foundation and Grant Nos. GM106027 and GM 106027-02 awarded by the National Institutes of Health. The United States Government has certain rights in the invention.

BACKGROUND OF THE INVENTION

Field of the Invention

The field of the invention is data processing, or, more specifically, methods, apparatus, and products for presenting a digital organism within a digital map.

Description Of Related Art

As medical research progresses alongside modern technology, it has become increasingly important to be able to store biological-related data and medical-related data in a manner that facilitates the usage of such data in research and treatment. In addition to storing biological-related data and medical-related data, improving the visualization of the data increases its utility to the users of said data.

Generally, current biological databases exist in three main forms: (1) databases focused on a disease of interest, e.g., Alzheimer's or cancer; (2) databases that handle

data of a specific type, e.g., protein-protein interactions or genomic data; and (3) databases for a particular tissue, e.g., the human brain atlas. Generally, these databases are not interactive, and such databases rarely retain spatial information across more than one scale.

SUMMARY OF THE INVENTION

Methods, apparatuses, and products for presenting a digital organism within a digital map, including: receiving a plurality of digital organism objects associated with the digital organism, wherein each digital organism object comprises a biological layer indicator and a structure indicator; and rendering a portion of the digital organism within the digital map based at least on a first digital organism object, the first digital organism object being one of the plurality of digital organism objects.

The foregoing and other objects, features and advantages of the invention will be apparent from the following more particular descriptions of example embodiments of the invention as illustrated in the accompanying drawings wherein like reference numbers generally represent like parts of example embodiments of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 illustrates a block diagram of automated computing machinery comprising an example computer useful in presenting a digital organism within a digital map, in accordance with certain embodiments of the present disclosure.

Figure 2 illustrates a flowchart of an illustrative method for presenting a digital organism within a digital map, in accordance with certain embodiments of the present disclosure.

Figure 3 illustrates an example digital map illustrating hierarchical relationships in an organism, in accordance with certain embodiments of the present disclosure.

Figure 4 illustrates a schematic diagram of example digital DNA for rendering digital maps, in accordance with certain embodiments of the present disclosure.

Figure 5 illustrates example data structures for tissue in a digital organism, in accordance with certain embodiments of the present disclosure.

Figure 6 illustrates example hierarchical diagram of biological layers associated with a digital organism, in accordance with certain embodiments of the present disclosure.

Figure 7 illustrates a flowchart of an example method for clustering data for searching a digital map, in accordance with certain embodiments of the present disclosure.

Figure 8 illustrates an example algorithm for shrinkage clustering, in accordance with certain embodiments of the present disclosure.

Figure 9 illustrates an example algorithm for progeny sampling, in accordance with certain embodiments of the present disclosure.

Figure 10 illustrates algorithm with an example set of data, in accordance with certain embodiments of the present disclosure.

DETAILED DESCRIPTION OF EXAMPLE EMBODIMENTS

Example methods, apparatus, and products for presenting a digital organism within a digital map in accordance with the present invention are described with reference to the accompanying drawings, beginning with Figure 1. Figure 1 illustrates a block diagram of automated computing machinery comprising an example computer 152 useful in presenting a digital organism within a digital map, in accordance with certain embodiments of the present disclosure. The computer 152 of Figure 1 includes at least one computer processor 156 or 'CPU' as well as random access

memory 168 ('RAM') which is connected through a high speed memory bus 166 and bus adapter 158 to processor 156 and to other components of the computer 152.

Stored in RAM 168 is a digital mapping application 206, a module of computer program instructions for presenting a digital organism within a digital map, in accordance embodiments of the present disclosure. A digital organism may be any data structure that allows for linking between a plurality of layers, or scales, of the digital organism, such that mapping the digital organism allows for an interactive search of the digital organism between or among the plurality of scales.

The digital organism may include a plurality of features defining the digital organism. For example, the plurality of features may define the organism's: (1) biological layer (i.e., tissue, cell, protein); (2) structure (i.e., morphology or protein signaling network structure), (3) position on the digital map (i.e. adjacency to other cells, location within a tissue); and/or (4) modules needed to visualize its properties (i.e., BioWheel or microscope image). In the same or alternative embodiments, more, fewer, and/or different features may be included. For example, the features may include a biological layer indicator, a structure indicator, a map position indicator, and/or a visualization module indicator.

The plurality of features defining a digital organism may be encoded in a variety of means. For example, the features included in a digital organism may be encoded in a unique barcode or digital DNA associated with each object. The digital DNA represents a data structure where predetermined values in predetermined fields are associated with features that may be included in a particular digital organism. For example, the digital DNA may be structured such that the features of a particular digital organism are expressed using a sequence of colors. For example, a first sequence of colors may indicate that a particular digital organism is a 'cell' type biological layer whereas a second sequence of colors may indicate that a particular digital organism is a 'tissue' type biological layer. Such values may be encoded by

other means (e.g., letters or numbers) and the chosen implementation may be based on design choice made between computing speed and types of features of interest.

The digital organism may also include a plurality of digital organism objects, with each object including a plurality of defining features, and each digital organism object linked to other, related digital organism objects within the digital organism. For example, a digital organism may be structured to be a relatively higher-level biological structure, with a digital organism object associated with the relatively higher-level structure, a digital organism object associated with each of any associated secondary structures, a digital organism object associated with each of any associated tertiary structures, etc. Each of these digital organism objects are linked to the other digital organism objects associated with the digital organism, enabling a mapping and interactive search of the full digital organism.

Digital mapping application 206 may be configured to present a digital organism within a digital map by receiving a plurality of digital organism objects associated with the digital organism, where each digital organism object comprises at least a biological layer indicator and a structure indicator.

Digital mapping application 206 may be further configured to present a digital organism within a digital map by rendering a portion of the digital organism within the digital map based at least on a first digital organism object, the first digital organism object being one of the plurality of digital organism objects. The data or information necessary to render a digital organism and/or digital organism object may be encoded in a variety of ways, as detailed above. For example, if a digital organism object were encoded in digital DNA, the rendering may be based on a sequence of colors encoded in the object, and a specific set of visualization modules may be displayed to represent the organism. As a specific example, this may include the morphology of an average cell associated with this phenotype, and/or a circular BioWheel rendering the interactions of proteins within a tissue. The organism location in the map may be obtained by reading the digital DNA, and then placing the

appropriate object visualization in that location (e.g., a microscope image of brain cell is placed in its assigned location within an image of a brain slice; charts to represent that object are made available when a user clicks on the map in the object's location.)

In some embodiments, digital mapping application 206 may also be configured to enable a search function of the digital map, wherein the search function is operable to search across the plurality of digital organism objects (e.g., enabling spatial and functional searches of the digital organism). The search function associated with a digital organism may enable various users of example computer 152 to make the most of the data associated with the various digital organisms and/or digital organism objects. The digital organism objects and the digital organism object properties may be searchable by keywords, by images, and by relationships, as described in greater detail below.

Searches may be based on similarity of the search query with objects/organisms, images, words, or relationships in the map. Such searches may make use of a variety of clustering algorithms and/or clustering optimization algorithms, as described in more detail below. Searches may identify and retrieve matches to objects like tissues, cells and protein pathways; highlight in a specific color all cells in the map with a feature of interest; or look for the proteins associated with specific cell morphologies across tissues. As such, the search tool enables a way for users to identify relationships and test hypotheses prior to any experimental work.

In some embodiments, clustering may be made use of in order to group data and/or objects into clusters in order to improve, enable, and/or optimize a particular search. Clustering is an unsupervised machine learning task that aims to group objects into meaningful clusters that reflect the hidden nature of these objects based on observations of certain object features. In general, the clustering task is performed in a sequence of two major steps: (i) derive relationships between any pair of two objects using either a distance measure or a similarity measure; (ii) group objects into certain number of clusters based on the relationships obtained from the previous step.

A clustering method usually refers to the second step alone, while the first step is often treated as part of data preparation or pre-processing.

One type of clustering that may be of particular use in certain embodiments is shrinkage clustering. Shrinkage clustering takes the similarity relationships among objects as input and generates object clusters as output, with the goal of best mimicking the similarity relationships, i.e. a minimizing function (a). In some embodiments, computer 152 may also include in RAM 168 one or more object clustering application(s) 208 and one or more data object file(s) 211. Data object file(s) 211 may also include one or more data object(s) 440 and one or more object relationship store(s) 442. Object clustering application 208 may be operable to use one or more data object file(s) 211 in order to implement one or more clustering algorithms, as described in more detail below.

Object clustering application 208 is a module of computer program instructions that, when executed, automatically and rapidly groups objects into different clusters based on features of each object and their relationship to one another in a computing system according to embodiments of the present invention. The object clustering application 208 may present a user interface to a system administrator or other user that enables the system administrator or other user to initiate grouping of objects into different clusters in the computing system 152.

Data object files 211 may be configured to store data objects 440 to be clustered. Data object files 211 also includes object relationship store 442, which stores relationships between the various objects used to generate object clusters. In the example of Figure 1, object clustering application 208 may be configured to automatically and rapidly group data objects 440 into different clusters based on features of each object and their relationship to one another (e.g., as stored in object relationship store 442) in the computing system 152 in accordance with embodiments of the present disclosure.

Grouping data objects 440 in the computing system 152 may include receiving, by the object clustering application 208, a request to generate object clusters in the computing system. In the example of Figure 1, receiving a request to generate object clusters in the computing system 152 may be carried out by receiving a request to generate object clusters in the computing system 152 through a user interface presented by the object clustering application 208. Alternatively, object clustering application 208 may determine that object clusters should be generated automatically, or by a trigger in computing system 152.

In an initialization stage, a system starts with a large number of clusters to allow enough space for shrinkage, and each object is randomly assigned to one cluster. The algorithm then iteratively optimizes the cluster membership of each object and shrinks superfluous clusters until the system converges or the maximum iteration number is reached. Within each iteration, the optimization potential of each object is computed and ranked according to a function (b). The optimization potential is the difference in function (a) when the object is assigned to the optimal cluster from its current cluster. Objects with the greatest optimization potential are re-assigned to their optimal clusters. Preferably, multiple objects instead of a single object from cluster to cluster each time to accelerate computing. An example algorithm is described in more detail below and with reference to Figure 8. Figure 8 illustrates the algorithms of Shrinkage Clustering that permute memberships of a (Left) single object and (Right) multiple objects per iteration.

Formula 1 below is an example function that may be used as function (a). Function (a) is the objective function for the whole system, where S is the similarity matrix containing quantitative similarity relationship between each pair of objects, A is the cluster membership matrix describing the cluster membership of each object.

Formula 1

$$\min_{A} \left\| S - AA^{T} \right\|_{F} \text{, subject to } A_{ij} \in \{0,1\}, \sum_{j} A_{ij} = 1, \sum_{i} A_{ij} \neq 0, \forall i, j \quad \text{(a)}$$

Formula 2 below is an example function that may be used as function (b). Function (b) calculates the optimization potential of the i^{th} object in comparison to its current cluster assignment, in which M^{ik} calculates the optimization potential of the i^{th} object if placed in the k^{th} cluster.

Formula 2

$$v_i = \min_j(M_{ij} - \sum_j (M \circ A)_{ij}), \text{ wherein } M_{ik} = \sum_{j \in C_k} (1 - 2S_{ij}) \qquad \text{(b)}$$

As described in more detail below with reference to Figure 7, in the beginning of this process, the system contains clusters of relatively similar sizes. After a couple of iterations, some clusters will grow to contain more objects, while some clusters shrink in their sizes and collapse. Whenever a cluster becomes empty, i.e. containing no objects, it will be removed. In the scenario of constrained clustering when a minimum cluster size is required for all resulting clusters, cluster sizes are checked in the beginning of each iteration. Clusters with sizes smaller than the required minimum size will be removed, while objects within these removed clusters will be re-assigned to the rest of clusters based on the optimization potential. In most cases, the algorithm will converge, meaning that the system evolves to the stage when all objects are placed in their optimal clusters and no membership permutations are needed. That is when the iteration process stops to output the final cluster membership for all objects. A maximum iteration number is usually pre- determined as a safety measure before the process begins, so that there is a limit on the total number of iterations in case the system runs into infinite loops.

Another type of clustering algorithm that may be used includes a progeny clustering algorithm. Progeny clustering is an unsupervised machine learning algorithm for solving clustering tasks. The aim of this algorithm is to identify the optimal number of clusters in a population of objects based on their numerical features, including mimicking the similarity relationships, i.e. a minimizing function (a). In some embodiments, one or more object clustering application(s) 208 and one or more data object file(s) 211 may be operable to implement a progeny clustering algorithm as described in more detail below.

Progeny Clustering is based on stability analysis, but employs a sampling technique to reflect cluster identity as well as to reduce computation time. The measure of stability is based on a co-occurrence probability matrix that captures true classification and false classification when new samples are repetitively drawn and clustered. Reference datasets, similar to those used in Gap Statistics, are employed to overcome potential biases inherent in the algorithm and data space.

An example implementation of progeny clustering is described herein. In the algorithm, let $\{x_{ij}\}$, i=1,…,N; and j=1,…,M, be a finite dataset on M features of a digital organism object (e.g., protein expression levels, phenotyping metrics) for N independent observations (e.g., AML patients, cells). This algorithm may also make use of a clustering method (e.g., K-means) that partitioned the data into K clusters, $C_1,...,C_K$. As a cluster analysis groups observations that share similar characteristics and distinguishes those that do not, each cluster C_K, k=1,…,K, would have a distinct characteristic or be compact in space. Therefore, the whole population is assumed to be heterogeneous and can be inherently divided into k more homogeneous subpopulations. Then, each observation in $\{xij^{(K)}\}$ can be viewed as being randomly sampled from a subpopulation corresponding to the cluster it belongs to (C_K).

In contrast to traditional resampling methods that operate on the entire dataset, a progeny clustering algorithm may implement a sampling method to exploit the inherent heterogeneity of the population as well as to reduce the computation costs of the entire analysis. The sampling method may sample values from each feature individually to construct new imaginary samples within each cluster. These imaginary samples may be termed “Progenys” and this process “progeny sampling.” For example, if N* (e.g., 5, 10, 20) is the number of Progenys sampled for each cluster C_K, the validation dataset $\{yij^{(K)}\}$, i=1,…,N*, j=1,…,M, k=1,…,K, would have a total size of K×N* (KN*) observations with M features. To construct each $yij^{(K)}$ from C_K, a sample is randomly drawn from the jth feature in C_K. As the location, the span and the density of each feature space are characteristic of each cluster and somewhat different from that of other clusters, progeny sampling allows assessment

of the distinctness, homogeneity and compactness of each cluster without using the same samples and enables reduction in the sample size for validation.

The new observations sampled from each cluster $\{yij^{(K)}\}$ may be combined into one new dataset $\{yij\}$, which may be clustered using the same method as what is used when clustering the original dataset $\{xij\}$. The clustering assignments will be represented in a (KN*×KN*) co-occurrence matrix Q, with the entries defined as illustrated below in Formula 3.

Formula 3

$$Q(i,j) = \begin{cases} 1, & \text{if observation } i \text{ and } j \text{ are in the same cluster} \\ 0, & \text{otherwise} \end{cases}$$

The co-occurrence matrix is symmetric (i.e., Q(i,j)=Q(j,i). Furthermore, the co-occurrence matrix is arranged in such a way that observations reconstructed from the same original cluster are next to each other (i.e., the $\{y_{((K-1)\times N^*+1)j}\}$ to $\{y_{(K\times N^*)j}\}$ observations are resampled from C_K. The co-occurrence matrix thus can be divided into two regions: k (N*×N*) blocks of "true classification" along the diagonal, and (K-1)×K (N*×N*) blocks of "false classification". If there is absolute agreement between the new and the original clustering assignments, Q would be a perfect block diagonal matrix of K non-overlapping blocks of ones along the diagonal, surrounded by blocks of zeroes.

After repetitively constructing new datasets and performing cluster analysis R times, a series of R co-occurrence matrices may be obtained. Each co-occurrence matrix may be denoted as $Q^{(r)}$, r=1,…,R. To summarize $Q^{(r)}$ from each repetition, a co-occurrence probability matrix P may be defined as illustrated below in Formula 4.

Formula 4

$$P(i,j) = \frac{\sum_r Q^{(r)}(i,j)}{R}$$

The co-occurrence probability matrix P has the same property as $Q^{(r)}$, consisting of K (N*×N*) blocks of "true classification" likelihood along the diagonal and (K-1)×K (N*×N*) blocks of "false classification" likelihood in the rest of the matrix. The

more robust and stable the clustering is, the higher ratio of true classification vs. false classification there will be. A score for clustering stability may then be defined as illustrated below in Formula 5.

Formula 5

$$S = \frac{\sum_k \sum_{i \in C_k} \sum_{j \neq i \in C_k} P(i,j)/(N^*-1)}{\sum_k \sum_{i \in C_k} \sum_{j \notin C_k} P(i,j)/(KN^*-N^*)}$$

To minimize biases inherent in the dataset and algorithm, reference datasets that are randomly sampled from the same data space may serve as the control. For example, the reference dataset $\{xij^{(K)}\}$ can be generated either from a uniform distribution over the range of each feature or from a uniform distribution over a box aligned with the principle components of the data. In the following discussion, the former is used to illustrate certain embodiments of the present disclosure. For example, T reference datasets may be generated by Monte Carlo simulation, and each of them treated with the same steps performed on the original dataset with output denoted as $\{S^{*(K)(t)}\}$, t=1,…,T. The difference in score at each number of clusters when comparing S to S* is illustrated below in Formula 6.

Formula 6

$$D^{(K)} = S^{(K)} - \frac{\sum_t S^{*(K)(t)}}{T}, \text{ where } K=K_{min}, \ldots, K_{max}$$

To choose the optimal number of clusters K_o, either of the two criteria illustrated below in Formula 7 may be followed.

Formula 7

$$K_o = \arg\max D^{(K)}$$

$$K_o = \arg\max\left(D^{(K)} - D^{(K-1)}\right), D^{(K)} > D^{(K-1)}$$

Further details of the example progeny clustering algorithm are described below with reference to Figures 8–9.

Also stored in RAM 168 of computer 152 are one or more digital mapping files 222, a module of computer program instructions for storing digital organisms (e.g., at

digital organism store 436) and certain visualization modules (e.g., at visualization store 432). In some embodiments, the data associated with the digital organism may be stored in RAM 168. For example, the data associated with the digital organism may be stored at digital mapping files 222. As an additional example, the data may be stored at digital organism store 436 within digital mapping files 222. In the same or alternative embodiments, the data may be stored in different portions of RAM 168, other memory internal to example computer 152, external memory communicatively coupled to example computer 152 (e.g., data storage 170), or any other appropriate memory device.

Figure 2 illustrates a flowchart of an illustrative method for presenting a digital organism within a digital map, in accordance with certain embodiments of the present disclosure. The method depicted in Figure 2 can include receiving 205 a plurality of digital organism objects associated with the digital organism. The digital organism objects may be included as part of a design of a digital organism, where the design comprises a set of features defining a biological layer of the organism, apposition on the digital map, a structure, and a set of required visualization modules. Such features may be specified, for example, using a biological layer indicator, a structure indicator, a map position indicator, a visualization module indicator, and any other indicators.

The example method depicted in Figure 2 also includes rendering 210 a portion of the digital organism within the digital map based at least on a first digital organism object that is one of the digital organism objects received 205 above. Rendering 210 a portion of the digital organism within the digital map based at least on a first digital organism object may be carried out, for example, by structuring the required visualization modules using the set of features specified in the indicators described above. For example, a portion of the digital organism may be rendered 210 using a first digital organism object by placing certain visualization modules as indicated by the features of the digital organism object in the digital map.

The example method depicted in Figure 2 also includes placing 215 the rendered portion within the digital map based at least on the map position indicator. Such a map position indicator may specify the position of a rendered object on the digital map. The position of the rendered object may be specified in relative terms such as, for example, an adjacency to other cells, a location within a tissue, and so on.

The example method depicted in Figure 2 also includes enabling 225 a search function of the digital map. Such a search function may be operable to search the plurality of digital organism objects, as described above with reference to Figure 1. The search function can include a relationship search operable to identify a relationship between biologic parts of the digital organism, a keyword search of a plurality of layers associated with the digital organism, a clustering algorithm consisting of a shrinkage clustering algorithm or a progeny clustering algorithm, and so on.

The example method depicted in Figure 2 also includes receiving 230 a user input, the user input associated with a selection of a portion of the rendered portion and rendering 235, based at least on the user input, a second portion of the digital organism based at least on a second digital organism object. For example, a portion of the digital organism at a particular layer may be selected by a user, and in response to a user selecting the portion of the digital organism, method 200 may display a different layer of the digital organism, rendering the different layer and/or receiving additional data as needed.

Although method 200 is illustrated as having 205, 210, 215, 225, 230, and 235 in a particular order, readers of skill in the art will recognize that the various steps of the example method may be carried out in a different order, simultaneously, or may be omitted altogether. For example, a digital map may be generated without enabling a search function.

Figure 3 illustrates and example digital map 300 illustrating hierarchical relationships in an organism, in accordance with certain embodiments of the present disclosure. The hierarchical relationships between tissue, cells and proteins are illustrated by digital maps 301, 302, 303, 304. In the illustrative example provided by map 300, a digital map of brain tissue 304 is illustrated. A user of digital map 300 may zoom in on a portion of brain tissue map 304 in order to retrieve a second digital map of brain tissue cells 301. Digital map 301 may be further zoomed to another layer at digital map 302, and further to digital map 303. Likewise, a user may zoom out of any layer to a higher layer. Patient data associated with each digital map (or portion of a digital map) may be searchable, both by queries, like acute myeloid leukemia patients, and/or by dropping images into the web frame or desktop software program. As an illustrative example, the survival outcome 305 for three subpopulations of patients is shown together with identified phenotypes 306 of the patients' cells and clinical-relevant protein signaling changes that occur in the cells' cytoskeleton.

In some embodiments, these maps may provide teachers and students an interactive platform for studying biological systems. Whereas textbooks or videos can walk students through anatomy, these maps allow the students to walk through a body, tissue or cell themselves and explore relationships visually. Further, by illustrating known relationships and predicting similarities across biological data, a digital map may give researchers rapid, easy access to data in a format that otherwise would be buried in articles. The map also may provide a means to support or refute hypotheses before going into the wet lab. Researchers can compare their current results (e.g., images, protein expression) to prior knowledge simply by dropping in their data or image online. Still further, clinicians may reference a digital map to compare their patients' biopsies and/or omics tests with other patients, whose outcome is known. Furthermore, they can learn more about the biological interactions associated with a particular disease, and be informed of new research on protein signaling when they chose the optimal drug for specific patients.

In some embodiments, a digital map may also aid in phenotyping, or finding spatial patterns in proteins, cell or tissue shapes that are displayed in the map. For example, all tissues that have endothelial cells expressing HIF1 at an average ratio of 1:3 nuclear:cytoplasm or above.

A digital map may add a spatial context to high-dimensional data. As described in more detail below with reference to Figure 4, the digital map may also retain the biological hierarchy (users can zoom into tissues to see cells) and allow users to interact with the data.

Figure 4 illustrates a schematic diagram of example digital DNA 400 for rendering digital maps, in accordance with certain embodiments of the present disclosure. As described in more detail above with reference to Figure 1, example digital DNA 400 may be one means of encoding a set of features defining a digital organism object (and/or a digital organism).

In some embodiments, the digital DNA may include certain values encoded by a series of colors and intensities. In the same or alternative embodiments, the values may be encoded by other means (e.g., letters or numbers). The chosen implementation may be based on design choice made between computing speed and types of features of interest. In example digital DNA 400 the values are encoded using color, intensity, and color sequence. For example, the first two slots (or rows in the color vector illustrated) may identify the object and scale of interest (e.g., tissue - brain, cell - endothelial, protein – apoptosis pathway). The second two slots may identify the coordinates of the object on the map, and the phenotype or types (if known) to which the object belongs. Subsequent slots may further describe features of the object and the ways to render it visually. With digital DNA, a single digital organism may be encoded with a single data structure (e.g., one digital DNA). With this encoding mechanism, component digital organism objects may refer to different portions of the digital DNA defining a single digital organism.

In alternative embodiments, other data encoding means may be used without departing from the scope of the present disclosure. For example, each portion of a digital organism may be associated with a different digital organism object, with each object being encoded. Each object may be encoded differently. Other examples of encoding mechanisms include bar codes, compressed data, and other encoding mechanisms operable to maintain links between layers of a digital organism are maintained.

Figure 5 illustrates example data structures 501, 502 for tissue in a digital organism, in accordance with certain embodiments of the present disclosure. Data structures 501, 502 illustrate the links between various component structures of a digital organism. For example, 501 illustrates an example module describing brain tissue and three subclasses: vascular cells, glia, and neurons. As an additional example, 502 illustrates an example module describing bone marrow tissue and two subclasses: mesenchymal stem cells and leukemic cells. The leukemic cells subclass has a further subclass of protein signaling of leukemic cells. The links between subclasses may be maintained through links within and/or between data structures.

Figure 6 illustrates example hierarchical diagram 600 of biological layers associated with a digital organism, in accordance with certain embodiments of the present disclosure. In some embodiments, hierarchical diagram 600 may include structures 601, 602, 603. For example, structure 601 may be associated with a particular type of tissue (e.g., brain tissue). Properties associated with the layer associated with the structure are stored in the data structure. For example, structure 601 may include clinical associations, cluster memberships (e.g., for clustering algorithms, as described in more detail above with reference to Figure 1), features defining the digital organism object, and adjacency graph information. Structure 601 may have associated therewith one or more agent structures. For example, structure 601 may have associated therewith agent structure 602. Structure 602, in the illustrative example of Figure 6, may be associated with certain types of cells of the tissue described by structure 601. In some embodiments, structure 602 may include

information types similar to those included in structure 601, but associated with the digital organism layer associated with structure 602. In some embodiments, structure 602 may also have an associated agent structure. For example, structure 602 may have associated therewith agent structure 603. Structure 603, in the illustrative example of Figure 6, may be associated with certain types of proteins of the cell described by structure 602. In some embodiments, structure 603 may include information types similar to those included in structures 601, 602.

In some embodiments, the information stored in example structures 601, 602, 603 may include that information necessary to render all or a portion of a digital organism on a digital map. The object-agent relationship of example structures 601, 602, 603, may enable the layers of the digital organism to be searchable independently or in combination.

Figure 7 illustrates a flowchart of an example method 700 for clustering data for searching a digital map, in accordance with certain embodiments of the present disclosure. Method 700 illustrates an example shrinkage clustering algorithm. Method 700 may include receiving a data set, identifying relationships, grouping objects, calculating optimization potential, identifying objects, and reassigning objects. In some embodiments, method 700 may begin at 702, although in alternative embodiments other appropriate entry points may be found in accordance with the present disclosure. For example, method 700 may begin at 710, wherein a optimization potential is recalculated.

Method 700 is also described in more detail above with reference to Figure 1 and below with reference to Figure 8. In some embodiments, method 700 may begin at 702. Once begun, method 700 may proceed to 704, where method 700 may receive a data set comprising a plurality of objects. These "objects" are to be distinguished from the digital organism objects described in more detail above with reference to Figure 1. The objects described herein may refer to all or a portion of the data associated with a particular digital organism object. Once the data set is received,

method 700 may proceed to 706, where relationships between the plurality of data objects are identified. As described in more detail above with reference to Figure 1, the relationships between data objects may, in some embodiments, be stored at one or more object relationship store(s) 442 of computer 152.

After identifying relationships, method 700 may proceed to 708, wherein objects are grouped into a predefined maximum number of clusters using the identified relationships. Once grouped, method 700 may proceed to 710, wherein an optimization potential for each object is calculated. Method 700 may then proceed to 712, wherein the objects with the highest optimization potential are identified and those so identified may be reassigned to other clusters to generate an optimum number of clusters at 714.

In some embodiments, method 700 may then proceed to 716, where method 700 ends. In alternative embodiments, method 700 may return to other steps of method 700. For example, method 700 may proceed in an iterative fashion, returning to step 708, 710, 712, and/or 714.

Figure 8 illustrates an example algorithm for shrinkage clustering, in accordance with certain embodiments of the present disclosure. The example algorithm may include algorithms that permute memberships of single and multiple objects per iteration. For example, algorithm 802 depicts shrinkage clustering that permute memberships of single objects, while algorithm 804 depicts shrinkage clustering that permute memberships of multiple objects per iteration. As described in more detail above with reference to Figure 1, generating object clusters includes inputting a similarity matrix and outputting the object clustering solution. The process first initializes a random cluster membership matrix and then starts minimizing the global optimization potential by permuting object cluster memberships.

In some embodiments, shrinkage clustering may be used to automatically and rapidly group objects into different clusters based on features of each object and their

relationship to one another. For example, in the context of biological data, shrinkage clustering may be used to (1) identify prognostic biomarkers and patient groups from proteomic data; (2) distinguish major cell phenotypes from post-processed cell images; and/or (3) discover gene clusters that underlie deer mice hypoxia adaptation at high altitudes from gene expression data. In addition to direct object clustering problems, shrinkage clustering may also be applied to ensemble clustering, which operates above the level of any single clustering algorithms. In this configuration, shrinkage clustering processes the clustering results from a plurality of clustering algorithms and outputs a final clustering solution based on the structure indicated from consistencies among individual results.

In some embodiments, the algorithm may also be operable to identify an optimal number of clusters, as well as to be tailored to different clustering needs (e.g. constrained clustering of data objects).

Example algorithms 805, 810 may be modified without departing from the scope of the present disclosure. For example, modifications may include setting maximum cluster size, setting minimum cluster number, etc. Other variations may include a similar framework for soft/fuzzy clustering, clustering data containing missing values, utilizing cluster center as an additional criteria for optimization, etc.

Figure 9 illustrates an example algorithm 900 for progeny sampling, in accordance with certain embodiments of the present disclosure, as described in more detail above with reference to Figure 1.

Example algorithm 900 may be programmed and executed using almost any programming languages and any operating systems. In contrast to other clustering algorithms, algorithm 900 may construct artificial samples by independently and randomly picking feature values. Since nearly all statistical and machine learning algorithms require some step of sampling (e.g., shuffling/bootstrapping/extracting samples), algorithm 900 may be applied in a wide range of algorithm and application

fields, including various classification algorithms (e.g. nearest neighbors, linear discriminant analysis, artificial neural network, support vector machine).

Further, algorithm 900 may provide an improvement in computational efficiency. Clustering is usually the most time consuming step in some clustering evaluation methods, the computation time of which increases with an increase in the sample size; this may be a computational constraint when sampling is done at the scale of the entire dataset (e.g., Consensus Clustering) or at least at half of the dataset (e.g., Clest, Model Explorer). However, progeny clustering may be effective even when using a small amount of samples for iterative clustering.

Example algorithm 900 includes taking an input that includes: continuous data, minimum cluster number of interest, maximum cluster number of interest, and number of iterations (which can be set as default). Algorithm 900 may then iterate over steps illustrated in Figures 9-10 at each cluster number between minimum cluster number and maximum cluster number. These iterative steps may include basic clustering of the whole dataset using a cluster number, iterating progeny sampling and basic clustering of the newly sampled data from progeny clustering using the same cluster number, and calculating a score difference after all iterations.

In some embodiments, progeny clustering may be used for any continuous data to identify the number of clusters inherently existing in a population of objects. These objects can be almost anything. In the context of a digital organism, the objects may represent cells, patients, institutions, companies, plants, drinks, cars, etc. With reference to the search functions described in more detail above with reference to Figures 1-5, progeny clustering may be used to identify sub-groups of cancer patients for different therapies or clinical trials based on patient lab tests for researchers/doctors. In other contexts, progeny clustering may be used to identify key groups of start-ups/companies with varying performance based on their portfolio for venture capital investment, identifying groups of customers based on their purchase history/background information for vendors developing their sales strategy.

Figure 10 illustrates algorithm 900 with an example set of data 1001–1007, in accordance with certain embodiments of the present disclosure. The example data set may be followed from example original data set 1001, to example clustered data set 1002, to example prototype sampling 1003, to example prototypes 1004, to example co-occurrence matrix 1005, and to example co-occurrence probability matrix 1006. Algorithm 900, as applied to the example data sets, may also result in example stability score 1007.

Example embodiments of the present disclosure are described largely in the context of a fully functional computer system for generating object groups. Readers of skill in the art will recognize, however, that the present invention also may be embodied in a computer program product disposed upon computer readable storage media for use with any suitable data processing system. Such computer readable storage media may be any storage medium for machine-readable information, including magnetic media, optical media, or other suitable media. Examples of such media include magnetic disks in hard drives or diskettes, compact disks for optical drives, magnetic tape, and others as will occur to those of skill in the art. Persons skilled in the art will immediately recognize that any computer system having suitable programming means will be capable of executing the steps of the method of the invention as embodied in a computer program product. Persons skilled in the art will recognize also that, although some of the example embodiments described in this specification are oriented to software installed and executing on computer hardware, nevertheless, alternative embodiments implemented as firmware or as hardware are well within the scope of the present invention.

The present disclosure may include systems, methods, and/or computer program products. A computer program product may include a computer readable storage medium (or media) having computer readable program instructions thereon for causing a processor to carry out aspects of the present invention. The computer readable storage medium can be a tangible device that can retain and store instructions for use by an instruction execution device. A computer readable storage

medium, as used herein, is not to be construed as being transitory signals *per se*, such as radio waves or other freely propagating electromagnetic waves, electromagnetic waves propagating through a waveguide or other transmission media (e.g., light pulses passing through a fiber-optic cable), or electrical signals transmitted through a wire.

Aspects of the present disclosure may be described herein with reference to flowchart illustrations and/or block diagrams of methods, apparatus (systems), and computer program products according to embodiments of the invention. It will be understood that each block of the flowchart illustrations and/or block diagrams, and combinations of blocks in the flowchart illustrations and/or block diagrams, can be implemented by computer readable program instructions.

It will be understood from the foregoing description that modifications and changes may be made in various embodiments of the present invention without departing from its true spirit. The descriptions in this specification are for purposes of illustration only and are not to be construed in a limiting sense. The scope of the present invention is limited only by the language of the following claims.

CLAIMS

What is claimed is:

1. A method of presenting a digital organism within a digital map, the method comprising:
receiving a plurality of digital organism objects associated with the digital organism, wherein each digital organism object comprises a biological layer indicator and a structure indicator; and
rendering a portion of the digital organism within the digital map based at least on a first digital organism object, the first digital organism object being one of the plurality of digital organism objects.

2. The method of claim 1, wherein the first digital organism object further comprises a map position indicator.

3. The method of claim 2, further comprising placing the rendered portion within the digital map based at least on the map position indicator.

4. The method of claim 1, wherein at least one of the plurality of digital organism objects further comprises a visualization module indicator.

5. The method of claim 1, further comprising enabling a search function of the digital map, wherein the search function is operable to search the plurality of digital organism objects.

6. The method of claim 5, wherein the search function comprises a relationship search operable to identify a relationship between biologic parts of the digital organism.

7. The method of claim 5, wherein the search function comprises a keyword search of a plurality of layers associated with the digital organism.

8. The method of claim 5, wherein the search function comprises a clustering algorithm, the clustering algorithm consisting of a shrinkage clustering algorithm or a progeny clustering algorithm.

9. The method of Claim 1, wherein the plurality of digital organism objects are encoded as a digital DNA object.

10. The method of claim 1, further comprising:

receiving a user input, the user input associated with a selection of a portion of the rendered portion; and

rendering, based at least on the user input, a second portion of the digital organism based at least on a second digital organism object, the second digital organism object being one of the plurality of digital organism objects.

11. An apparatus for presenting a digital organism on a digital map, the apparatus comprising a computer processor, a computer memory operatively coupled to the computer processor, the computer memory having disposed within it computer program instructions that, when executed by the computer processor, cause the apparatus to carry out the steps of:

receiving a plurality of digital organism objects associated with the digital organism, wherein each digital organism object comprises a biological layer indicator, a structure indicator, a map position indicator, and a visualization module indicator;

rendering a portion of the digital organism within the digital map based at least on a first digital organism object, the first digital organism object being one of the plurality of digital organism objects;

placing the rendered portion within the digital map based at least on the map position indicator; and

enabling a search function of the digital map, wherein the search function is operable to search the plurality of digital organism objects.

12. The apparatus of claim 11, wherein the search function comprises a relationship search operable to identify a relationship between biologic parts of the digital organism.

13. The apparatus of claim 11, wherein the search function comprises a keyword search of a plurality of layers associated with the digital organism.

14. The apparatus of claim 11, wherein the search function comprises a clustering algorithm, the clustering algorithm selected from the group consisting of a shrinkage clustering algorithm and a progeny clustering algorithm.

15. The apparatus of claim 11, wherein the plurality of digital organism objects are encoded as a digital DNA object.

16. The apparatus of claim 11, wherein the computer instructions, when executed by the computer processor, further cause the apparatus to carry out the steps of:

receiving a user input, the user input associated with a selection of a portion of the rendered portion; and

based at least on the user input, rendering a second portion of the digital organism based at least on a second digital organism object, the second digital organism object being one of the plurality of digital organism objects.

17. A computer program product for presenting a digital organism on a digital map, the computer program product disposed upon a computer readable medium, the computer program product comprising computer program instructions that, when executed, cause a computer to carry out the steps of:

receiving a plurality of digital organism objects associated with the digital organism, wherein each digital organism object comprises a biological layer indicator; a structure indicator; a map position indicator; and a visualization module indicator;

rendering a portion of the digital organism within the digital map based at least on a first digital organism object, the first digital organism object being one of the plurality of digital organism objects;

placing the rendered portion within the digital map based at least on the map position indicator;

enabling a search function of the digital map, wherein the search function is operable to search the plurality of digital organism objects;

receiving a user input, the user input associated with a selection of a portion of the rendered portion; and

based at least on the user input, rendering a second portion of the digital organism based at least on a second digital organism object, the second digital organism object being one of the plurality of digital organism objects.

18. The computer program product of claim 17, wherein the search function comprises a search associated with a relationship between biologic parts of the digital organism.

19. The computer program product of claim 17, wherein the search function comprises a keyword search across a plurality of layers associated with the digital organism.

20. The computer program product of claim 17, wherein the search function comprises a clustering algorithm, the clustering algorithm selected from the group consisting of a shrinkage clustering algorithm and a progeny clustering algorithm.

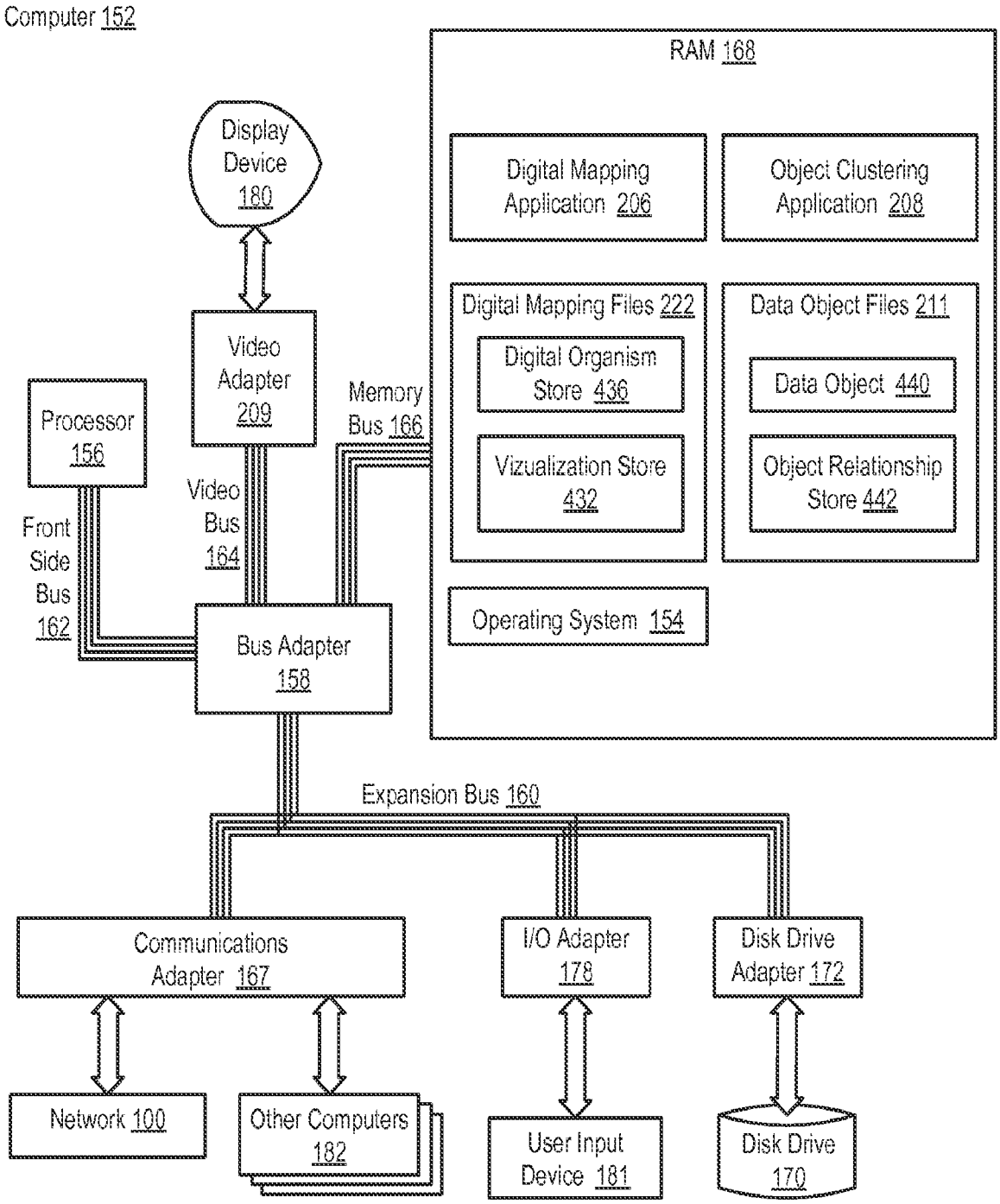

Computer 152
RAM 168
Display Device 180
Digital Mapping Application 206
Object Clustering Application 208
Digital Mapping Files 222
Data Object Files 211
Digital Organism Store 436
Data Object 440
Vizualization Store 432
Object Relationship Store 442
Operating System 154
Video Adapter 209
Memory Bus 166
Processor 156
Video Bus 164
Front Side Bus 162
Bus Adapter 158
Expansion Bus 160
Communications Adapter 167
I/O Adapter 178
Disk Drive Adapter 172
Network 100
Other Computers 182
User Input Device 181
Disk Drive 170

FIG. 1

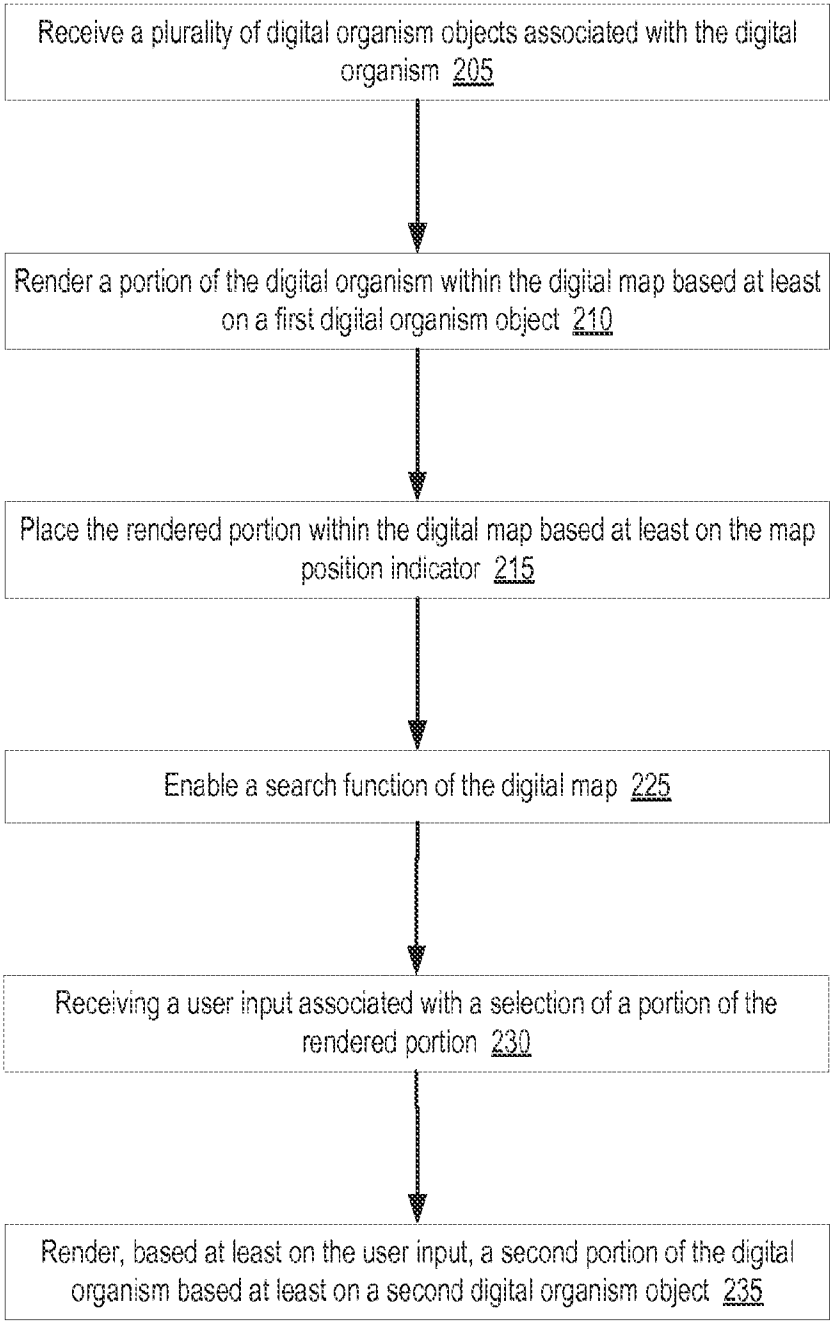

Receive a plurality of digital organism objects associated with the digital organism 205
Render a portion of the digital organism within the digital map based at least on a first digital organism object 210
Place the rendered portion within the digital map based at least on the map position indicator 215
Enable a search function of the digital map 225
Receiving a user input associated with a selection of a portion of the rendered portion 230
Render, based at least on the user input, a second portion of the digital organism based at least on a second digital organism object 235

FIG. 2

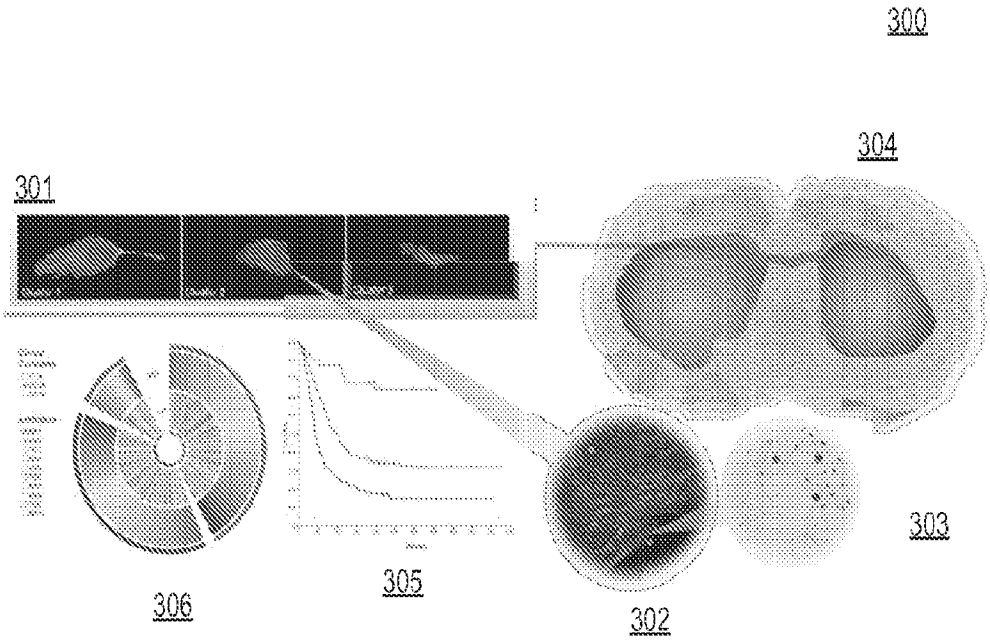


FIG. 3

400

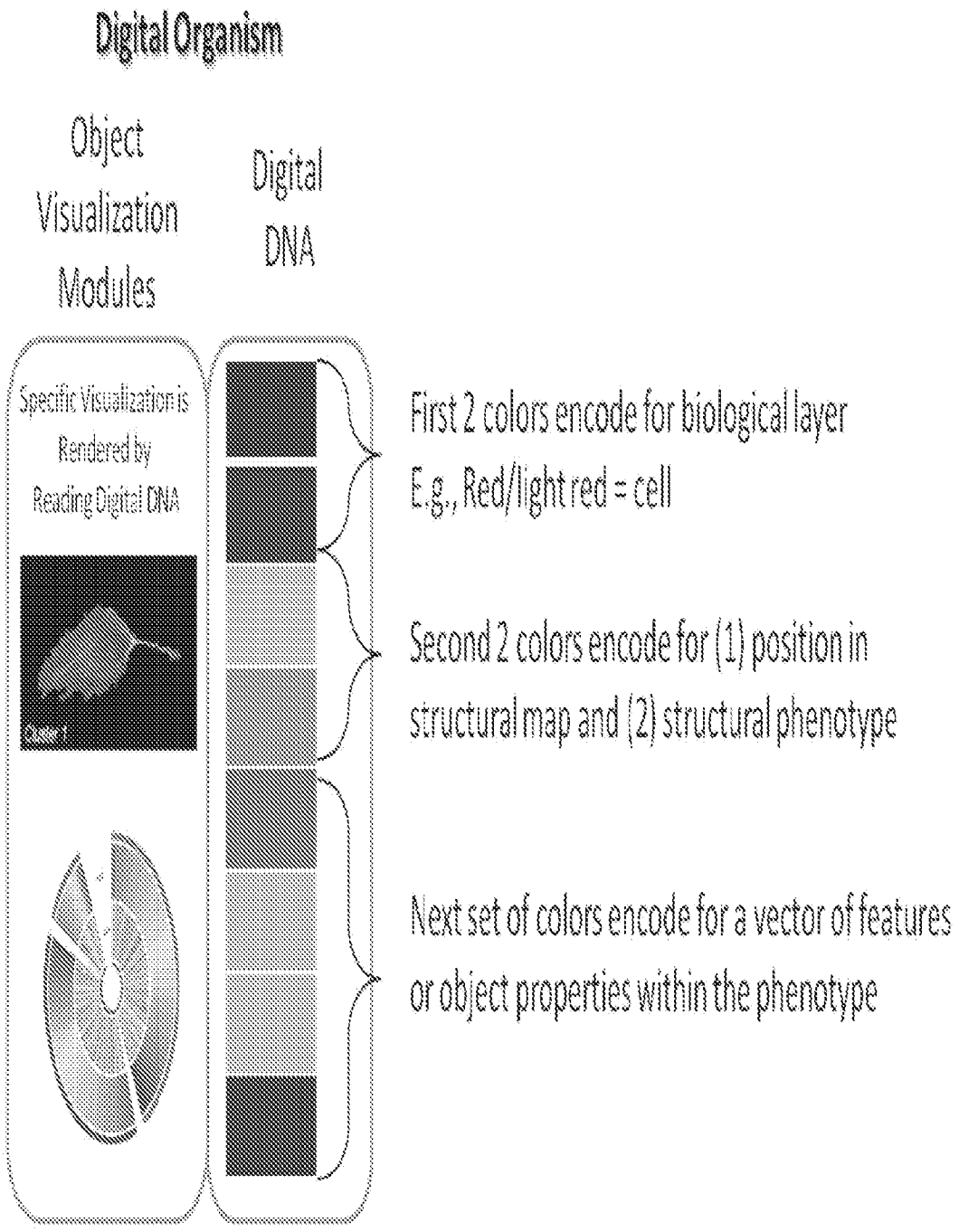
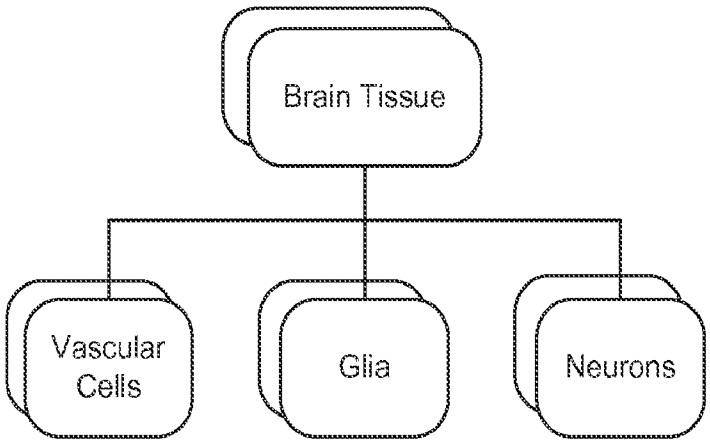

Digital Organism
Object Visualization Modules
Digital DNA
Specific Visualization is Rendered by Reading Digital DNA
First 2 colors encode for biological layer
E.g., Red/light red = cell
Second 2 colors encode for (1) position in structural map and (2) structural phenotype
Next set of colors encode for a vector of features or object properties within the phenotype

FIG. 4

501


Brain Tissue
Vascular Cells
Glia
Neurons

502

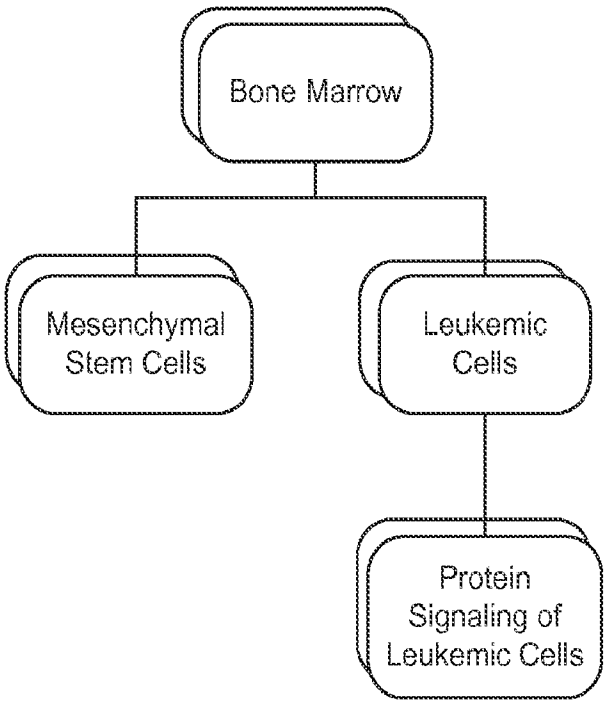

Bone Marrow
Mesenchymal Stem Cells
Leukemic Cells
Protein Signaling of Leukemic Cells

FIG. 5

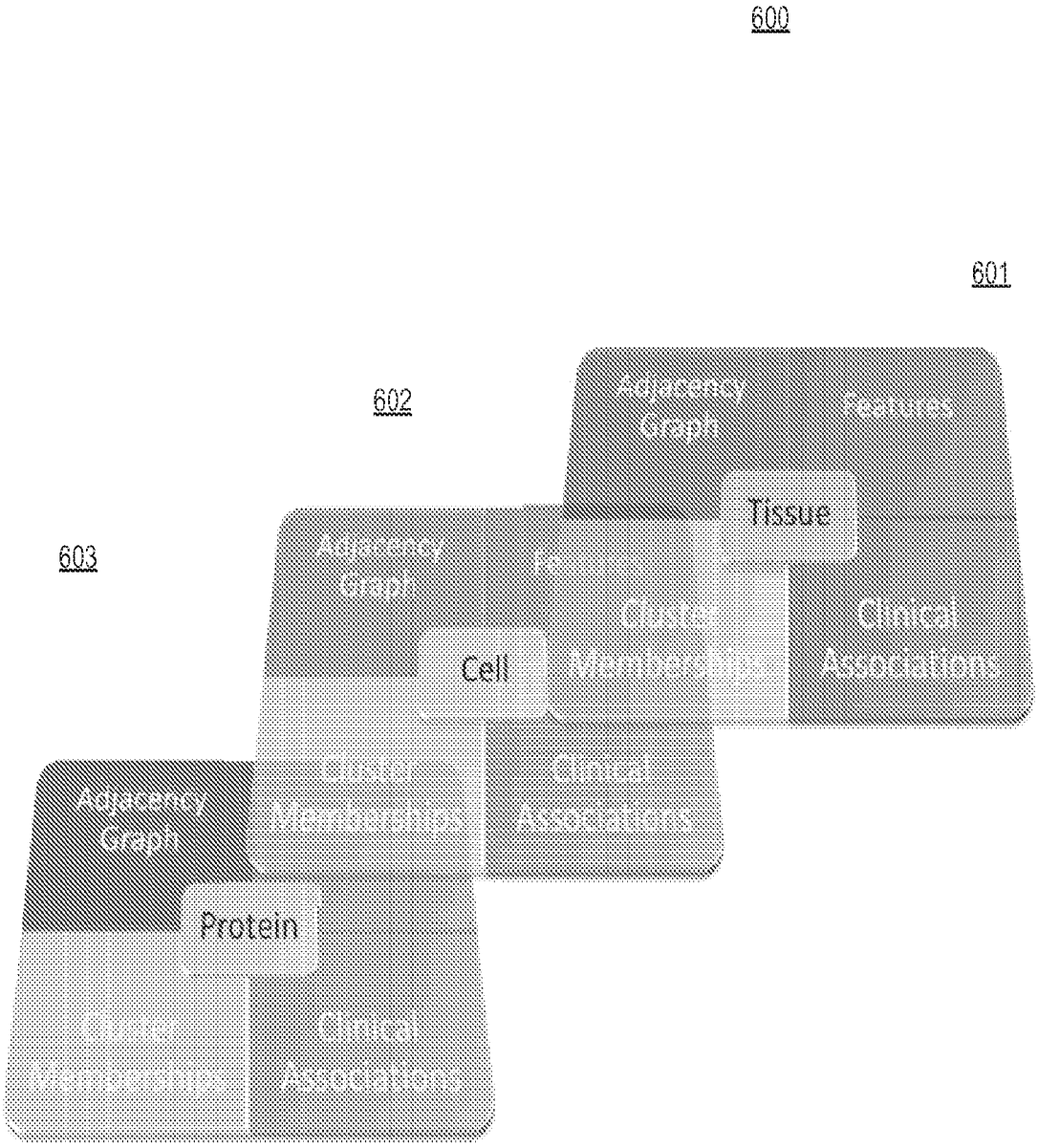

600
601
602
603
Adjacency Graph
Features
Tissue
Cell
Clinical Associations
Protein

FIG. 6

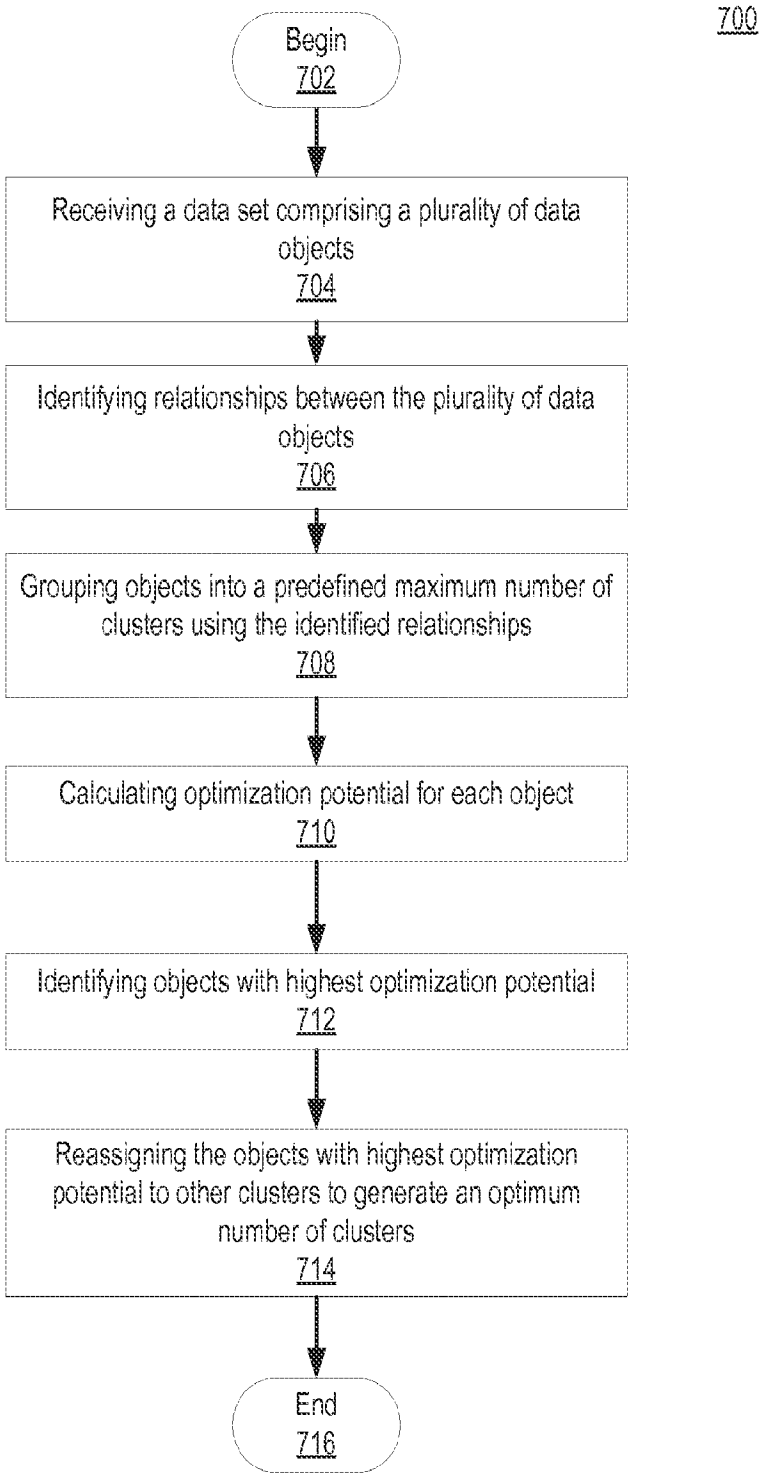

700
Begin
702
Receiving a data set comprising a plurality of data objects
704
Identifying relationships between the plurality of data objects
706
Grouping objects into a predefined maximum number of clusters using the identified relationships
708
Calculating optimization potential for each object
710
Identifying objects with highest optimization potential
712
Reassigning the objects with highest optimization potential to other clusters to generate an optimum number of clusters
714
End
716

FIG. 7

802

Input: $S_{N\times N}$ (similarity matrix)
T (maximum number of iterations)
K_0 (initial cluster number)

Initialization:

a. Generate a random $A_{N\times K_0}$

b. $\tilde{S} = 1 - 2S$

c. $t = 1$

repeat

1. Remove empty columns in A,
i.e. $\{j | \sum_{i=1}^{N} A_{ij} = 0\}$.
2. $M = \tilde{S}A$
3. Compute v_i, where
$v_i = \min_j M_{ij} - \sum_{j=1}^{K} (M \circ A)_{ij}$
4. Permute the membership of $\tilde{X}$ to C', where
$\tilde{X} = \arg\min_i v_i$, $C' = \arg\min_j M_{\tilde{X}j}$
5. $t = t + 1$

until $\sum_{i=1}^{N} v_i = 0$ **or** $t > T$

Output: A (cluster assignment)

804

Additional Input: P_0 (initial number of permutations).

Additional Initialization: $p = P_0$

repeat

1. Remove empty columns in A,
i.e. $\{j | \sum_{i=1}^{N} A_{ij} = 0\}$.
2. $M = \tilde{S}A$
3. Compute v_i, where
$v_i = \min_j M_{ij} - \sum_{j=1}^{K} (M \circ A)_{ij}$
4. Compute R, i.e. the size of $\{i | v_i < 0\}$

if $p \geq R$ **then**
  if $R \geq 10$ **then**
    $p = R/2$
  else
    $p = 1$
  end if
end if

5. Permute the memberships of the top p objects with greatest minimization potential
6. $t = t + 1$

until $\sum_{i=1}^{N} v_i = 0$ **or** $t > T$

FIG. 8

900

Input $\{x_{ij}\}$, K_{min}, K_{max}, $\tilde{N}$, R
for $K = K_{min}$ to K_{max} do
  Cluster $\{x_{ij}\}$ into K clusters
  Initiation Q = Zero Matrix of size $N \times N$
  for $r = 1$ to R do
    for $k = 1$ to K do
      Construct $\{y_{ij}^{(k)}\}$ of size $\tilde{N}$ by *Prototype Sampling* from $\{x_{ij}^{(k)}\}$
    end for
    Cluster $\{y_{ij}\}$ into Formula 5
    $Q = Q + Q^{(r)}$
  end for
  $P^{(K)} = Q/R$
  Compute $\{S^{(K)}\}$ by Equation 3
end for
Output $\{S^{(K)}\}$

FIG. 9

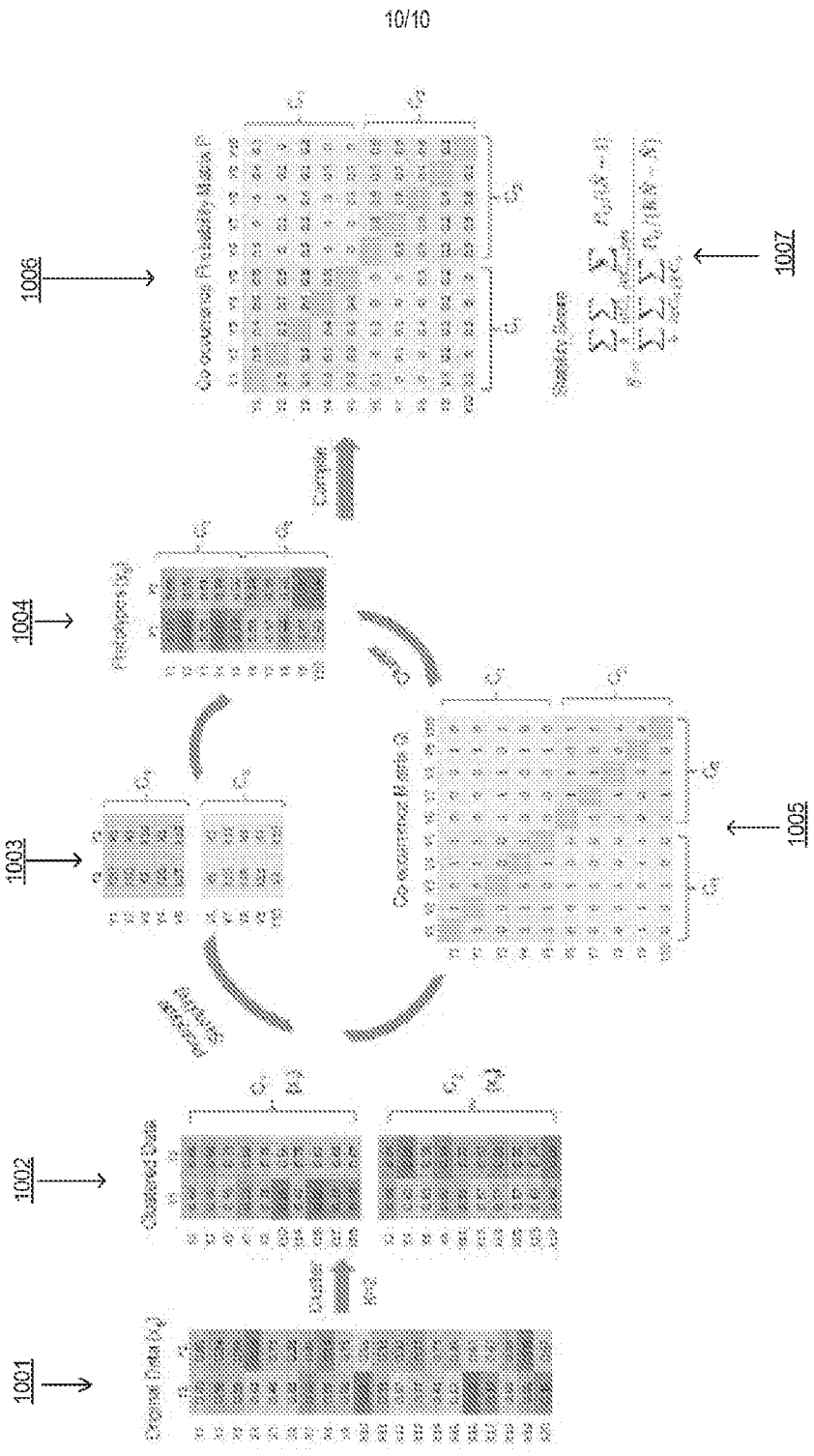


FIG. 10

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2015/032402

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F19/26 G06F19/00
ADD. G06F17/30

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)
G06N G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

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

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JISHANG WEI ET AL: "Visual cluster exploration of web clickstream data", VISUAL ANALYTICS SCIENCE AND TECHNOLOGY (VAST), 2012 IEEE CONFERENCE ON, IEEE, 14 October 2012 (2012-10-14), pages 3-12, XP032308428, DOI: 10.1109/VAST.2012.6400494 ISBN: 978-1-4673-4752-5 page 3 - page 11, right-hand column, paragraph 3	1-20

-/--

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

* 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	Date of mailing of the international search report
28 January 2016	05/02/2016

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Volkmer, Markus

1

Form PCT/ISA/210 (second sheet) (April 2005)

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	LOÏC LECERF ET AL: "Visalix: A Web Application for Visual Data Analysis and Clustering", ACM KDD 2009 (INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING), PARIS, FRANCE, JUNE 28-31, 2009, 28 June 2009 (2009-06-28), pages 1-4, XP055244395,	1-4
A	page 1 - page 4, right-hand column, paragraph 2	5-8,10

A	J. AN ET AL: "J-Circos: an interactive Circos plotter", BIOINFORMATICS., vol. 31, no. 9, 24 December 2014 (2014-12-24), pages 1463-1465, XP055244422, GB ISSN: 1367-4803, DOI: 10.1093/bioinformatics/btu842 page 1463 - page 1465, left-hand column, paragraph 1	1-20

A	M. KRZYWINSKI ET AL: "Circos: An information aesthetic for comparative genomics", GENOME RESEARCH, vol. 19, no. 9, 18 June 2009 (2009-06-18), pages 1639-1645, XP055244408, ISSN: 1088-9051, DOI: 10.1101/gr.092759.109 page 1 - page 21	1-20

A	EREL LEVINE ET AL: "Resampling Method For Unsupervised Estimation Of Cluster Validity", ARXIV.ORG, CORNELL UNIVERSITY LIBRARY, 201 OLIN LIBRARY CORNELL UNIVERSITY ITHACA, NY 14853, 18 May 2000 (2000-05-18), XP080020426, the whole document	1-20

T	CHENYUE W. HU ET AL: "Progeny Clustering: A Method to Identify Biological Phenotypes", SCIENTIFIC REPORTS, vol. 5, 12 August 2015 (2015-08-12), page 12894, XP055244377, DOI: 10.1038/srep12894 the whole document	

1