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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
08 March 2018 (08.03.2018)

W!P O | PCT

(10) International Publication Number
WO 2018/044649 Al

(51) International Patent Classification:
H03M 7/46 (2006.01) *G06F 21/62* (2013.01)
G06F 21/31 (2013.01)

(21) International Application Number:
PCT/US20 17/048207

(22) International Filing Date:
23 August 2017 (23.08.2017)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
15/249,354 27 August 2016 (27.08.2016) US

(71) Applicant: **SECURED2 CORPORATION** [US/US]; 6160 Summit Drive North, Suite 360, Minneapolis, MN 55430 (US).

(72) Inventors: **KLUM, R., Daren;** c/o Secured2 Corporation, 6160 Summit Drive North, Suite 360, Minneapolis, MN 55430 (US). **NEAFUS, Thomas;** c/o Secured2 Corporation, 6160 Summit Drive North, Suite 360, Minneapolis, MN 55430 (US). **KLUGE, Andrew, Jeffrey;** c/o Secured2 Corporation, 6160 Summit Drive North, Suite 360, Minneapolis, MN 55430 (US). **KLUGE, William, Paul;** c/o Secured2 Corporation, 6160 Summit Drive North, Suite 360, Minneapolis, MN 55430 (US).

(74) Agent: **LARSON, James, A.;** Hamre, Schumann, Mueller & Larson, P.C., 45 South Seventh Street, Suite 2700, Minneapolis, MN 55402 (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, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN,

(54) Title: MULTI-DIMENSIONAL RUN-LENGTH ENCODING

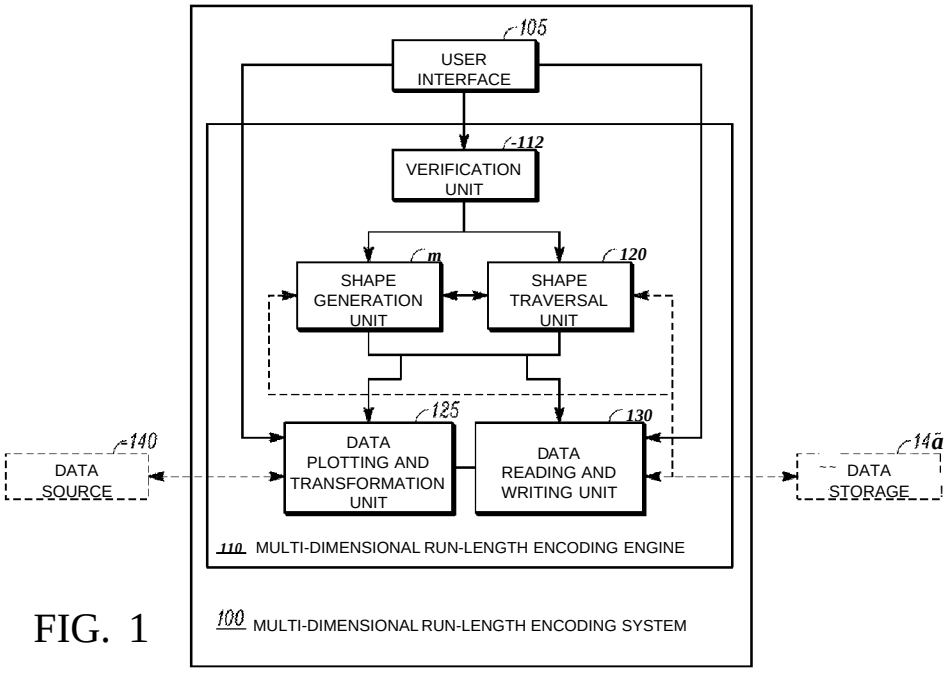

FIG. 1

(57) Abstract: Methods and systems for multi-dimensional run-length encoding of data are provided. In one embodiment a method for multi-dimensional run-length encoding of an unprocessed data file is provided. The method includes obtaining an admission key and determining a traversal path within a virtual multi-dimensional shape based on the admission key. The method also includes transforming unprocessed data of the unprocessed data file into a plurality of compressed data segments. Also, the method includes plotting the plurality of compressed data segments onto a plurality of data points along the traversal path to obtain a plurality of secured data segments. Further, the method includes generically sorting the plurality of secured data segments to obtain a plurality of generically sorted data segments, and writing the plurality of generically sorted data segments into a processed data file.

WO 2018/044649 A1

[Continued on next page]

HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, 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))*

MULTI-DIMENSIONAL RUN-LENGTH ENCODING

FIELD

Embodiments of this disclosure relate generally to data compression and security. More specifically, the embodiments relate to a method and system for multi-dimensional run-length encoding of data.

BACKGROUND

As the size of data increases the market is demanding new ways to reduce data size while also securing it. Various methods of compression save companies money by reducing the amount of data stored. This can reduce the costs of hardware purchases and maintenance to maintain this data. However, as data needs expand, companies require access to more data centers or must upgrade the thousands of machines they currently own. On top of rising costs for data storage, companies are plagued with the constant worry of an unauthorized third party illegally gaining access to their secure computer networks which can cost time, resources, customers and ultimately profits.

SUMMARY

Methods and systems for multi-dimensional run-length encoding of data are described.

The embodiments described herein can provide benefits over existing computer technology in that data can be compressed to a smaller size than current compression techniques and also provide enhanced security over the data. The embodiments described herein can also compress data faster than existing compression techniques near the same data compression ratio (i.e., the ratio between an uncompressed data size and a compressed data size). The embodiments described herein can achieve these benefits even when larger sets of data (e.g., petabyte or larger sized data) are to be compressed. Also, the embodiments described herein can be applied to all forms of data files (e.g., document, picture, directory, live/stored audio, live/stored video, user

log login credentials, system commands, etc.). Thus, the embodiments described herein can red reduce data storage requirements for storing data and can provide faster compression tim times than existing compression techniques.

In s In some embodiments, data can be compressed using the methods and systems des described herein into a format with a ~90% data compression ratio or better. That is, the the data can undergo multi-dimensional run-length encoding and the resulting data will tak take up, at a maximum, ½10% of the storage space that the original data requires.

The The embodiments described herein can provide compression of a data set using one one or more of run-length encoding, data deduplication, and data transformation.

The The embodiments described herein can also secure the compressed data by plo plotting the compressed data along a traversal path within a virtual multi-dimensional sha shape whereby the traversal path and the virtual multi-dimensional shape are uniquely tied tied to a particular admission key.

Th Thus, the embodiments described herein can provide a secure data storage and con compression solution that adds security., shrinks file-sizes, and can scale with user req requirements.

In c In one embodiment, a method for multi-dimensional run-length encoding of an unp unprocessed data file is provided. The method includes obtaining an admission key and det determining a traversal path within a viitual multi-dimensional shape based on the adm admission key. The method also includes tiansforming unprocessed data of the unp unprocessed data file into a plurality of compressed data segments. Also, the method inc includes plotting the plurality of compressed data segments onto a plurality of data poi points along the traversal path to obtain a plurality of secured data segments. Further, the the method includes genetically sorting the plurality of secured data segments to obtain a pla plurality of generically sorted data segments, and writing the plurality of generically sort sorted data segments into a processed data file.

In a In another embodiment, a method for restoring a data file that has been pro processed using multi-dimensional run-length encoding is provided. The method

inc includes obtaining an admission key and reading a plurality of generically sorted data seg segments that forms a processed data file. The method also includes determining a trav traversal path within a virtual multi-dinoiensional shape based on the admission key. Als Also, the method includes plotting the plurality of generically sorted data segments onto a pla plurality of data points along the traversal path to obtain a plurality of compressed dat data segments. Further, the method includes reassembling the plurality of compressed dat data segments into unprocessed data, and writing the unprocessed data into the data file.

In y In yet another embodiment, a multi-dimensional run-length encoding system is pro provided. The system includes a user interface and a multi-dimensional run-length enc encoding engine. The user interface receives an admission key from a user. The multi-dim dimensional run-length encoding engine includes a verification unit, a shape generation unit unit, a shape traversal unit, a data plotting and transformation unit, and a data reading and and writing unit. The verification unit verifies the admission key. The shape generation unit unit determines a virtual multi-dimensional shape equation that is unique to the adm admission key. The shape traversal unit determines a traversal path within a virtual mu multi-dimensional shape defined by the virtual multi-dimensional shape equation and that that is unique to the admission key. The data plotting and transformation unit tran transforms unprocessed data into a plurality of compressed data segments and plots eac each of the plurality of compressed data segments onto a separate data point among a plu plurality of data points along the traversal path to form a plurality of secured data seg segments. The data reading and writing unit generically sorts the plurality of secured dat data segments to form a plurality of generically sorted data segments and writes the plu plurality of generically sorted data segments into a processed data file.

In y In yet another embodiment, a multi-dimensional run-length encoding system is pro provided. The system includes a user interface and a multi-dimensional run-length enc encoding engine. The user interface receives an admission key from a user. The multi-dim dimensional run-length encoding engine includes a verification unit, a shape generation unit unit, a shape traversal unit, a data plotting and transformation unit, and a data reading

and writing unit. The verification unit verifies the admission key. The shape generation unit determines a virtual multi-dimensional shape equation that is unique to the admission key. The shape traversal unit determines a traversal path within a virtual multi-dimensional shape defined by the virtual multi-dimensional shape equation and that is unique to the admission key. The data reading and writing unit reads a plurality of generically sorted data segments from a processed data file. The data plotting and transformation unit plots each of the plurality of generically sorted data segments onto a separate data point among a plurality of data points along the traversal path to obtain a plurality of compressed data segments, and reassembles the plurality of compressed data segments to obtain unprocessed data that forms an unprocessed data file.

DRAWINGS

Fig. 1 illustrates a multi-dimensional run-length encoding system 100 according to one embodiment.

Fig. 2 illustrates a flow chart of a method for multi-dimensional run-length encoding of an unprocessed data file using the system, according to one embodiment.

Fig. 3 illustrates a flow chart of a method for restoring an unprocessed data file that has been processed using multi-dimensional run-length encoding using the system 100, according to one embodiment.

Fig. 4 illustrates a cuboid using a virtual multi-dimensional shape equation, according to one embodiment.

Figs. 5A illustrates a traversal path with a point that extends out of a virtual multi-dimensional shape, according to one embodiment.

Figs. 5B illustrates a traversal path within a virtual multi-dimensional shape, according to one embodiment.

Fig. 6 schematically depicts an architecture of a computing device and computing system optionally used in connection with computer-implemented systems and methods described in this document

DEDF.TATTFDDESCRIPTION

The The following description describes methods and systems for multi-dimensional run run-length encoding of data.

The The embodiments described herein can provide benefits over existing computer tec technology in that data can be compressed beyond current compression techniques and alsca also provide enhanced security over the data. The embodiments described herein can alscalso compress datat faster than existing compression techniques near the same data concompression ratio. The embodiments described herein can also scale with larger sets of dat data (e.g., petabyte or larger sized data) is Also, the embodiments described herein can be be applied to all forms of data files (e.g., document, picture, directory, live/stored audio, livelive/stored video, user login credentials, system commands, etc.). Thus, the emembodiments described herein can reduce data storage requirements for storing data and can can provide faster compression times than existing compression techniques.

In sIn some embodiments, data can be compressed using the methods and systems des described herein into a format with a -90% data compression ratio or better. That is, the the data can undergo multi-dimensional run-length encoding and the resulting data will take take up, at a maximum, 10% of the storage space that the original data requires.

The The embodiments described herein can provide compression of a data set using one one or more of run-length encoding, data deduplication, and data transformation.

Mu Multi-dimensional run-length encoding, as defined herein, refers to compressing dat data using one or more of run-length encoding, data deduplication, and data tran transformation.

Ru Run-length encoding, as defined herein, refers to a form of lossy or lossless data con compression in which sequences of identical data are stored as a data value that has a stor storage size smaller than the storage size of a single occurrence of the sequence of ide identical data.

DatData deduplication, as defined herein, refers to a form of compression in which sec second or later occurrences of a sequence of identical data is replaced with a reference poi pointer that points to a location of the data value representing the sequence of identical dat data. sIn some embodiments, the reference pointer can be coordinates of a data point alo along a traversal path where a data value is located.

DatData transformation, as defined herein, refers to a form of transformation in whi which a mathematical operation is operated on data to transform the data.

TheThe embodiments described herein can also secure the compressed data by plo plotting the compressed data along a traversal path within a virtual multi-dimensional sha shape whereby the traversal path and the virtual multi-dimensional shape are uniquely tied tied to a particular admission key.

A dA data transformation as defined herein refers to a mathematical operation app applied on a data set that converts the *dat a* set into a different data set.

ProProcessed data, as defined herein refers, to data that has undergone multi-dim dimensional run-length encoding as described in the embodiments provided herein. SimSimilarly, a processed file or processed data file, as defined herein, refers to a data file that that has undergone dimensional multi-nin-length encoding as described in the em embodiments provided herein.

UnUnprocessed data, as defined herein refers, to data that is not processed using mu multi-dimensional run-length encoding as described in the embodiments provided her herein. Similarly, an unprocessed file or unprocessed data file, as defined herein, refers to a to a data file that is not processed using multi-dimensional run-length encoding as des described in the embodiments provided herein.

A vA virtual multi-dimensional *shape,* as defined herein, refers to a shape defined by by one or more mathematical equations mat can be used in multi-dimensional run-len length encoding for providing a boundary for a traversal path along which compressed dat data and/or reference pointer(s) are located.

Ge Genetically sorting data or sorting data in a generic format, as defined herein, refe refers to sorting data in a format that is irrespective of the content of the data and/or the seq sequence of the data when unprocessed. In some embodiments, genetically sorting data can can include sorting data based on a coordinate system used for plotting compressed data on onto a traversal path. In these embodiments, the generically sorting data can include sort sorting data in, for example, an ascending coordinate order, a descending coordinate ord order, etc.

Th Thus, the embodiments described herein can provide a secure data storage and con compression solution that adds security, shrinks file-sizes, and can scale with user req requirements.

Fig Fig. 1 illustrates a multi-dimensional run-length encoding system 100 according to one embodiment. The multi-dimensional run-length encoding system 100 allows dat data (hereinafter referred to as unprocessed data) to undergo multi-dimensional run-len length encoding that compresses and secures the unprocessed data (hereinafter referred to as processed data). The multi-dimensional run-length encoding system 100 includes a use user interface 105 and a multi-dimensional run-length encoding engine 110. The mu multi-dimensional run-length encoding system 100 can be connected to a data source 140 140 to retrieve unprocessed data for multi-dimensional run-length encoding, and can be con connected to a data storage 145 for storing processed data transformed by multi-dim dimensional run-length encoding.

The The user interface 105 is connected to the multi-dimensional run-length enc encoding engine 110. The user interface 105 allows a user to provide an admission key (e.g (e.g., password) and allows a user to modify operation settings of the multi-dimensional run run-length encoding engine 110. The user interface 105 can also allow a user to provide unp unprocessed data to the system 100. Examples of the user interface 105 can include, but but are not limited to, a touch screen, a key pad, a voice command recognition system, a fing fingerprint recognition system, any other type of biometric recognition system, etc.

The multi-dimensional run-length encoding engine 110 includes a verification unit 112, a shape generation unit 115, a shape traversal unit 120, a data plotting and transformation unit 125 and a data reading and writing unit 130.

The verification unit 112 is connected to the user interface 105, the shape generation unit 115 and the shape traversal unit 120. The verification unit 112 receives an admission key from the user interface 105. The verification unit 112 determines whether the admission key is recognized and sends instructions to the shape generation unit 115 and the shape traversal unit 120 that can vary based on whether the admission key is recognized or not recognized.

The shape generation unit 115 is connected to the verification unit 112, the shape traversal unit 120, the data plotting and transformation unit 125 and the data reading and writing unit 130. The shape generation unit 115 can also be connected to the data storage 145.

When the admission key is not recognized by the verification unit 112, the shape generation unit 115 generates a virtual multi-dimensional shape equation and any scalers associated with the virtual multi-dimensional shape equation that is tied to the admission key. The shape generation unit 115 sends the generated virtual multi-dimensional shape equation (with any scalers) to the shape traversal unit 120 and/or to the data storage 145 for storage.

When the admission key is recognized by the verification unit 112, the shape generation unit 115 obtains a previously generated virtual multi-dimensional shape equation (with any scalers) that is uniquely associated with the admission key. In some embodiments, the previously generated virtual multi-dimensional shape equation (with any scalers) can be obtained from the data storage 145.

The shape traversal unit 120 is connected to the verification unit 112, the shape generation unit 115, the data plotting and transformation unit 125 and the data reading and writing unit 130. The shape traversal unit 120 can also be connected to the data storage 145.

WhWhen the admission key is not recognized by the verification unit 112, the shape travtraversal unit 120 receives the generated virtual multi-dimensional shape equation (with any scalers) from the shape generation unit 115 for from the data storage 145. The shape travtraversal unit 120 uses the generated virtual multi-dimensional shape equation (with any scalscalers) to generate a traversal path that is uniquely associated to the admission key. TheThe shape traversal unit 120 can send the generated traversal path to the data storage 145 145 for storage.

WhWhen the admission key is recojmized by the verification unit 112, the shape travtraversal unit 120 obtains a previously generated traversal path that is uniquely associated with the admission key. sIn some embodiments, the previously generated travtraversal path can be obtained from the data storage 145.

TheThe data plotting and transformation unit 125 is connected to the user interface 105105, the shape generation unit 115; the shape traversal unit 120 and the data reading and writwriting unit 130. The data plotting and transformation unit 125 can also be connected to tto the data source 140. The data plotting and transformation unit 125 receives the virtvirtual multi-dimensional shape equation (with any scalers) from the shape generation uniunit 115 and receives the traversal path from the shape traversal unit 120.

WhWhen an unprocessed data file is to undergo multi-dimensional run-length encencoding, the data plotting and transformation unit 125 obtains the unprocessed data file (e.g(e.g., from the user interface 105, the data source 140, etc.) and compresses the unpunprocessed data in the unprocessed data file using one or more of run-length encoding, datdata deduplication, and data transformaition to obtain compressed data file segments. TheThe data plotting and transformation unit 125 then secures the compressed data segsegments by using the virtual multi-dimensional shape equation (with any scalers) and the the traversal path to plot the compressed data segments along the traversal path of the virtvirtual multi-dimensional shape defined by the virtual multi-dimensional shape equation to oto obtain secured data segments.

WhWhen a processed file is to be restored to its original form (e.g., an unprocessed dat data file), the data plotting and transfonnation unit 125 uses the virtual multi-dimdimensional shape equation (with any scalers) and the traversal path uniquely ass associated to the admission key in order to arrange generically sorted data segments that form form the processed file in the proper order to obtain the compressed data segments. ThaThat is, the processed file includes the processed data arranged in a generic format (e.g., asc ascending coordinate order, descending coordinate order, etc.) and the data plotting and tran transformation unit 125 arranges the generically sorted data segments based on the virtvirtual multi-dimensional shape equation (with any scalers) and the traversal path to obt obtain the compressed data segments.

ThThe data plotting and transformation unit 125 then decompresses and reareassembles the compressed data segments that have undergone one or more of run-len length encoding, data deduplication, and data transformation to obtain unprocessed dat data. ThThe data plotting and transformation unit 125 then forms an unprocessed data file usiusing the unprocessed data. ThThe unprocessed data file can then be provided to the user.

ThThe data reading and writing unit 130 is connected to shape generation unit 115, the the shape traversal unit 120, and the data plotting unit 125. ThThe data reading and wri writing unit 130 can also be connected to the data storage 145.

WhWhen an unprocessed data file is to undergo multi-dimensional run-length enc encoding, the data reading and writing unit 130 obtains the secured data segments from the the data plotting and transformation unit 125, sorts the secured data segments into a gen generic format (e.g., ascending coordinate order, descending coordinate order, etc.) to obt obtain generically sorted data segments, and then writes the generically sorted data seg segments to a file to obtain the processed data file. For example, even though the trav traversal path starts at (0,0,0) then goes to (1,0,0) and then (0,1,0), the data reading and and writing unit 130 can generically sort the secured data in ascending order and start at (0, (0,0,0) then (0,1,0) and then (1,0, 0).

The data reading and writing unit 130 can then provide the processed data file to, for example, the user interface 105 and/or the data storage 145 for storage.

When a processed file is to be restored to its original form (e.g., an unprocessed data file), the data reading and writing unit 130 obtains the processed data file (e.g., from the user interface 105, the data storage 145, etc.), reads the generically sorted data from the processed data file, and provides the generically sorted data to the data plotting and transformation unit 125.

A method for providing multi-dimensional run-length encoding of an unprocessed data file is discussed below with respect to Fig. 2. A method for restoring a processed data file is discussed below with respect to Fig. 3

Fig. 2 illustrates a flow chart of a method 200 for multi-dimensional run-length encoding of an unprocessed data file using the system 100 to obtain a processed data file, according to one embodiment.

The method 200 begins at 205 whereby the user interface 105 waits to receive an admission key from a user. The admission key can be any type of password, biometric authentication, etc. used to authenticate the user. When the user interface 105 receives an admission key, the user interface 105 relays the admission key to the verification unit 112 and the method 200 proceeds to 210.

At 210, the verification unit 112 determines whether the admission key is recognized. If the admission key is recognized, the method 200 proceeds to 215 and 220. In some embodiments, the method 200 can proceed to 215 and 220 concurrently. If the admission key is not recognized, the method 200 proceeds to 225.

At 215, the shape generation unit 115 obtains a previously generated virtual multi-dimensional shape equation (with any scalers) that is uniquely associated with the admission key. In some embodiments, the previously generated virtual multi-dimensional shape equation (with any scalers) can be obtained from the data storage 145.

At At 220, the shape traversal unit 120 obtains a previously generated traversal path that that is uniquely associated with the admission key. In some embodiments, the pre previously generated traversal path can be obtained from the data storage 145. Once both both the previously generated multi-dimensional run-length encoding equation (with any any scalers) and the previously generated traversal path are obtained, the method 200 then then proceeds to 235.

At At 225, the shape generation unit 115 generates a new virtual multi-dimensional sha shape equation and any scalers associated with the virtual multi-dimensional shape equ equation that will be tied to the admission key. In some embodiments, a new virtual mu multi-dimensional shape equation, including any scalers, can be generated for a new use user randomly. In some embodiments, a new virtual multi-dimensional shape equation can can be generated for a new user pseudo-randomly, including any scalers, such that the new new virtual multi-dimensional shape equation is based on, for example, information pro provided by the user. The shape generation unit 115 sends the newly generated virtual mu multi-dimensional shape equation (with any scalers) to the shape traversal unit 120 and and/or to the data storage 145 for storage. The method 200 then proceeds to 230.

At At 230, the shape traversal unit 120 receives the newly generated virtual multi-dim dimensional shape equation (with any scalers) from the shape generation unit 115 or fro from the data storage 145. The shape traversal unit 120 then uses the newly generated virt virtual multi-dimensional shape equation (with any scalers) to generate a traversal path that that is uniquely associated to the admission key. The shape traversal unit 120 can then sen send the generated traversal path to the data storage 145 for storage. The method 200 then then proceeds to 235.

A vA virtual multi-dimensional *shape* equation can be any type of mathematical equ equation that defines a virtual multi-dimensional shape. The virtual multi-dimensional sha shape equation can be based on any type of coordinate system including, but not limited to, to, a Cartesian coordinate system, a polar coordinate system, a spherical coordinate syst system, etc.

For For example, in one embodiment, the virtual multi-dimensional shape equation for for a particular admission key can be:

$$D \leq Ax^2 + By^2 + Cz^2$$

where where A, B, C and D are scalers of the equation. In this example, the virtual multi-dim dimensional shape equation is defined using a Cartesian coordinate system and rep represents a cuboid. Fig. 4 illustrates a cuboid 400 using the virtual multi-dimensional sha shape equation above. The virtual multi-dimensional shape equation creates a boundary for for the traversal path to stay within.

The The traversal path can be any pathway within the virtual multi-dimensional sha shape equation. The shape traversal unit 120 can randomly generate the traversal path wit within the virtual multi-dimensional shape defined by the virtual multi-dimensional sha shape equation. Each point along the traversal path represents a position of data. By ass associating data with each point along title traversal path, the system 100 can determine wh where data is located when retrieving a data file.

In sIn some embodiments, the shape generation unit 115 and the shape traversal unit 120 120 can switch between different types of coordinate systems for referring to specific poi points along the traversal path.

Fig Fig. 5A illustrates a traversal paith 500 with data points 505 with respect to the cub cuboid 400 shown in Fig. 4. In this emlx)diment, the traversal path 500 begins at data poi point 505a (0,0,0) (i.e., the center point of the cuboid 400 shown in Fig. 4) and exp expands out from this point through the virtual multi-dimensional shape. Each data poi point 505 along the traversal path 500 represents a position for data. As shown in Fig. 5A, 5A, data point 505n is outside of the virtual multi-dimensional shape equation (and the therefore the cuboid 400). Accordingly, as shown in Fig. 5B, the system 100 relocates the the data point 505n of the traversal path 500 such that it is within the boundary defined by by the virtual multi-dimensional shape equation. Discussion of plotting data to a trav traversal path, such as the traversal path 500, is provided below.

At At 235, the data plotting and transformation unit 125 transforms and plots the unpunprocessed data file. a Transforming the data includes the data plotting and tran transformation unit 125 obtaining the unprocessed data file (e.g., from the user interface 105 105, the data source 140, etc.) and compressing and segmenting the unprocessed data in the the unprocessed data file using one or more of run-length encoding, data deduplication, and and data transformation to form compressed data segments.

In cIn one embodiment, unprocessed data is compressed and segmented using run-len length encoding, data deduplication, and data transformation. These techniques are dis discussed in detail below. In run-length encoding, the data plotting and transformation unit unit 125 finds sequences of identical unprocessed data. The sequences of identical unpunprocessed data are replaced with a data value that has a storage size smaller than the stor storage size of a single occurrence of the sequence of identical unprocessed data.

For For example, if the unprocessed data is made up of bits [101 101 101 1101 1101 1101101111011 110101 101 101001000100010], run-length encoding can identify seq sequences within the data and create data segments for storage. Accordingly, in the abo above example, run-length encoding can create run-length encoded data segments [3-101 101], [3-1 101], [2-1 11,10], [3-101], ([3-COlO] from the unprocessed data that can be used for for storage.

The The run-length encoded data is then further compressed using data ded deduplication. In data deduplication, the data plotting and transformation unit 125 rep replaces second or later occurrences of a sequence of identical data with a reference poi pointer that points to the location of the data value representing the sequence of ide identical data. In some embodiments, the reference pointer can be coordinates of a data poi point along a traversal path where a data value is located. The reference pointer has a stor storage size smaller than the storage size of the data value representing a sequence of ide identical unprocessed data.

Usi Using the example above, the run-length encoded data segments [3-101], [3-1101101], [2-111,10], [3-101] can be further compressed to obtain data deduplicated

segsegments [3-101], [3-1 101], [2-1 1110], [A], [3-0010], where A is a reference pointer poi pointing to the data point along the traversal path where the first compressed data seg segment [3-101] is located.

TheThen, the data deduplicated data is further compressed using data tratransformation. In data transformation, the data plotting and transformation unit 125 ideidentifies sequences of unprocessed data that can be transformed to an already identified seq sequence of identical unprocessed data using a mathematical operation. These ideidentified sequences of unprocessed data can then be replaced with a reference pointer to the to the already identified sequence of identical unprocessed data with an operation indiindicator of the type of mathematical operation used to obtain the original sequence of unprocessed data. The mathematical operation can include, but is not limited to, an invinverse operation, a multiplication operation, a division operation, an addition opeoperation, a subtraction operation, an exponential operation, etc.

In tIn the above illustrated example, assuming that the inverse of the data segment [00 [00 10] is data segment [1101], data transformation can further compress the data deddeduplicated data segments [3-101], [3-1101], [2-11110], *[A*], [3-0010] to obtain tratransformed data segments [3-101], [3-1 101], [2-1 1110], *[A*], [B *] e Here, A* is a refereference pointer pointing to the locatio n along the traversal path of the first compressed dat data segment [3-101]. s Also, B* indicates a reference pointer pointing to the location alo along the traversal path of the second compressed data segment [3-1 101] and indicates per perforaiirig an inverse operation on the second compressed data segment [3-1 101] to obt obtain [3-0010]. s In some embodiments, the transformed data segments become the concompressed data segments that can then be plotted onto a traversal path.

PloPlotting the data includes the data plotting and transformation unit 125 securing the the compressed data segments by using the virtual multi-dimensional shape equation (wi(with any scalers) and the traversal path to plot the compressed data segments along the trav traversal path of the virtual multi-dimensional shape defined by the virtual multi-dimdimensional shape equation to obtain secured data segments.

TheThe data plotting and transformation unit 125 plots the compressed data segsegments (having undergone one or moife of run-length encoding, data deduplication, andand data transformation) onto the traversal path uniquely associated with the admission keykey to obtain secured data segments. cID: one embodiment, the data plotting and tran transformation unit 125 proceeds through the unprocessed data file in a specified order andand sequentially plots each sequence of unprocessed data, reference pointer, and each refereference pointer with an operation identifier to a data point along the traversal path.

In sIn some embodiments, multiple instances of the data plotting and transformation unit unit 125 can compress and segment the unprocessed data and plot the compressed data segsegments at the same time to increase the speed in which the unprocessed data file undundergoes multi-dimensional run-length encoding.

In sIn some embodiments, the traversal path may include extra data points at the end of tof the traversal path that are not used. tIn these instances, the data plotting and tran transformation unit 125 in combination with the data reading and writing unit 130 can rem remove the extra data points so that they are not written into a processed data file.

OnOnce the data plotting and transiformation unit 125 transforms and plots the unpunprocessed data to obtain secured data segments, the method 200 proceeds to 240.

At At 240, the data reading and writing unit 130 obtains the secured data segments fromfrom the data plotting and transformation unit 125, sorts the secured data segments into a gea generic format to obtain generically sorted data segments, and then writes the gengenerically sorted data segments to a file to obtain a processed data file.

In sIn some embodiments, the data plotting and transformation unit 125 can indicate thatthat a data point is to a sequence of unprocessed data by preceding the sequence of unpunprocessed data with a "0" bit followed by the number of bytes of data in the sequence of of unprocessed data. The data plotting and transformation unit 125 can indicate that a datdata point is to a reference pointer by preceding the reference pointer with a "1" bit. WhWhen the data point is to a reference pointer, the "1" bit may not be followed by the

num number of bytes of data of the reference pointer, as each reference pointer may have the sam same size.

In sIn some embodiments, the data reading and writing unit 130 can generically sort the the secured data segments based on coordinates of the traversal path in which the sec secured data is stored. That is, the data reading and writing unit 130 can sort the sec secured data segments in a generic format, for example, in ascending coordinate order, des descending coordinate order, etc. In one example, even though the traversal path starts at (at (0,0,0) then goes to (1,0,0) and then (0,1,0), the data reading and writing unit 130 can can sort the secured data in descending order and start at (1,0,0) then (0,1,0) and then (0, (0,0, 0).

TheThe data reading and writing unit 130 can then write the generically sorted data seg segments onto a file to obtain the processed data file. The data reading and writing unit 130130 can then provide the processed data file to, for example, the user interface 105 and and/or the data storage 145 for storage.

TheThe method 200 described herein can provide a processed data file that can be diffdifficult for a hacker to hack. Even if the hacker had a complete understanding of the met method 200, the hacker would not be able to read the processed data file without kn knowledge of the multi-length run-lengith encoding shape and the traversal path. WitWithout the traversal path, the hacker would be required to decode every data point in whi which data is stored and try to combine with every other data point. In order to decode eac each data point the hacker would be required to find the coordinate values for each of the the data points which would be difficult without knowledge of the traversal path.

TheThe method 200 can also provide a processed data file that is significantly sma smaller than the original unprocessed file. In some embodiments, the unprocessed data file file can be compressed into a processed data file format with a -90% data compression rati ratio or better.

Fig.Fig.13illustrates aflow chart ofa method 300 forrestoring an unprocessed data file filethat has been processed using multi-dimensional run-length encoding using the systsystem 100, according to one embodiment.

TheThe method 300 begins at 305 whereby the user interface 105 waits to receive an an admission key from a user. The admission key can be any type of password, bio biometric authentication, etc. used to authenticate the user. When the user interface 105 rec receives an admission key, the user interface 105 relays the admission key to the veriverification unit 112 and the method 300 proceeds to 310.

At At 310, the verification unit 112 determines whether the admission key is rec recognized. If the admission key is recognized, the method 300 proceeds to 315 and 320320. In some embodiments, the method 300 proceeds to 315 and 320 concurrently. If the the admission key is not recognized, the method 200 proceeds to 335.

At At 335, the verification unit 112 notifies the user that the admission key is not rec recognized and requests the user to pnmde the correct admission key.

At At 315, the shape generation unit 115 obtains a previously generated virtual mu multi-dimensional shape equation (with any scalers) that is uniquely associated with the adm admission key. In some embodiments, the previously generated virtual multi-dimdimensional shape equation (with any scalers) can be obtained from the data storage 145145.

At At 320, the shape traversal unit 120 obtains a previously generated traversal path thatthat is uniquely associated with the admission key. In some embodiments, the pre previously generated traversal path can be obtained from the data storage 145. Once botboth the previously generated multi-dimensional run-length encoding equation (with anyany scalers) and the previously generated traversal path are obtained, the method 300 thenthen proceeds to 325.

At At 325, the data reading and writing unit 130 obtains the processed data file (e.g(e.g., from the user interface 105, the data storage 145, etc.), reads the generically sorted datdata segments from the processed data Hie, and provides the generically sorted data

segsegments to the data plotting and transformation unit 125. The method 300 then proproceeds to 330.

At At 330, the data plotting and transformation unit 125 uses the previously gengenerated virtual multi-dimensional shape equation (with any scalers) and the traversal pathpath that are uniquely associated to the admission key in order to arrange the generically sortsorted data segments in the proper order to obtain the compressed data segments, and decdecompresses the compressed data segments that have undergone one or more of run-lenlength encoding, data deduplication, and data transformation to obtain the unprocessed datdata that can be formed into the unprocessed file. In particular, the data plotting and tratransformation unit 125 can arrange the generically sorted data segments at their respective data points. The data plotting and transformation unit 125 can then reareassemble the compressed data segments and write the uncompressed data in the order of tof the traversal path to obtain the unprocessed data file.

In sIn some embodiments, the data plotting and transformation unit 125 can reareassemble the unprocessed data file at multiple different data points along the traversal pathpath at the same time. hWhen a particuktr data point includes a reference pointer that poipoints to a data point that has not yet been reassembled, the data plotting and tratransformation unit 125 can move onto a different data point along the traversal path anddand come back to this data point later. This can increase the speed in which the unpunprocessed data file can be restored.

OnOnce all of the compressed data segments are reassembled into the unprocessed datdata file, the unprocessed data file can then be provided to the user.

Fig Fig. 6 is a schematic diagram of an exemplary architecture for a computer devdevice 1200, such as the one or more computer devices described above with respect to Fig Figs. 1-5. The computer device 1200 and any of the individual components thereof can be be used for any of the operations descried in accordance with any of the computer-imimplemented systems and methods described herein.

The computer device 1200 generally includes a processor 1210, memory 1220, a network input/output (I/O) 1225, storage 1230, and an interconnect 1250. The computer device 1200 can optionally include a user I/O 1215, according to some embodiments. The computer device 1200 can be in communication with one or more additional computer devices 1200 through a network 1240.

The computer device 1200 is generally representative of hardware aspects of a variety of user devices 1201 and a server device 1235. The illustrated user devices 1201 are exemplary and are not intended to be limiting. Examples of the user devices 1201 include, but are not limited to, a desktop computer 1202, a cellular/mobile phone 1203, a tablet device 1204, and a laptop computer 1205. It is to be appreciated that the user devices 1201 can include other devices such as, but not limited to, a personal digital assistant (PDA), a video game console, a television, or the like. In some embodiments, the user devices 1201 can alternatively be referred to as client modules 1201. In such embodiments, the client modules 1201 can be in communication with the server device 1235 through the network 1240. One or more of the client modules 1201 can be in communication with another of the client modules 1201 through the network 1240 in some embodiments.

The processor 1210 can retrieve and execute programming instructions stored in the memory 1220 and/or the storage 1230. The processor 1210 can also store and retrieve application data residing in the memory 1220. The interconnect 1250 is used to transmit programming instructions and/or application data between the processor 1210, the user I/O 1215, the memory 1220, the storage 1230, and the network I/O 1225. The interconnect 1250 can, for example, be one or more busses or the like. The processor 1210 can be a single processor, multiple processors, or a single processor having multiple processing cores. In some embodiments, the processor 1210 can be a single-threaded processor. In some embodiments, the processor 1210 can be a multi-threaded processor.

The user I/O 1215 can include a display 1216 and/or an input 1217, according to some embodiments. It is to be appreciated that the user I/O 1215 can be one or more devices connected in communication with the computer device 1200 that is physically separate from the computer device 1200. For example, the display 1216 and input 1217 for the desktop computer 1202 can be connected in communication but be physically separate from the computer device 1200. In some embodiments, the display 1216 and input 1217 can be physically included with the computer device 1200 for the desktop computer 1202. In some embodiments, the user I/O 1215 can physically be part of the user device 1201. For example, the cellular/mobile phone 1203, the tablet device 1204, and the laptop 1205 include the display 1216 and input 1217 that are part of the computer device 1200. The server device 1235 generally may not include the user I/O 1215. In some embodiments, the server device 1235 can be connected to the display 1216 and input 1217.

The display 1216 can include any of a variety of display devices suitable for displaying information to the user. Examples of devices suitable for the display 1216 include, but are not limited to, a cathode ray tube (CRT) monitor, a liquid crystal display (LCD) monitor, a light emitting diode (LED) monitor, or the like.

The input 1217 can include any of a variety of input devices or means suitable for receiving an input from the user. Examples of devices suitable for the input 1217 include, but are not limited to, a keyboard, a mouse, a trackball, a button, a voice command, a proximity sensor, an ocular sensing device for detennining an input based on eye movements (e.g., scrolling based on an eye movement), or the like. It is to be appreciated that combinations of the foregoing inputs 1217 can be included for the user devices 1201. In some embodiments the input 1217 can be integrated with the display 1216 such that both input and output are performed by the display 1216.

The memory 1220 is generally included to be representative of a random access memory such as, but not limited to, Static Random Access Memory (SRAM), Dynamic Random Access Memory (DRAM), or Hash. In some embodiments, the memory 1220

can can be a volatile memory. In some embodiments, the memory 1220 can be a non-vol volatile memory. In some embodiments, at least a portion of the memory can be virtual me memory.

The The storage 1230 is generally included to be representative of a non-volatile me memory such as, but not limited to, a hard disk drive, a solid state device, removable me memory cards, optical storage, flash memory devices, network attached storage (NAS), or con connections to storage area network (SAN) devices, or other similar devices that may stor store non-volatile data. In some embodiments, the storage 1230 is a computer readable me medium. In some embodiments, the storage 1230 can include storage that is external to the the computer device 1200, such as in a cloud.

The The network I/O 1225 is configured to transmit data via a network 1240. The net network 1240 may alternatively be referred to as the communications network 1240. Exa Examples of the network 1240 include, but are not limited to, a local area network (LA (LAN), a wide area network (WAN), the Internet, or the like. In some embodiments, the the network I/O 1225 can transmit data via the network 1240 through a wireless con connection using WiFi, Bluetooth, or other similar wireless communication protocols. In In some embodiments, the computer device 1200 can transmit data via the network 124 1240 through a cellular, 3G, 4G, or other wireless protocol. In some embodiments, the net network I/O 1225 can transmit data via a wire line, an optical fiber cable, or the like. It is is to be appreciated that the network I/O 1225 can communicate through the network 124 1240 through suitable combinations of the preceding wired and wireless communication me methods.

The The server device 1235 is generally representative of a computer device 1200 that that can, for example, respond to requests received via the network 1240 to provide, for exa example, data for rendering a website to a the user devices 1201. The server device 1235 can can be representative of a data server, an application server, an Internet server, or the like like.

As Aspects described herein can be embodied as a system, method, or computer rea readable medium. sIn some embodiments, the aspects described can be implemented in har hardware, tsoftware (including firmware for the like), or combinations thereof. Some asp aspects can be implemented in a computer readable medium, including computer rea readable tinstructions for execution by a processor. Any combination of one or more con computer readable niedium(s) can be used.

The The computer readable medium can include a computer readable signal medium and and/or a computer readable storage medium. A computer readable storage medium can inc include any tangible medium capable of storing a computer program for use by a pro programmable processor to perform functions described herein by operating on input dat data and generating an output. A computer program is a set of instructions that can be use used, directly or indirectly, in a computer system to perform a certain function or det determine a certain result. Examples of computer readable storage media include, but are are not limited to, a floppy disk; a hard disk; a random access memory (RAM); a read-onl only memory (ROM); a semiconductor memory device such as, but not limited to, an era erasable programmable read-only memory (EPROM), an electrically erasable pro programmable read-only memory (EEPROM), Flash memory, or the like; a portable con compact disk read-only memory (CD-ROM); an optical storage device; a magnetic sto storage device; other similar device; or t suitable combinations of the foregoing. A con computer readable signal medium can include a propagated data signal having computer rea readable tinstructions. Examples of propagated signals include, but are not limited to, an opt optical propagated signal, an electro-maignetic propagated signal, or the like. A con computer readable signal medium can include any computer readable medium that is not not a computer readable storage medium that can propagate a computer program for use by by a programmable processor to perform functions described herein by operating on inp input data and generating an output.

Some embodiments can be provided to an end-user through a cloud-computing infrastructure. Cloud computing generally includes the provision of scalable computing resources as a service over a network (e.g., the Internet or the like).

Although a number of methods and systems are described herein, it is contemplated that a single system or method can include more than one of the above discussed subject matter. Accordingly, multiple of the above systems and methods can be used together in a single system or method.

The examples disclosed in this application are to be considered in all respects as illustrative and not limitative. The scope of the invention is indicated by the appended claims rather than by the foregoing description; and all changes which come within the meaning and range of equivalency of the claims are intended to be embraced therein.

CLCLMMS

1. 1. A nA tmethod for multi-dimensional run-length encoding of an unprocessed data file file, the tmethod comprising:

obt obtaining an admission key;

det determining a traversal path witliin a virtual multi-dimensional shape based on the the admission key;

tra transforming unprocessed data of the unprocessed data file into a plurality of con compressed data segments;

plo plotting each of the plurality of compressed data segments onto a data point alo along the traversal path to obtain a plurality of secured data segments;

gen generically tsorting the tsecured dlata segments to obtain a plurality of generically sort sorted data segments; and

wri writing the plurality of generically tsorted data segments into a processed data file file.

2. 2. TheThe tmethod of claim m, wherein determining the traversal path within the virtual mu multi-dimensional shape based ton the admission key includes:

ver verifying the admission key;

gen generating a tvirtual multi-dimensional shape tequation defining the virt virtual multi-dimensional shape tthat is unique to the admission key when the adm admission key is not verified; and

gen generating the traversal path within the virtual multi-dimensional shape that that is unique to the admission key when the admission key is not verified; or

wh wherein determining the traversal path within the virtual multi-dimensional sha shape based ton the admission key includes:

ver verifying the admission key;

obtaining from a storage location a virtual multi-dimensional shape equation defining the virtual multi-dimensional shape that is unique to the admission key when the admission key is verified; and

obtaining from the storage location the traversal path within the virtual multi-dimensional shape that is unique to the admission key when the admission key is verified.

3. The method of claim 2, wherein the virtual multi-dimensional shape equation includes a scaler that is unique to the admission key.

4. The method of any one of claims 1-3, wherein transforming the unprocessed data of the unprocessed data file into the plurality of compressed data segments includes identifying a first sequence of unprocessed data and a second sequence of unprocessed data that is identical to the first sequence of unprocessed data, replacing the first and second sequences of unprocessed data with a data value that has a storage size that is smaller than the storage size of each of the first and second sequences of unprocessed data, and

wherein the data value is a first compressed data segment of the plurality of compressed data segments and the data value is a second compressed data segment of the plurality of compressed data segments; or

wherein transforming the unprocessed data of the unprocessed data file into the plurality of compressed data segments includes identifying a first sequence of unprocessed data and a second sequence of unprocessed data that is identical to the first sequence of unprocessed data, replacing the first sequence of unprocessed data with a data value that has a storage size that is smaller than the storage size of the first and second sequences of unprocessed data, and replacing the second sequence of unprocessed data with a reference pointer that points to a data point of the data value,

wherein the reference pointer has a storage size that is smaller than the storage size of each of the first and second sequences of unprocessed data, and

wherein the data value is a first compressed data segment of the plurality of compressed data segments and the reference pointer is a second compressed data segment of the plurality of compressed data segments; or

wherein transforming the unprocessed data of the unprocessed data file into the plurality of compressed data segments includes identifying a first sequence of unprocessed data and a second sequence of unprocessed data that when applying a mathematical operation on the second sequence of unprocessed data transforms the second sequence of unprocessed data into a new sequence of unprocessed data that is identical to the first sequence of unprocessed data, replacing the first sequence of unprocessed data with a data value that has a storage size that is smaller than the storage size of each of the first and second sequences of unprocessed data, and replacing the second sequence of unprocessed data with a reference pointer that points to the data value and an operation indicator indicating the mathematical operation applied to the second sequence of unprocessed data to obtain the first sequence of unprocessed data,

wherein the data value is a first compressed data segment of the plurality of compressed data segments, and

wherein the reference pointer and the operation indicator is a second compressed data segment of the plurality of compressed data segments.

5. 5. A An method for restoring a data file that has been processed using multi-dimensional run-length encoding, the method comprising:

obtaining an admission key;

reading a plurality of genetically sorted data segments that forms a processed data file;

determining a traversal path within a virtual multi-dimensional shape based on the admission key;

plotplotting the plurality of genetically sorted data segments onto a plurality of data poipoints along the traversal path to obtain a plurality of compressed data segments;

reareassembling the plurality of compressed data segments into unprocessed data thatthat forms the data file.

6. 6. TheThe method of claim 5, wherein determining the traversal path within the virtual mulmulti-dimensional shape based on the admission key includes:

ververifying the admission key;

obtobtaining from a storage location a virtual multi-dimensional shape equation defidefining the virtual multi-dimensional shape that is unique to the admission key when the the admission key is verified; and

obtobtaining from the storage location the traversal path within the virtual multi-dimdimensional shape that is unique to the admission key when the admission key is ververified.

7. 7. TheThe method of claim 6, wherein the virtual multi-dimensional shape equation incincludes a scaler that is unique to the admission key.

8. 8. TheThe method of any one of claims 5-7, wherein reassembling the plurality of concompressed data segments into the unprocessed data includes:

reareading at a data point of the plurality of data points a data value representing a sequence of unprocessed data; and

reareplacing the data value with the sequence of unprocessed data; or

wherein reassembling the plurality of compressed data segments into the unprocessed data includes:

reareading at a first data point of the plurality of data points a reference poipointer that points to a second data point of the plurality of data points providing

a fia first instance of a first compressed data segment of the plurality of compressed dat data segments; and

rep replacing the reference pointer with the first compressed data segment; or or

wh wherein reassembling the plurality of compressed data segments into the unpunprocessed data includes:

rea reading at a first data point of the plurality of data points a reference poi pointer that points to a second data point of the plurality of data points providing a fia first instance of a first compressed data segment of the plurality of compressed dat data segments;

rea reading at the first data point an operation indicator representing a mat mathematical operation;

rep replacing the reference pointer with the first compressed data segment; and and

app applying the mathematical operation onto a sequence of unprocessed dat data represented by the first compressed data segment.

9. 9. A nA multi-dimensional run-length encoding system comprising:

a usa user interface that receives an admission key from a user; and

a ma multi-dimensional run-length encoding engine that includes:

a ve a verification unit that verifies the admission key,

a sha shape generation unit that determines a virtual multi-dimensional shape equequation that is unique to the admission key,

a sha shape traversal unit that determines a traversal path within a virtual mu multi-dimensional shape defined by the virtual multi-dimensional shape equequation and mat is unique to the admission key,

a da data plotting and transformation unit that transforms unprocessed data into into a plurality of compressed data segments and plots each of the plurality of

concompressed data segments onto a separate data point among a plurality of data poi points along the traversal path to form a plurality of secured data segments, and

a data reading and writing unit that generically sorts the plurality of sec secured data segments to form a plurality of generically sorted data segments and writes the plurality of generically sorted data segments into a processed data file file.

10. 10. The The multi-dimensional run-length encoding system of claim 9, wherein the virt virtual multi-dimensional shape equation includes a scaler that is unique to the ad admission key.

11. 11. The The multi-dimensional run-length encoding system of any one of claims 9 or 10, wh wherein the data plotting and transformation unit identifies a first sequence of unp unprocessed data and a second sequence of unprocessed data that is identical to the first seq sequence of unprocessed data, and replaces the first and second sequences of unp unprocessed data with a data value that has a storage size that is smaller than the storage size size of each of the first and second sequences of unprocessed data, and

wh wherein the data value is a first compressed data segment of the plurality concompressed data segments and the data value is a second compressed data segment of the the plurality of compressed data segments; or

wh wherein the data plotting and transformation unit identifies a first sequence of unp unprocessed data and a second sequence of unprocessed data that is identical to the first seq sequence of unprocessed data, and replaces the first sequence of unprocessed data with a da data value and replaces the second sequence of unprocessed data with a reference poi pointer that points to the data value, wherein the reference pointer has a storage size that is s is smaller than the storage size of each of the first and second sequences of unprocessed dat data, and

wherein the data value is a first compressed data segment of the plurality of compressed data segments and the reference pointer is a second compressed data segment of the plurality of compressed data segments; or

wherein the data plotting and transformation unit identifies a first sequence of unprocessed data and a second sequence of unprocessed data that when applying a mathematical operation on the second sequence of unprocessed data transforms the second sequence of unprocessed data into a new sequence of unprocessed data that is identical to the first sequence of unprocessed data, and replaces the first sequence of unprocessed data with a data value and replaces the second sequence of unprocessed data with a reference pointer that points to the data value and an operation indicator indicating the mathematical operation applied to the second sequence of unprocessed data to obtain the first sequence of unprocessed data,

wherein the data value is a first compressed data segment of the plurality of compressed data segments, and

wherein the reference pointer and the operation indicator is a second compressed data segment of the plurality of compressed data segments.

12. 12. A nA multi-dimensional run-length encoding system comprising:

a us a user interface that receives an admission key from a user; and

a ma multi-dimensional run-length encoding engine that includes:

a v a verification unit that verifies the admission key,

a sh a shape generation unit that determines a virtual multi-dimensional shape equequation that is unique to the admission key,

a sh a shape traversal unit that determines a traversal path within a virtual mu multi-dimensional shape defined by the virtual multi-dimensional shape equequation and that is unique to the admission key,

a d a data reading and writing unit that reads a plurality of genetically sorted dat data segments from a processed data file, and

a data plotting and transformation unit that plots each of the plurality of generically sorted data segments onto a separate data point among a plurality data data points along the traversal path to obtain a plurality of compressed data segments, and reassembles the plurality of compressed data segments to obtain unprocessed data that forms an unprocessed data file.

13. The multi-dimensional run-length encoding system of claim 12, wherein the virtual multi-dimensional shape equation includes a scaler that is unique to the admission key.

14. The multi-dimensional run-length encoding system of any one of claims 12 or 13, wherein the data plotting and transformation unit reads at a data point of the plurality of data points a data value representing a sequence of unprocessed data, and replaces the data value with the sequence of unprocessed data.

15. The multi-dimensional run-length encoding system of any one of claims 12-14, wherein the data plotting and transformation unit reads at a first data point of the plurality of data points a reference pointer that points to a second data point of the plurality of data points that provides a first instance of a first compressed data segment of the plurality of compressed data segments, and replaces the reference pointer with the first compressed data segment; or

wherein the data plotting and transformation unit reads at a first data point of the plurality of data points a reference pointer that points to a second data point of the plurality of data points that provides a first instance of a first compressed data segment of the plurality of compressed data segments, reads at the first data point an operation indicator representing a mathematical operation, replaces the reference pointer with the first compressed data segment, and applies the mathematical operation onto a sequence of unprocessed data represented by the first compressed data segment.

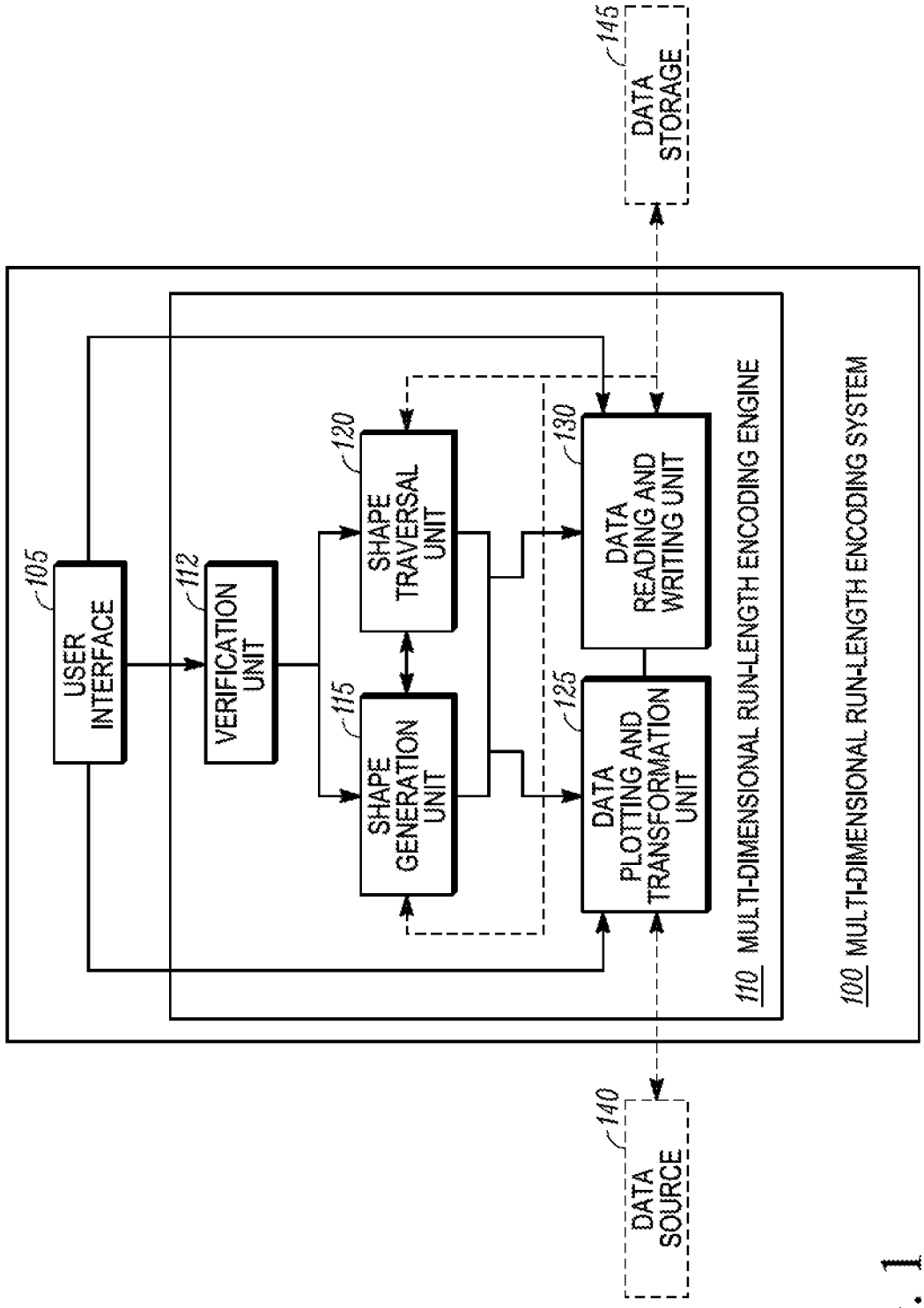
100 MULTI-DIMENSIONAL RUN-LENGTH ENCODING SYSTEM
110 MULTI-DIMENSIONAL RUN-LENGTH ENCODING ENGINE
105 USER INTERFACE
112 VERIFICATION UNIT
115 SHAPE GENERATION UNIT
120 SHAPE TRAVERSAL UNIT
125 DATA PLOTTING AND TRANSFORMATION UNIT
130 DATA READING AND WRITING UNIT
140 DATA SOURCE
145 DATA STORAGE

FIG. 1

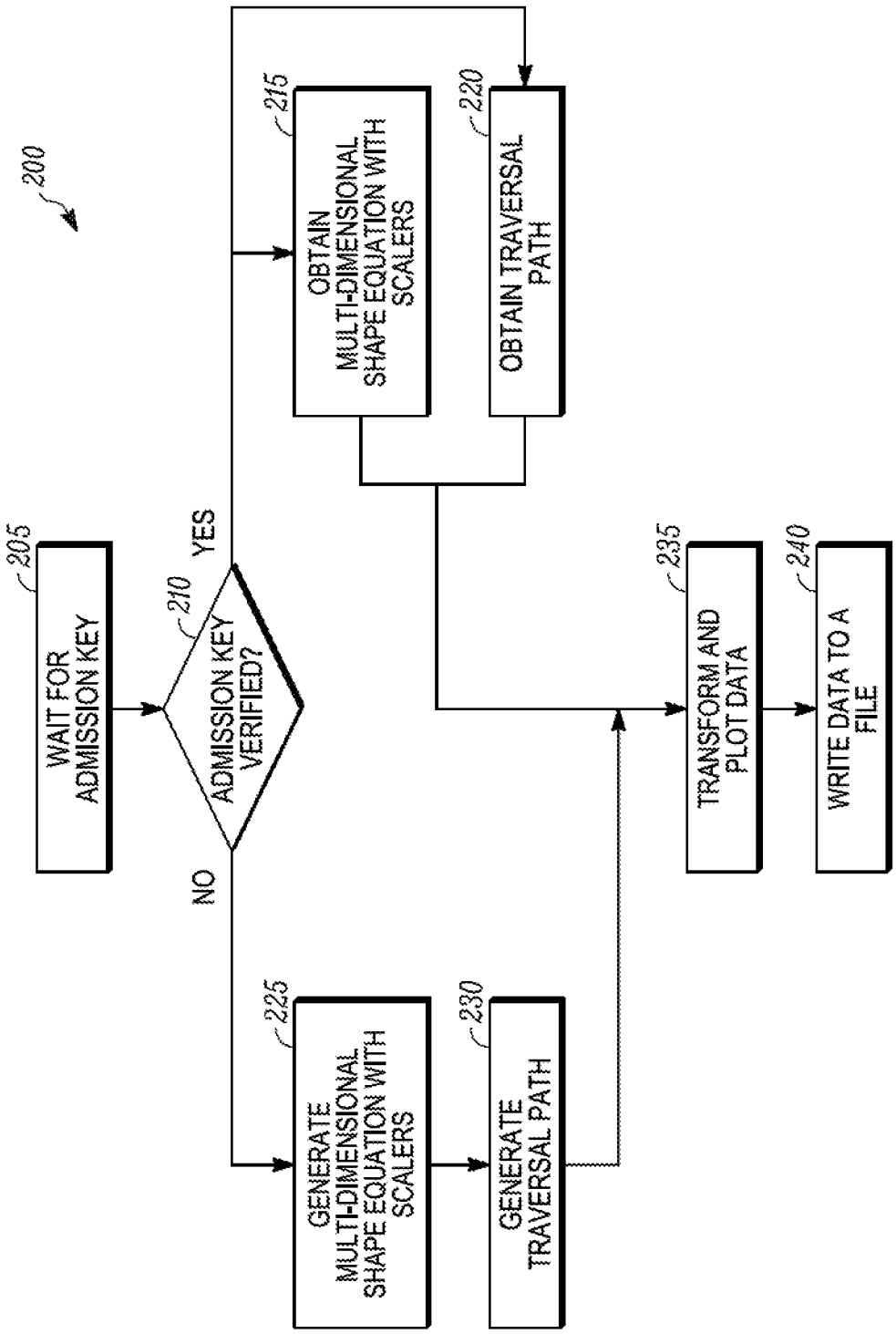
200
205
WAIT FOR ADMISSION KEY
210
ADMISSION KEY VERIFIED?
YES
NO
215
OBTAIN MULTI-DIMENSIONAL SHAPE EQUATION WITH SCALERS
220
OBTAIN TRAVERSAL PATH
225
GENERATE MULTI-DIMENSIONAL SHAPE EQUATION WITH SCALERS
230
GENERATE TRAVERSAL PATH
235
TRANSFORM AND PLOT DATA
240
WRITE DATA TO A FILE

FIG. 2

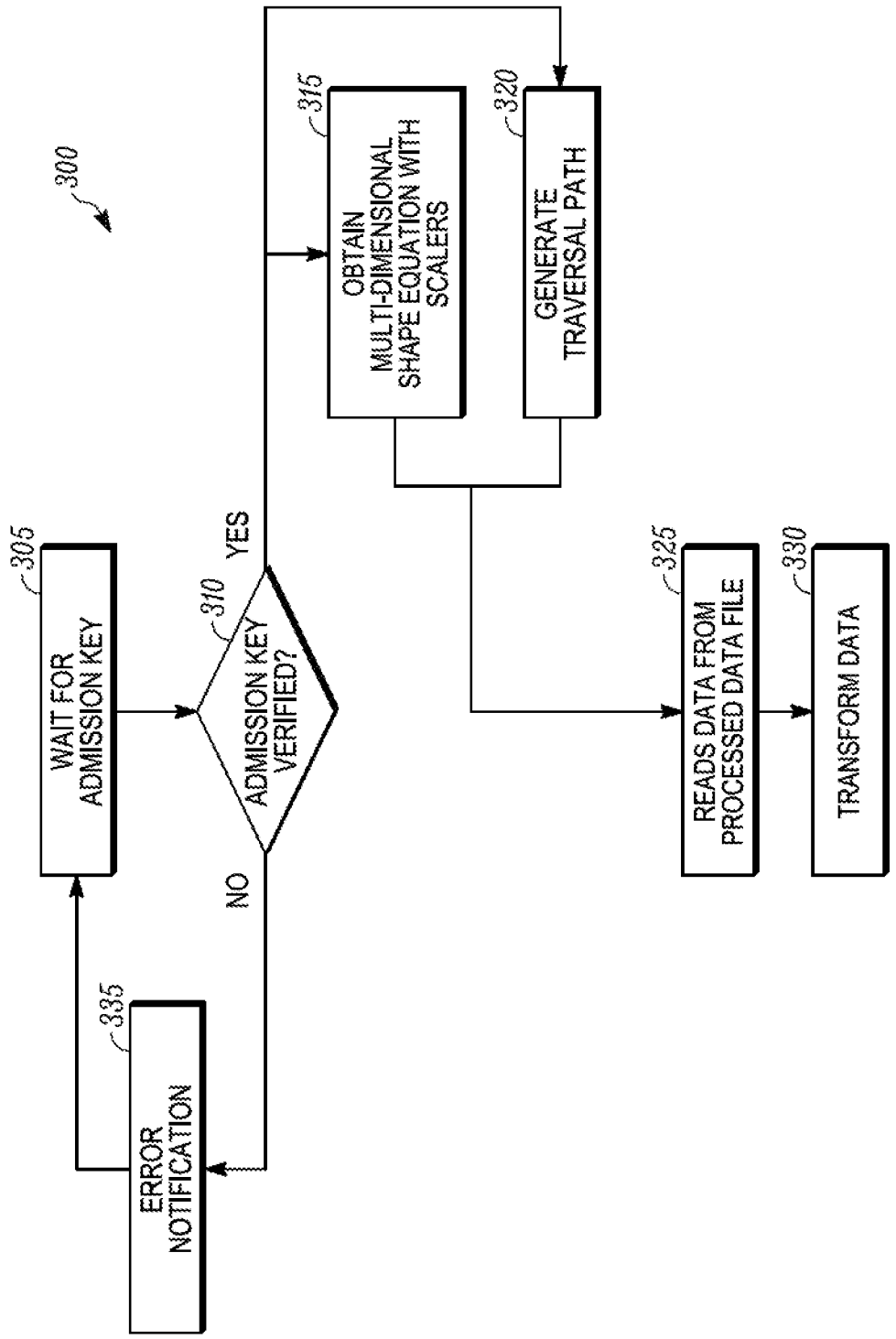
300
305
WAIT FOR ADMISSION KEY
310
ADMISSION KEY VERIFIED?
YES
NO
315
OBTAIN MULTI-DIMENSIONAL SHAPE EQUATION WITH SCALERS
320
GENERATE TRAVERSAL PATH
325
READS DATA FROM PROCESSED DATA FILE
330
TRANSFORM DATA
335
ERROR NOTIFICATION

FIG. 3

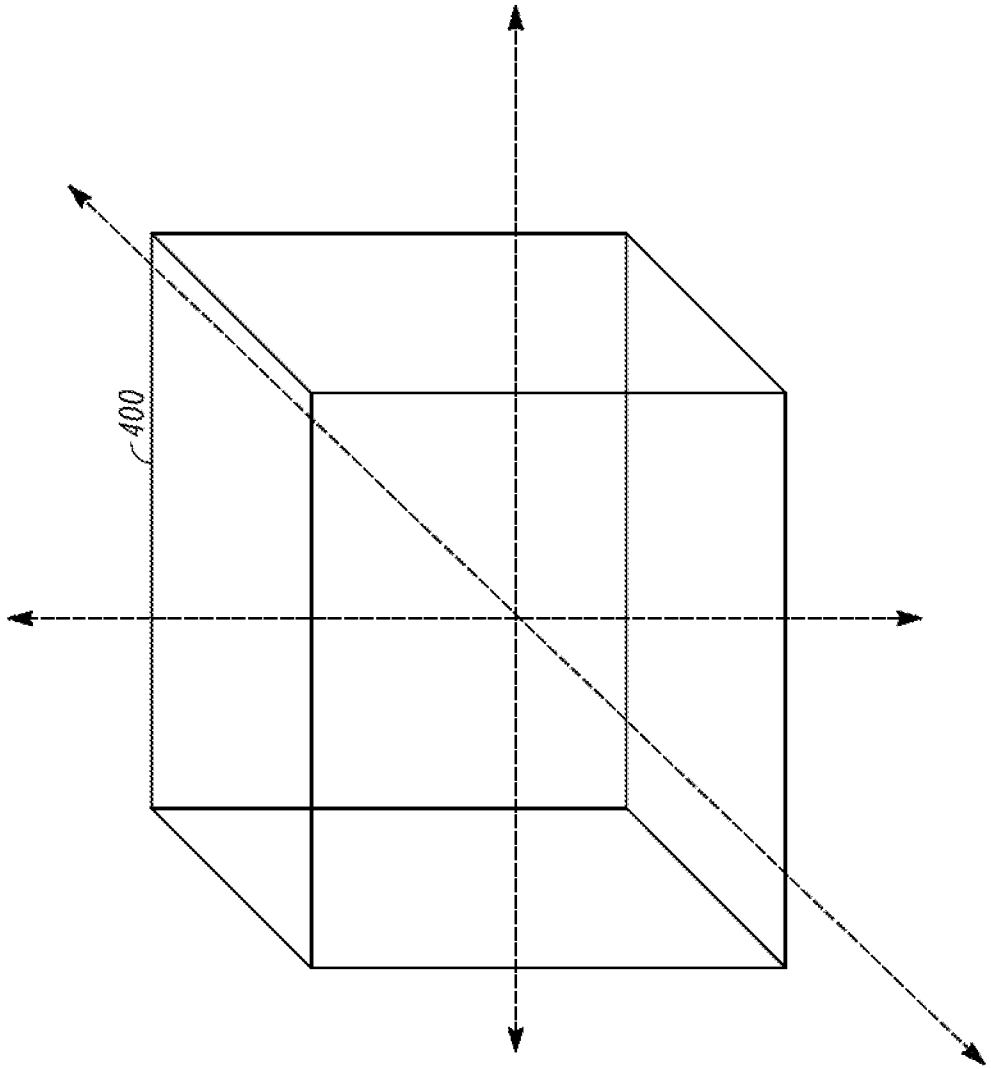

FIG. 4

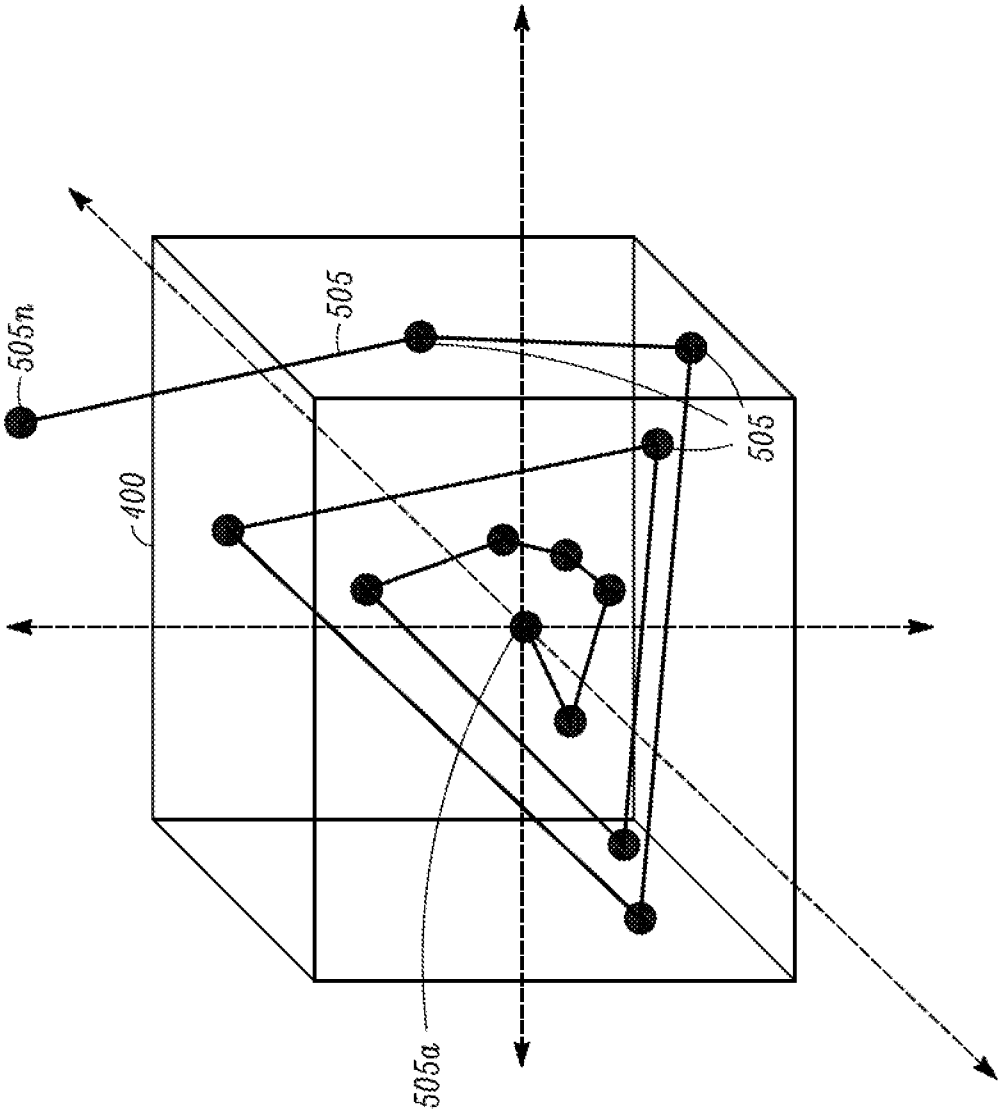

FIG. 5A

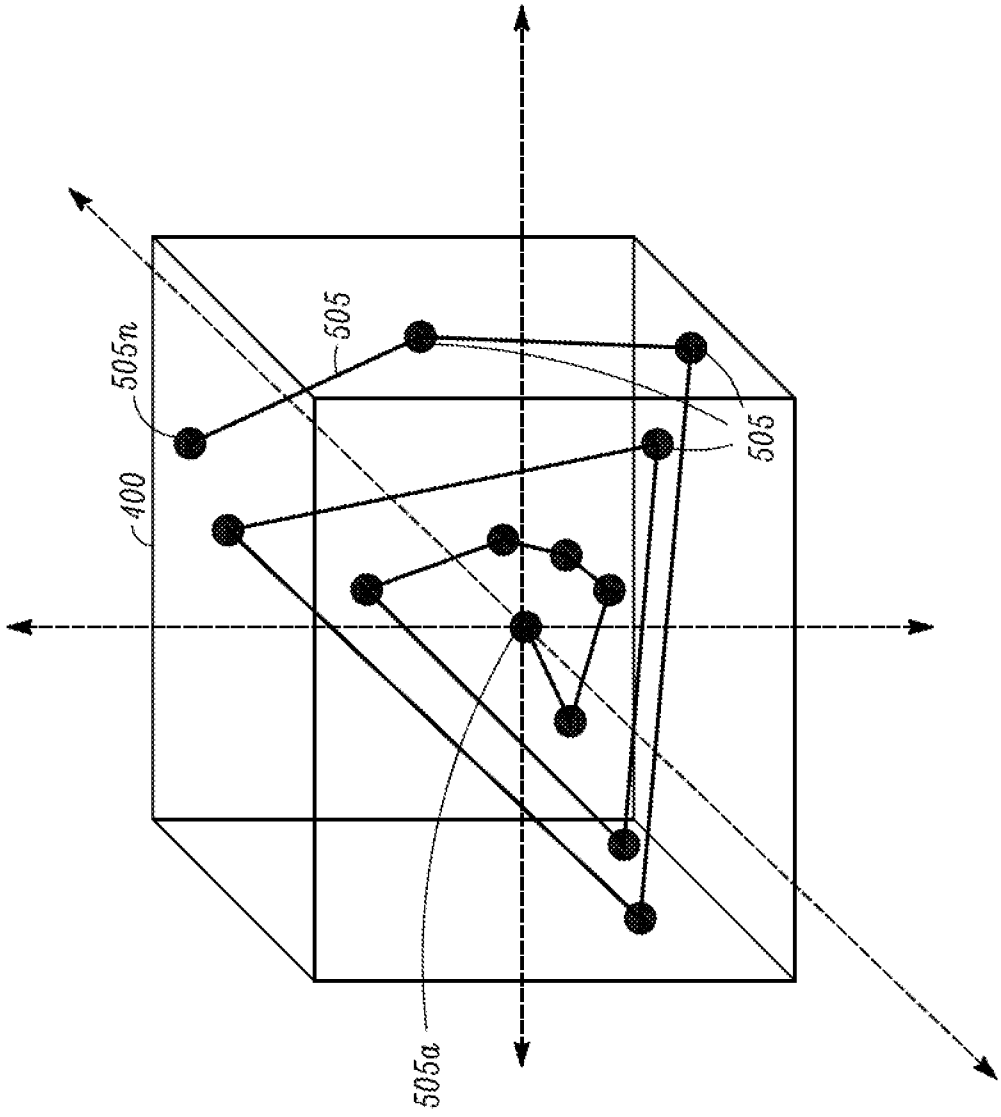

FIG. 5B

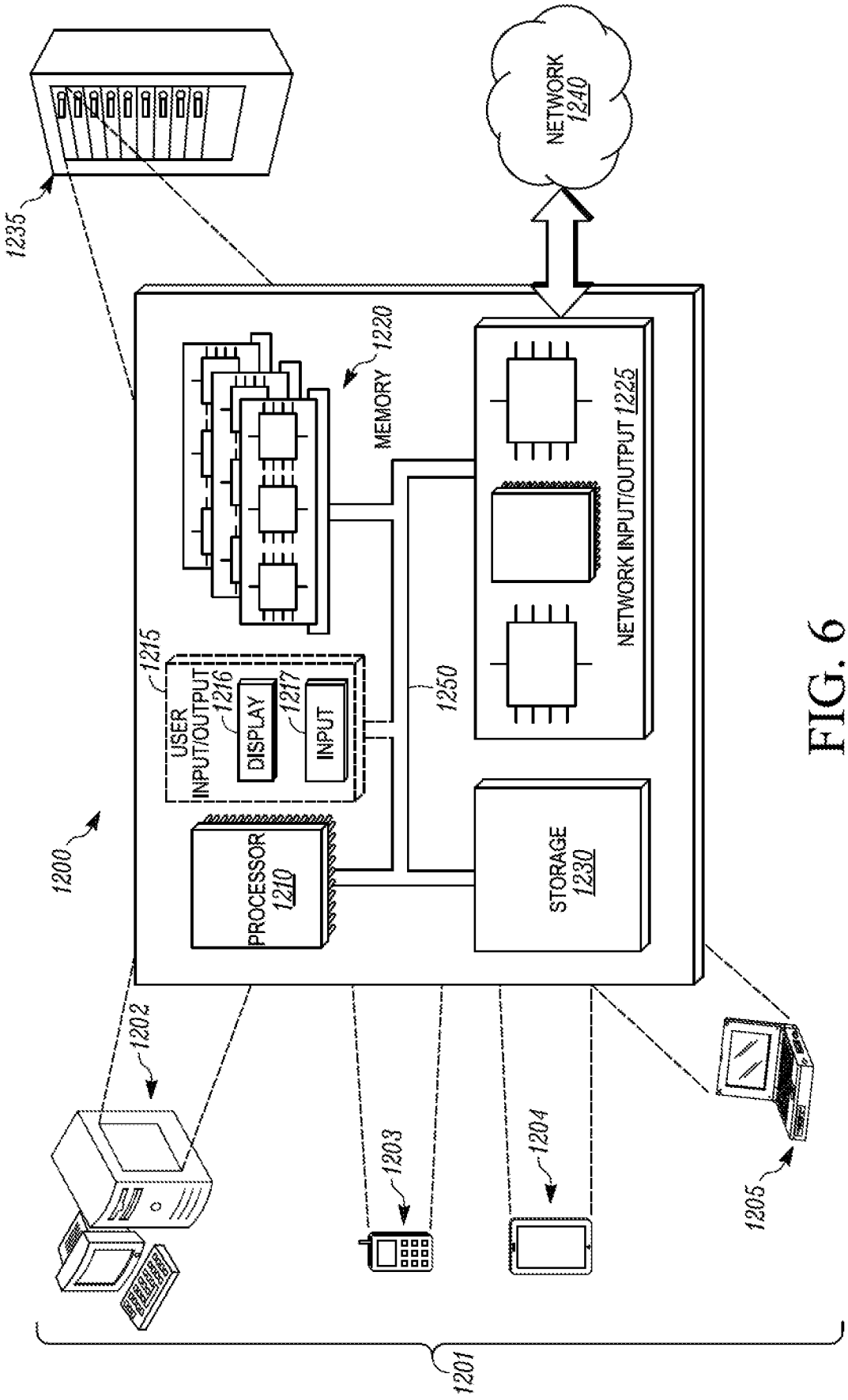
NETWORK
1240
1235
MEMORY 1220
NETWORK INPUT/OUTPUT 1225
1215
USER
INPUT/OUTPUT
1216
DISPLAY
1217
INPUT
1250
PROCESSOR
1210
STORAGE
1230
1200
1202
1203
1204
1205
1201

FIG. 6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2017/048207

A. CLASSIFICATION OF SUBJECT MATTER

H03M 7/46(2006.01)i, G06F 21/31(2013.01)i, G06F 21/62(2013.01)i

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)
H03M 7/46; H04L 9/08; H03M 7/30; H03M 1/34; H03M 5/14; G01S 5/02; G06F 21/3 1; G06F 21/62

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords: admission key, traveral path, transform, compressed data segments, sort, multi-dimensional run-length encoding

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2015-0311916 Al (LAURI AARNE JOHANNES WIROLA et al.) 29 Oct ober 2015 See paragraphs [0039H0068] , [0092H0171] ; and f igures 1-4, 7-11 .	1-15
A	US 2009-0060198 Al (MARK CAMERON LITTLE) 05 March 2009 See paragraphs [0023]- [0068] ; and f igures 1A 1C, 5.	1-15
A	EP 2381579 Al (ITRON METERING SOLUTIONS UK LTD) 26 Oct ober 2011 See paragraphs [0024] - [0039]; and f igures 5-7 .	1-15
A	US 2014-0104083 Al (WUHAN TEXTILE UNIVERSITY) 17 Apr i l 2014 See paragraphs [0089]- [0106] ; and f igures 8-11 .	1-15
A	US 2008-0024350 Al (SUDARSHAN LAKSHMIPU SRINIVASAN et al.) 31 January 2008 See paragraphs [0034]- [0044] ; and f igures 1, 2.	1-15

☐ 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
05 December 20 17 (05. 12.20 17)	**12 December 2017 (12.12.2017)**

Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea
Facsimile No. +82-42-481-8578

Authorized officer
AHN, Jeong Hwan
Telephone No. +82-42-481-8633

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2017/048207

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2015-0311916 Al	29/10/2015	CN 104285377 A	14/01/2015
		EP 2826149 Al	21/01/2015
		US 9319065 B2	19/04/2016
		WO 2013-136129 Al	19/09/2013
US 2009-0060198 Al	05/03/2009	US 8064599 B2	22/11/2011
EP 2381579 Al	26/10/2011	None	
US 2014-0104083 Al	17/04/2014	CN 102298953 A	28/12/2011
		CN 102298953 B	30/04/2014
		US 8957792 B2	17/02/2015
		WO 2012-163070 Al	06/12/2012
US 2008-0024350 Al	31/01/2008	US 7417570 B2	26/08/2008