

## **TWO-DIMENSIONAL CODE HAVING ERROR CORRECTION AND METHOD AND SYSTEM FOR GENERATING AND READING THEREOF**

### **FIELD OF THE INVENTION**

**[0001]** Embodiments of the invention relate to two-dimensional (2D) machine-readable code having error correction feature, including 2D images containing machine-readable data and error correction feature, and particularly to its representation, and method and system for generating and reading such 2D codes.

### **BACKGROUND**

**[0002]** One-dimensional (1D) and 2D codes are commonly used from price tags on grocery items in supermarkets to tickets for concerts or postal mail. Non-limiting examples include barcodes and other global trade item numbers (GTINS). It is reported that GTINs are scanned up to 10 billion times daily. Existing 1D and 2D codes have error correction codes built into them. Error correction code most commonly used with barcodes and other 2D codes is the Reed-Solomon error correction code. The 1D codes and 2D codes often represent binary language, wherein each line or pixel represents a binary bit, i.e., 1 or 0. The Reed-Solomon code is applied to the 1D or 2D code to ensure readability of the 1D or 2D code should it be damaged.

**[0003]** Household dust measuring approximately 0.5 microns to 100 microns is a significant impediment to scanning 1D codes or 2D codes. While there are other causes of damage, household dust due to its common occurrence would be associated with the majority of damage.

**[0004]** When the Reed-Solomon code is applied to 1D or 2D codes, there is a tolerance for a certain percentage of pixels or lines that can be damaged while ensuring readability. The way the Reed-Solomon code works is mathematically complex and was first introduced in 1960. Today, it is not only used in 1D or 2D barcodes but also used extensively to correct data transmission errors in mobile phones, compact discs, satellites, space probes, as well as other data storage and examples.

**[0005]** *Downs, Adam S., N. P. Sigmon and Richard Klíma. "The Mathematics of QR Codes." (2021),* incorporated by reference herein, describes some basics of how Reed-Solomon codes work, and how Reed-Solomon codes are incorporated into QR codes to increase the likelihood that they are able to transmit data reliably. While the Reed-Solomon code has been applied to QR code which involves binary bits, it would be challenging to apply the Reed-Solomon code to 1D or 2D code which are non-binary or are symbology-based.

**[0006]** Accordingly, there is a need for alternative and useful error correction techniques which are suitable at least for symbology-based 1D or 2D codes.

## **SUMMARY**

**[0007]** According to a first aspect of the invention, a 2D code is provided which comprises:

a plurality of optically readable and distinct data indicia, being different symbols, arranged as a first 2D code arrangement of the data indicia which represents a first data value;

an error correction feature which is incorporated into the data indicia of the first 2D code arrangement without adding data bit to the first 2D code arrangement,

wherein the error correction feature is based on a cluster size of each data indicium being larger than a predetermined pixel size, wherein the predetermined pixel size is based on a pixel size of a potential error, wherein at least a first one of the data indicia includes a plurality of separate portions wherein at least one of the separate portions has a cluster size larger than the predetermined pixel size; and

a reference indicium arranged within the first 2D code arrangement, wherein the first data value is represented by a plurality of distances between the reference indicium and the data indicia of the first 2D code arrangement, and by a plurality of angular displacements of the data indicia of the first 2D code arrangement which are relative to a reference direction, wherein the reference direction is based on a predetermined one of the data indicia or the reference indicium which is asymmetric or includes at most one line of symmetry.

**[0008]** Embodiments of the first aspect are provided as recited in claim 1 to claim 4.

**[0009]** According to a second aspect of the invention, a plurality of 2D codes are provided which comprise:

a plurality of 2D code arrangements of optically readable and distinct data indicia, being different symbols, wherein the 2D code arrangements of data indicia respectively represent a plurality of data values;

an error correction feature which is incorporated into the data indicia without adding data bit to each of the 2D code arrangements,

wherein the error correction feature is based on a cluster size of each of the data indicia being larger than a predetermined pixel size, wherein the predetermined pixel size is based on a pixel size of a potential error, wherein at least one of the data indicia includes a plurality of separate portions wherein at least one of the separate portions has a cluster size larger than the predetermined pixel size; and

a reference indicium arranged within each of the 2D code arrangements, wherein the data value of each of the 2D code arrangements is represented by distances between the reference indicium and the data indicia of the each of the 2D code arrangements, and by angular displacements of the data indicium of the each of the 2D code arrangements which are relative to a reference direction, wherein the reference direction is based on a predetermined one of the data indicia or the reference indicium which is asymmetric or includes at most one line of symmetry.

**[0010]** Embodiments of the second aspect are provided as recited in claim 6 to claim 7.

**[0011]** According to a third aspect of the invention, a method for 2D code generation is provided wherein the method comprises:

providing the 2D code according to any one of claim 1 to claim 4, including:

based on a plurality of bijective mappings between a plurality of combinations of code parameters and a plurality of data values to provide the first 2D code arrangement, generating the 2D code for a target data value being the first data value, wherein the code parameters include the distances and the angular displacements.

**[0012]** According to a fourth aspect of the invention, a 2D code reading method is provided wherein the method comprises:

based on a first 2D code arrangement of a plurality of optically readable and distinct data indicia, being different symbols,, wherein the first 2D code arrangement of the data indicia represents a first data value, wherein the data indicia include an error correction feature without adding data bit to the first 2D code arrangement,

reading the first 2D code arrangement by ascertaining cluster sizes of the data indicia therein, and detecting the data indicia of the first 2D code arrangement based on a determination that each of the ascertained cluster sizes is larger than a predetermined minimum pixel size, wherein the predetermined pixel size is based on a pixel size of a potential error,

wherein at least a first one of the data indicia includes a plurality of separate portions wherein at least one of the separate portions has a cluster size larger than the predetermined pixel size;

detecting a reference indicium of the first 2D code arrangement of the data indicia;

ascertaining a plurality of distances between the detected data indicia and the reference indicium;

ascertaining a plurality of angular displacements of the detected data indicia relative to a reference direction, wherein the reference direction is based on a predetermined one of the detected data indicia or the reference indicium which is asymmetric or includes at most one line of symmetry; and

based on the ascertained distances, the ascertained angular displacements, and a plurality of bijective mappings between a plurality of combinations of code parameters and a plurality of predetermined data values, ascertaining some data values from the plurality of predetermined data values which correspond to the detected data indicia, wherein the code parameters include the distances and the angular displacements.

**[0013]** Embodiments of the fourth aspect are provided as recited in claim 10 to claim 13.

**[0014]** According to a fifth aspect of the invention, there is provided a non-transitory, computer readable medium comprising computer-executable instructions configured to direct at least one computer processor to perform the method according to any one of claim 9 to claim 13.

**[0015]** According to a sixth aspect of the invention, an apparatus for reading two-dimensional (2D) code is provided, wherein the apparatus comprises:

at least one memory for storing computer-executable instructions; and  
at least one processor communicably coupled to the at least memory and configured to: execute the computer-executable instructions to perform the method according to any one of claim 9 to claim 13.

**[0016]** According to a seventh aspect of the invention, an apparatus for reading 2D code is provided, wherein the apparatus comprises:

at least one processor;  
at least one memory;  
an image scanning module stored in the at least one memory and executable by the at least one processor to obtain an image of the 2D code of any one of claim 1 to claim 4;  
an image processing module stored in the at least one memory and executable by the at least one processor to:  
read the first 2D code arrangement by ascertaining cluster sizes of the data indicia therein, and detecting the data indicia of the first 2D code arrangement based on a determination that each of the ascertained cluster sizes is larger than a predetermined minimum pixel size, wherein the predetermined pixel size is based on a pixel size of a potential error.

**[0017]** Embodiments of the seventh aspect are provided as recited in claim 17.

## **BRIEF DESCRIPTION OF DRAWINGS**

**[0018]** Embodiments of the invention will be described in detail with reference to the accompanying drawings, in which the illustrated 2D codes do not limit or define the dimensions, shapes and/or proportions of 2D codes implemented in practice according to embodiments of the invention. The drawings include:

**[0019]** Figure 1A is an example of a 2D code having an error correction feature; Figure 1B illustrates the 2D code of Figure 1A which is overlaid with errors; Figure 1C illustrates distance and angular displacement measurements applied to Figure 1A;

**[0020]** Figure 2A is an example of a 2D code having an error correction feature; Figure 2B illustrates distance and angular displacement measurements applied to Figure 2A;

**[0021]** Figure 3A is an example of a 2D code having an error correction feature; Figure 3B illustrates distance and angular displacement measurements applied to Figure 3A;

**[0022]** Figure 4A is an example of a 2D code having an error correction feature; Figure 4B illustrates distance and angular displacement measurements applied to Figure 4A;

**[0023]** Figure 5A is an example of a 2D code having an error correction feature; Figure 5B illustrates distance and angular displacement measurements applied to Figure 5A;

**[0024]** Figure 6 is an example of a 2D code having an error correction feature;

**[0025]** Figure 7 is an example of a 2D code having an error correction feature;

**[0026]** Figure 8A is an example of a 2D code; Figure 8B illustrates distance and angular displacement measurements applied to Figure 8A;

[0027] Figure 9A is a simplified flow chart showing a method for generating 2D code;  
[0028] Figure 9B shows a non-limiting example of bijective mappings;  
[0029] Figure 9C is a schematic representation of an apparatus for generating 2D code;  
[0030] Figure 10A is a simplified flow chart showing a method for reading 2D code; and  
[0031] Figure 10B is a schematic representation of an apparatus for reading 2D code.

## **DETAILED DESCRIPTION**

[0032] In the following description, numerous specific details are set forth in order to provide a thorough understanding of various illustrative embodiments of the invention. It will be understood, however, to one skilled in the art, that embodiments of the invention may be practiced without some or all of these specific details. It is understood that the terminology used herein is for the purpose of describing particular embodiments only, and is not intended to limit the scope of the invention. In the drawings, like reference labels or numerals refer to same or similar functionalities or features throughout the several views. In the drawings, directional arrows shown between features illustrate data or information transfer therebetween in accordance with description of some embodiments but are not limited as such. In other words, data or information transfer in reverse to the directional arrows and/or not shown by directional arrow among features may be envisaged and are not shown to avoid obscuring description of the embodiments.

[0033] Embodiments described in the context of one of the apparatuses or methods are analogously valid for the other apparatuses or methods. Similarly, embodiments described in the context of an apparatus are analogously valid for a method, and vice versa.

[0034] Features that are described in the context of an embodiment or example may correspondingly be applicable to the same or similar features in the other embodiments or examples. Features that are described in the context of an embodiment or example may correspondingly be applicable to the other embodiments or examples, even if not explicitly described in these other embodiments or examples. Furthermore, additions and/or combinations and/or alternatives as described for a feature in the context of an embodiment or example may correspondingly be applicable to the same or similar feature in the other embodiments or examples.

[0035] It should be understood that the articles "a", "an" and "the" as used with regard to a feature or element include a reference to one or more of the features or elements. The term "and/or" includes any and all combinations of one or more of the associated feature or element. The terms "comprising", "including", "having", "involving" and any of their related terms, as used in description and claims, are intended to be open-ended and mean that there may be additional features or elements other than the listed ones. Identifiers such as "first", "second", "third", and so on, are used merely as labels, and are not intended to impose numerical requirements on their objects, nor construed in a manner imposing any relative position or time sequence between limitations.

**[0036]** The terms “indicium”, “indicia”, and related terms may include references to images, text, numbers, symbols, shapes, etc. The term “size” may include references to area and dimensions. The term “dimensions” may include references to length, width, or both. The term “coupled” and related terms may be used in an operational sense and/or to refer to a direct physical connection or coupling. Thus, for example, two devices may be coupled directly, or via one or more intermediary devices; in another example, two indicia may be coupled to each other, e.g. connect or contact with each other. Based on the present disclosure, a person of ordinary skill in the art will appreciate a variety of ways in which coupling exists in accordance with the aforementioned definition.

**[0037]** The invention provides a new type of error correction method or feature designed specifically for 2D code which has data representations which are non-binary and are symbology based.

**[0038]** In existing methods, e.g., *Downs, Adam S., N. P. Sigmon and Richard Klíma. “The Mathematics of QR Codes.” (2021)*, error correction methods have focused heavily on error bits, i.e., bits which have “flipped” as error. However, with symbology based 2D codes, the same error correction concept poses a challenge as the “flipping” of a symbol, which is made up of a plurality of bits, is vastly different from the “flipping” of a single bit.

**[0039]** To overcome this challenge, the invention uses the concept of 2D code symbology or relative size between a 2D code indicia cluster size and a detectable pixel size of a potential error as a new type of error correction method or feature.

**[0040]** By providing all 2D code symbology or 2D code indicia as being far larger than the size of the potential error, this will ensure readability of the 2D code image despite damage or errors in the original unprocessed raw image. This technique when deployed will also allow the 2D code to embody or incorporate an error correction method or feature without adding additional data bits or information, unlike existing error correction methods. This non-addition of bits is extremely important to preserving the original aesthetic design and integrity of symbology based 2D codes, without altering or distorting the symbology in any way, since these 2D codes may use corporate logos and/or trade characters as symbology for the 2D code.

**[0041]** On the other hand, using additional data bits for error correction may adversely affect the aesthetic design of the symbol used. An illustrative example is the leaf logo that is used in Figure 1A, adding additional bits to the logo will distort the original image.

**[0042]** The invention is configured to work if a percentage of the pixels is not correctly read and if the chance of affecting successive pixels decreases exponentially. In other words, the invention is premised on the principles that the chance of a cluster of pixels affected by a potential error decreases exponentially as the cluster size of affected pixels increases; and by ensuring that each data indicium used in the 2D code is of a certain minimum cluster size, it can be ensured that the chance that each data indicium being affected by the potential error is of a certain threshold.

## **Two-dimensional code having error correction feature**

**[0043]** Reference is made to Figure 1A illustrating a non-limiting example of a two-dimensional (2D) code according to some embodiments. The 2D code comprises at least one reference indicium, e.g. a position detection pattern having three blocks arranged at the bottom left, top left, and top right corners of the 2D code. Other non-limiting examples of a reference indicium include '-' symbol as illustrated in Figures 2A to 4B, 6 to 8B, etc., a frame as illustrated in Figures 5A, 5B. The 2D code further comprises at least one data indicium, e.g., a leaf symbol, a square symbol. If the 2D code includes a plurality of data indicia, they may be represented by distinct (or different) symbols and/or non-distinct (or same) symbols of distinct dimensions. The various indicia are distinct and optically readable, e.g., by camera, scanner.

**[0044]** Figure 1A shows a 2D code arrangement (alternatively referred to as a first 2D code arrangement) of data indicia to represent a data value (alternatively referred to as a first data value). More particularly, the illustrated positions of the illustrated data indicia, i.e., leaf symbol and the square symbol, relative to the illustrated reference indicium, i.e., position detection pattern, as shown collectively in Figure 1A represent a first data value. The data indicia are re-arrangeable into distinct (or different) positions to respectively provide distinct (or different) arrangements (alternatively referred to as other 2D code arrangements) which respectively represent distinct (or different) data values (alternatively referred to as other data values). In other words, a plurality or set of data values may be respectively represented by a plurality or set of unique 2D codes, and vice versa.

**[0045]** Reference is made to Figure 1B illustrating the 2D code of Figure 1A which is damaged, i.e. contains errors such as, but not limited to, household dust and/or stray mark(s). An error correction feature has been incorporated into the data indicia of the first 2D code arrangement without adding data bit to the 2D code. The error correction feature is implemented by providing a cluster size (or area or dimensions) of each data indicium larger than a typical or expected pixel size of a potential error which may be predetermined. In an embodiment, the cluster size of each data indicium may be larger than a predetermined pixel size of a potential error or larger than a predetermined factor, e.g. 105%, 110%, of the predetermined pixel size of a potential error. Optionally, in addition, the cluster size of each data indicium may be smaller than a predetermined maximum cluster size or smaller than a predetermined factor, e.g. 95%, 90%, of the maximum cluster size. The maximum cluster size may be ascertained based on an overall size of the 2D code, a required data capacity of the 2D code, and/or a count of indicia to be arranged within the 2D code.

**[0046]** In an illustrative example, where a 2D code has a size of 2 centimetres (cm) by 2 cm in which 40,000 pixels are contained, i.e. each pixel has a size of 100 microns ( $\mu\text{m}$ ) by 100  $\mu\text{m}$ . It may be assumed that each dust particle has a size of 100 microns ( $\mu\text{m}$ ) by 100  $\mu\text{m}$  and affects only 1 pixel with a 10% chance.

**[0047]** In an illustrative example, where a 2D code includes three indicia, e.g. Figure 1A, the smallest indicium (or a first indicium) is a line rectangle symbol; a second indicium is a leaf symbol which is approximately two times the size of the line rectangle symbol; the largest indicium (or a third indicium) is a square symbol which is approximately three times the size of the line rectangle symbol. For the line rectangle symbol to be misidentified as the leaf symbol, the size of error has to be as large as the line rectangle symbol, i.e. 1.5 times or more pixels are detected instead of 1 time pixels. For the leaf symbol to be misidentified as the largest indicium, the size of error has to be half as large as the line rectangle symbol, i.e. 2.5 times or more pixels are detected instead of 2 times pixels. Assuming the probability of the error being larger than half the line rectangle symbol is  $y\%$ , the maximum cluster size in this example is approximately three times the line rectangle symbol with each indicium having about  $y\%$  chance of being misidentified as other indicia.

**[0048]** A symbology set for data indicia of 2D codes provides a plurality of predetermined symbols with dimensions and layout which incorporate an error correction feature. Such data indicum symbol may be provided as a unitary symbol in which all portions of the symbol are connected. Non-limiting examples of unitary symbols include the illustrated leaf symbol, square symbol, and rectangle symbol. Alternatively or additionally, symbols may be provided as a non-unitary symbol such as having separate portions, at least one of the separate portions has a cluster size larger than the predetermined pixel size of a potential error or larger than a predetermined factor, e.g. 105%, 110%, of the predetermined pixel size of a potential error. Non-limiting examples of non-unitary symbols include alphabet "i", alphabet "j" wherein at least the lower portion of "i" or "j" has a cluster size larger than a predetermined pixel size while the upper portion, i.e. dot portion may have a cluster size smaller than the predetermined pixel size. Optionally, the cluster size of the aforesaid portion of may additionally be smaller than a predetermined maximum cluster size or smaller than a predetermined factor, e.g. 95%, 90%, of the maximum cluster size.

**[0049]** Reference is made to Figure 1C which is based on Figure 1A. Figure 1C illustrates distances  $D_1$ ,  $D_2$  between the reference indicium and the data indicia, and angular displacements  $\Theta_1$ ,  $\Theta_2$  of the data indicia which are relative to a reference direction R. Here, the reference direction R is ascertained based on a predetermined part of the reference indicia, e.g. reference direction is an orthogonal or vertical direction relative to the top left block; alternatively, the reference direction R may be taken from an unoccupied corner such by measuring and/or extrapolating from the occupied corners. Other methods of ascertaining the reference direction are also possible.

**[0050]** Variations and modifications to the 2D code of Figure 1A may be envisaged, such as but not limited to the examples of 2D codes in Figures 2A, 2B, 3A, 3B, 4A, 4B, 5A, 5B, 6, 7A, 7B which are described in International Application No. PCT/SG2023/050345 dated 19 May 2023 and entitled "Two-Dimensional Code And Method And System For Generating And Reading Thereof", the contents of which are incorporated by reference and provide a

symbology based 2D code system. For clarity, various principles of data representation by the 2D codes are explained with reference to these figures as follows.

**[0051]** Figure 2A illustrates an example of a 2D code comprising a reference indicium '-' and a data indicium '\*'. Figure 2B, based on Figure 2A, further illustrates a distance  $D_3$  between the reference indicium and the data indicium, and an angular displacement  $\Theta_3$  of the data indicium which is relative to a reference direction R. Here, the reference direction R is based on the reference indicium, e.g., reference direction R is an orthogonal or vertical direction relative to at least a predetermined part of the reference indicium.

**[0052]** Figure 3A illustrates an example of a 2D code which comprises a reference indicium '-', and a first and a second data indicium '\*' and '+' respectively. Figure 3B, based on Figure 3A, further illustrates a distance  $D_4$  between the reference indicium and the first data indicium, a distance  $D_5$  between the reference indicium and the second data indicium, an angular displacement  $\Theta_4$  of the first data indicium which is relative to a reference direction R, and an angular displacement  $\Theta_5$  of the second data indicium which is relative to the reference direction R. Here, the reference direction R is based on the reference indicium, e.g., reference direction is an orthogonal or vertical direction relative to at least a predetermined part of the reference indicium.

**[0053]** Figure 4A illustrates an example of a 2D code which comprises a reference indicium '-', and a first and a second data indicium '\*' and '+' respectively. Figure 4B, based on Figure 4A, further illustrates a distance  $D_6$  between the reference indicium and the first data indicium, a distance  $D_7$  between the reference indicium and the second data indicium, an angular displacement  $\Theta_6$  of the first data indicium which is relative to a reference direction. Here, the reference direction is based on the first or the second data indicium, e.g. line distances  $D_6$  and  $D_7$ .

**[0054]** Figure 5A illustrates an example of a 2D code which comprises a boundary indicium represented by a frame, a reference indicium represented by an indicator at the frame, e.g. top right corner of the frame, and a data indicium '\*'. Here, the indicator is represented by a triangle symbol. The reference indicium is coupled to the boundary indicium. Figure 5B, based on Figure 5A, further illustrates a distance  $D_8$  between a predetermined part of the reference indicium and the data indicium, and an angular displacement  $\Theta_8$  of the data indicium which is relative to a reference direction R. Here, the reference direction R is based on the reference indicium, e.g. the indicator at the upper right corner of the frame. It may be predetermined that this indicator provides the reference direction as a vertical direction as shown in Figure 5B. Alternatively, it may be predetermined that this indicator may define the reference direction differently, e.g. horizontal direction to the right of Figure 5B or horizontal direction to the left of Figure 5B. Accordingly, the angular displacement  $\Theta_8$  would be taken differently.

**[0055]** In Figure 5A, the reference indicium and boundary indicium are coupled, e.g. presented as an integral or connected image with parts of the reference indicium and boundary indicium in contact and/or overlap. It is to be appreciated that, in some other examples, the

reference indicium and boundary indicium may be uncoupled, e.g. presented as distinct or unconnected images with no parts in contact or overlap.

**[0056]** Figure 6 illustrates an example of a two-dimensional code which is similar to Figure 2A but further provides for the data indicium to include a scaled dimension relative to a reference size which may be predetermined. For example, the reference size may be based on typical dimension of the same data indicium in Figure 2A. Here, the data indicium '\*' in Figure 6 is two times the reference size, e.g. dimensions, of the data indicium '\*' in Figure 2A. But it is to be appreciated that, in some other examples, the scaled dimension may be larger or smaller than a reference size.

**[0057]** Figure 7 illustrates an example of a 2D code which comprises a reference indicium '-', and a first, a second, and a third data indicium 'O', '+', 'I' respectively. The first, the second, and third data indicium 'O', '+', 'I' are distinct and overlapping. Their overlapping position is meant to represent same data values, however, the third indicium 'I' is non-visible or non-optically distinguishable as it is overlaid by the second indicium '+', while the overlapping first and the second indicium 'O', '+' are visible and optically distinguishable. Hence, the first and the second data indicium 'O', '+' can be properly represented and read. This example illustrates that two or more distinct indicia may be provided at a same position to represent same data values while remaining visible and optically distinguishable, but this depends on the design of the indicia.

**[0058]** Figure 8A and 8B illustrate an example of a 2D code which comprises a reference indicium '-', and a first, a second, and a third data indicium 'O', '+', 'p' respectively. Similar to Figure 7, the first and the second data indicium 'O', '+' are distinct and overlapping which are meant to represent same data values. The third data indicium 'p' is positioned offset from the other data indicia and would have represented a different data value due to the different position. However, according to embodiments of the invention, the third data indicium may represent same data value as the first or the second indicium after predetermined positional adjustment which is based on a difference between the non-overlap positions, e.g. angular offset between the relevant data indicia. In other words, certain indicia, e.g. 'p', may be predefined as indicia which data value is to be adjusted in a predetermined manner in order to provide the intended data value.

**[0059]** In the example of Figure 7, it may not be possible to provide an optically distinguishable third data indicium which overlaps with the first and the second data indicium 'O', '+' at the same position, e.g. approximately 20 mm and 35 degrees relative to the reference indicium. In other words, three data indicia of same data values may not be possibly provided at the same position. However, the example of Figures 8A and 8B provides three data indicia of same data values by providing two data indicia at the same position, e.g. approximately 20 mm and  $\Theta_9$  of 35 degrees relative to the reference indicium, and further providing a data indicia 'p' at an offset, e.g. approximately 20 mm and  $\Theta_{10}$  of 50 degrees relative to the reference indicium, which upon applying a predetermined positional adjustment, e.g. an angular offset

based on a difference between  $\Theta_9$  and  $\Theta_{10}$ , would represent the same data value as the two data indicia.

**[0060]** It is to be appreciated that these non-limiting variations may be applied individually or in combination with any of these or other variations.

**[0061]** For example, the reference indicium and/or data indicium may be represented by other symbols or images, e.g. square, circle, rectangle, triangle, rhombus, pentagon, hexagon, octagon, cube, cone, diamond, star, arrow, animal, tree, plant, food, building, logo, mountain, tea pot, tea leaf, coffee bean, etc.

**[0062]** For example, the reference indicium may comprise one or more identical or non-identical symbols or images arranged in a predetermined layout.

**[0063]** For example, the boundary indicium may be represented by other shapes or forms, e.g. geometric shapes, non-geometric shapes, partially enclosed frame, an open frame, a combination of symbols or images such as the above-identified examples. For example, the boundary indicium may be represented by other outlines, e.g. solid lines, dotted lines, symbols or images such as the above-identified examples. For example, the boundary indicium may be equivalent to, i.e., indistinguishable from, the reference indicium. For example, the boundary indicium may be overlapping with but distinguishable from the reference indicium. For example, the boundary indicium may be non-overlapping with and distinguishable from the reference indicium. For example, the boundary indicium or part thereof may be omitted if it is not utilised as a reference indicium.

### **Generation of two-dimensional code**

**[0064]** Reference is made to Figure 9A which is a simplified flow chart showing a method for generating two-dimensional codes.

**[0065]** In block 91, code parameters may be determined. Examples of code parameters include, but are not limited to, a distance between a reference indicium and a data indicium, an angular displacement of the data indicium which is relative to a reference direction, a size of the data indicium relative to a reference size, a colour of the data indicium, etc.

**[0066]** In block 92, a set or range of data values may be determined. The set or range of data values (i.e. data value set) refers to the set of all possible data values that may be represented by a plurality of 2D code arrangements and may be determined depending on desired application.

**[0067]** In block 93, a minimum pixel size may be determined based on an identified potential error to provide a predetermined minimum pixel size. Optionally, a maximum cluster size may be determined to provide a predetermined maximum cluster size. Optionally, factor(s) of the predetermined minimum pixel size and/or predetermined maximum cluster size may be determined.

**[0068]** In block 94, a set or plurality of symbols for data indicia and the sizes of the symbols may be determined. The set of symbols for data indicia may refer to the set or plurality of all

possible symbols. The size of each symbol is based on the predetermined value(s) ascertained in block 93, thereby configuring each data indicium to incorporate an error correction feature or provide error correction.

**[0069]** In block 95, a plurality of bijective mappings between a plurality of combinations of code parameters and a plurality of data values may be ascertained. With bijective mappings, each combination of code parameters is paired with exactly one element of the set of data values; each element of the set of data values is paired with exactly one combination of code parameters.

**[0070]** As an illustration, for pairings between element set X and element set Y to be bijective, the four following properties must hold:

1. Each element of X must be paired with at least one element of Y,
2. No element of X may be paired with more than one element of Y,
3. Each element of Y must be paired with at least one element of X, and
4. No element of Y may be paired with more than one element of X.

**[0071]** A non-limiting example of bijective mappings is provided in Figure 9B. In this example, the code parameters include distance, angular displacement, size of data indicium; the data value set defines a range of possible target data values.

**[0072]** In block 96, a target data value is provided for a code to be generated therewith. The target data value is one of the values in the data value set.

**[0073]** In block 97, using the bijective mappings, a two-dimensional code having incorporated error correction feature, according to any embodiment in the present disclosure, is generated for the target data value. A plurality of other 2D codes (other 2D code arrangements) may also be generated for other target data values respectively.

**[0074]** Reference is made to Figure 9C which is a schematic representation of an apparatus for generating two-dimensional code having error correction feature. The apparatus comprises: at least one processor; at least one memory for storing computer-executable instructions and communicably coupled to the at least one processor; and various modules stored in the at least one memory and described as follows.

**[0075]** A data value set module may be stored in the at least one memory and executable by the at least one processor to determine a plurality of data values, e.g. set or range of data values, including a target data value.

**[0076]** A code parameter module may be stored in the at least one memory and executable by the at least one processor to determine a plurality of combinations of code parameters.

**[0077]** A mapping module may be stored in the at least one memory and executable by the at least one processor to ascertain a plurality of bijective mappings between a plurality of combinations of code parameters and a plurality of data values.

**[0078]** A symbology module may be stored in the at least one memory and executable by the at least one processor to provide a set or plurality of set of symbols of predetermined sizes

(which may be determined in block 94) configured to incorporate an error correction feature in data indicia or provide error correction to data indicia.

**[0079]** A code generation module may be stored in the at least one memory and executable by the at least one processor to generate, using the bijective mappings, a two-dimensional code according to any embodiment in the present disclosure may be generated for the target data value, wherein the target data value is the at least one data value.

**[0080]** The apparatus may further comprise network interface for facilitating communication between the apparatus and other communication, scan and/or computing devices, and input/output interface for receiving inputs from other communication and/or computing devices or manual inputs and for providing outputs to other communication, computing, display and/or print devices.

### **Reading of two-dimensional code**

**[0081]** Reference is made to Figure 10A which is a simplified flow chart showing a method for reading two-dimensional code which may be any one of the embodiments described in the present disclosure.

**[0082]** In block 101, an image of a 2D code is obtained or detected such as by an image scanning module, e.g. optical scanner or optical camera, which is executable by a processor. Based on the image of the 2D code, e.g., the first 2D code arrangement illustrated in Figure 1A, the 2D code is read. This reading includes ascertaining cluster sizes of the data indicia in the first 2D code arrangement, determining whether each of the ascertained cluster sizes of the data indicia in the first 2D code arrangement is larger than the predetermined minimum pixel size, and detecting these data indicia in the first 2D code arrangement based on a determination that each of their ascertained cluster sizes is larger than the predetermined minimum pixel size. The ascertaining and comparison step(s) may be performed by an image processing module which is executable by a processor.

**[0083]** In an example, where the data indicium is a medium rectangle indicium, the predetermined minimum pixel size is defined as 1401 moments, the predetermined maximum cluster size is defined as 2700 moments. An illustrative Python code for detecting a medium rectangle indicium which size is between 1401 moments and 2700 moments is set out below:

```
Open source Library CV2
# Classifying shapes
for contour in contours:
    M = cv.moments(contour)
    epsilon = 0.04 * cv.arcLength(contour, True)
    approx = cv.approxPolyDP(contour, epsilon, True)
    vertices = len(approx)
    if 2700 > M['m00'] > 1401:
        shape = "medium rectangle"
```

```

cv.drawContours(image, [approx], 0, (240, 95, 75), 2)
cv.putText(image, shape, (approx[0][0][0],
approx[0][0][1] + 5), cv.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 0), 2)
cx = int(M['m10'] / M['m00'])
cy = int(M['m01'] / M['m00'])

```

**[0084]** It is to be appreciated that moments are used in the above code snippet as an approximation of the number of pixels. It is to be appreciated that the other codes in Open Source Computer Vision (CV) library may be used to detect dimensions, including length and width, of symbols.

**[0085]** Reading of various data indicia in the 2D code (or first 2D code arrangement) may be performed at substantially the same time or sequentially, by an image scanning which is executable by a processor.

**[0086]** In block 102, at least one reference indicium is detected. This detection may be performed at substantially the same time as block 101, or sequentially.

**[0087]** In block 101 or block 102, as appropriate, distances between the reference indicium the detected data indicia are ascertained; angular displacements of the data indicia which are relative to a reference direction, is ascertained. The reference direction may be based on any data indicia or reference indicium, e.g., a first one of the indicia, which is predetermined and is asymmetric or includes at most one line of symmetry. The ascertaining step(s) may be performed by an image processing module which is executable by a processor.

**[0088]** In block 104, based on the ascertained distance, the ascertained angular displacement, and a plurality of bijective mappings between a plurality of combinations of code parameters and a plurality of data values (block 103), some data values from the plurality of data values which correspond to the at least one data indicium are ascertained. Here, the code parameters include the distance and the angular displacement. The ascertaining step(s) may be performed by a mapping module and a decoding module which are executable by a processor.

**[0089]** The above processes in blocks 101 to block 104 are performed or iterated *mutatis mutandis* in relation to other 2D code arrangements which respectively represent a plurality of other data values. In other words, at least some of the data indicia of the first 2D code arrangement are re-arranged relative to the data indicia of the first 2D code arrangement in order to represent other data values through these other 2D code arrangements, wherein the error correction feature is incorporated into the data indicia of these other 2D code arrangements without adding data bits to these other 2D code arrangements since the symbols used are taken from the same set of symbols. The re-arranged data indicia in these other 2D code arrangements are read as described in block 101. The reference indicium in each these other 2D code arrangements (other reference indicium) may be ascertained as described in block 102. Distances and angular displacements between the data indicia and the other reference indicium may be ascertained as described. Determination of data values

corresponding to these other 2D code arrangements may be ascertained as described in block 104.

**[0090]** In some embodiments, the first one of the indicia is the reference indicium or one of the at least one data indicium.

**[0091]** In some embodiments, the at least one data indicium includes a first data indicium and a second data indicium which are arranged at an overlap position. Accordingly, in block 104, the ascertaining step includes ascertaining, respectively, a first data value and a second data value as same data values.

**[0092]** In some embodiments, the at least one data indicium includes a first data indicium and a second data indicium which are arranged at non-overlap positions. Accordingly, in block 104, the ascertaining step includes ascertaining either the first data indicium or the second data indicium as having a predetermined positional adjustment, ascertaining a first data value and a second data value, respectively for the first data indicium or the second data indicium, as same data values. Here, the code parameters include predetermined positional adjustment based on offset between two predetermined indicia.

**[0093]** Reference is made to Figure 10B which is a schematic representation of an apparatus for reading two-dimensional code. The apparatus comprises: at least one processor; at least one memory for storing computer-executable instructions and communicably coupled to the at least one processor; and various modules stored in the at least one memory and described as follows.

**[0094]** An image scanning module may be stored in the at least one memory and executable by the at least one processor to obtain an image of a two-dimensional code which may be any one of the embodiments described in the present disclosure.

**[0095]** An image processing module may be stored in the at least one memory and executable by the at least one processor to perform image processing as described in block 101 and block 102.

**[0096]** A mapping module (table) may be stored in the at least one memory and having a plurality of bijective mappings between a plurality of combinations of code parameters and a plurality of data values.

**[0097]** A decoding module may be stored in the at least one memory and executable by the at least one processor to: based on the bijective mappings in the mapping module, ascertain or decode at least one data value of the plurality of data values which corresponds to the at least one data indicium processing, as described in block 103.

**[0098]** The apparatus may further comprise network interface for facilitating communication between the apparatus and other communication and/or computing devices, and input/output interface for receiving inputs from other communication, scan and/or computing devices or manual inputs and for providing outputs to other communication, computing, display and/or print devices.

**[0099]** Embodiments of the invention provide several advantages including but not limited to the following:

**[0100]** In *Downs, Adam S., N. P. Sigmon and Richard Klíma. "The Mathematics of QR Codes." (2021)*, error correction methods have focused heavily on error bits, i.e., bits which have "flipped" as error. Section 3, describes an example QR code which originally requires 800 bits to present the data in full for the code, but when the QR code incorporates Reed-Solomon code, it requires additional 0s. The example QR of version 4 which incorporates Reed-Solomon code requires seven additional 0s; thus a total of 807 bits is now required for the data section of the example QR code instead of the original 800 bits.

**[0101]** In contrast, the invention does not rely on error bits or "flipped" bits, nor require or add extra data bits each time a different data value is represented. This is advantageous as adding extra data bits may alter or distort the symbology-based 2D code, e.g. alter or distort the data indicia symbols. In certain applications where preservation of the integrity of the symbols is crucial, e.g. corporate logo and/or trade character, an altered or distorted data indicia symbol would be highly undesirable.

**[0102]** As determination of cluster sizes of data indicia symbols is dependent on pixel size of potential errors, the 2D code would still likely be machine-readable if there is slight damage to the 2D code.

**[0103]** The invention may be implemented using code from Open Source CV2 library that allows for the error correction method to be performed computationally efficiently. The code details a method of detecting a cluster of pixels between a range of approximated moments. This is advantageous as a 2D code needs to achieve certain criteria of scanning speed in order to be adopted commercially.

**[0104]** The invention utilises simpler mathematical functions and hence processing the raw 2D code image will be more computationally efficient than other error correction methods that uses complex mathematical functions. This is advantageous in reducing the scanning speed and enhancing the efficiency of scanning devices.

**[0105]** The error tolerance of the invention can be adjusted to similar tolerance levels of other error correction codes by adjusting the cluster size relative to dust, stray marks, etc. A larger data indicia size relative to potential error size provides a better error tolerance.

**[0106]** It is to be appreciated that the flow charts showing logic flows are representative of exemplary methodologies for performing novel aspects of the invention. While, for purposes of simplicity of explanation, the one or more methodologies shown herein are shown and described as a series of acts, those skilled in the art will understand and appreciate that the methodologies are not limited by the order of acts. Some acts may, in accordance therewith, occur in a different order and/or concurrently with other acts from that shown and described herein. For example, those skilled in the art will understand and appreciate that a methodology could alternatively be represented as a series of interrelated states or events, such as in a

state diagram. Moreover, not all acts illustrated in a methodology may be required for a novel implementation.

**[0107]** The flow charts showing logic flow may be implemented in software, firmware, hardware, or any combination thereof. In software and firmware embodiments, the logic flow may be implemented by computer executable instructions or code stored on a non-transitory computer readable medium or machine readable medium, such as an optical, magnetic or semiconductor storage. The computer-executable instructions are configured to direct at least one computer processor to perform the logic flow. The embodiments are not limited in this context.

**[0108]** It is to be appreciated that the apparatuses described herein are representative of exemplary apparatuses for performing novel aspects of the invention. Those skilled in the art will understand and appreciate that the apparatuses are not limited by their elements described herein. In any of the apparatuses described herein, it may comprise at least one memory and at least one processor communicably coupled thereto. The at least one processor may be any type of computer processor, such as a microprocessor, an embedded processor, a digital signal processor (DSP), a network processor, a multi-core processor, a single core processor, or other device configured to execute code or computer-executable instructions to perform or implement the flowcharts, algorithms, processes, or operations detailed herein. The at least one memory may be a non-transitory computer readable medium or machine readable medium for storing or comprising the code or computer-executable instructions. Examples include, but are not limited to, random access memory (RAM), read only memory (ROM), logic blocks of a field programmable gate array (FPGA), erasable programmable read only memory (EPROM), and electrically erasable programmable ROM (EEPROM).

**[0109]** It is to be understood that the embodiments and features described above should be considered exemplary and not restrictive. Many other embodiments will be apparent to those skilled in the art from consideration of the specification and practice of the invention. Furthermore, certain terminology has been used for the purposes of descriptive clarity, and not to limit the disclosed embodiments of the invention.

## CLAIMS

1. A two-dimensional (2D) code comprising:
  - a plurality of optically readable and distinct data indicia, being different symbols, arranged as a first 2D code arrangement of the data indicia which represents a first data value;
  - an error correction feature which is incorporated into the data indicia of the first 2D code arrangement without adding data bit to the first 2D code arrangement,
  - wherein the error correction feature is based on a cluster size of each data indicium being larger than a predetermined pixel size, wherein the predetermined pixel size is based on a pixel size of a potential error,
  - wherein at least a first one of the data indicia includes a plurality of separate portions wherein at least one of the separate portions has a cluster size larger than the predetermined pixel size; and
  - a reference indicium arranged within the first 2D code arrangement, wherein the first data value is represented by a plurality of distances between the reference indicium and the data indicia of the first 2D code arrangement, and by a plurality of angular displacements of the data indicia of the first 2D code arrangement which are relative to a reference direction, wherein the reference direction is based on a predetermined one of the data indicia or the reference indicium which is asymmetric or includes at most one line of symmetry.
2. The 2D code of claim 1,
  - wherein the error correction feature is further based on the cluster size of each data indicium being smaller than a predetermined maximum cluster size.
3. The 2D code of claim 1 or claim 2, wherein the potential error includes dust and/or stray mark.
4. The 2D code of any one of claim 1 to claim 3, wherein the data indicia are configured to be re-arranged as a plurality of other 2D code arrangements of the data indicia which respectively represent a plurality of other data values, and wherein the error correction feature is incorporated into the data indicia of the other 2D code arrangements without adding data bits to the other 2D code arrangements.
5. A plurality of two-dimensional (2D) codes comprising:
  - a plurality of 2D code arrangements of optically readable and distinct data indicia, being different symbols, wherein the 2D code arrangements of data indicia respectively represent a plurality of data values;

an error correction feature which is incorporated into the data indicia without adding data bit to each of the 2D code arrangements,

wherein the error correction feature is based on a cluster size of each of the data indicia being larger than a predetermined pixel size, wherein the predetermined pixel size is based on a pixel size of a potential error,

wherein at least one of the data indicia includes a plurality of separate portions wherein at least one of the separate portions has a cluster size larger than the predetermined pixel size; and

a reference indicium arranged within each of the 2D code arrangements, wherein the data value of each of the 2D code arrangements is represented by distances between the reference indicium and the data indicia of the each of the 2D code arrangements, and by angular displacements of the data indicium of the each of the 2D code arrangements which are relative to a reference direction, wherein the reference direction is based on a predetermined one of the data indicia or the reference indicium which is asymmetric or includes at most one line of symmetry.

6. The plurality of 2D codes of claim 5, wherein the error correction feature is further based on the cluster size of each data indicium being smaller than a predetermined cluster size which is determined based on a count of the data indicia in each arrangement, a size of the 2D codes, and/or a data capacity of the 2D codes.

7. The plurality of 2D codes of claim 5 or claim 6, wherein the potential error includes dust and/or stray mark.

8. A two-dimensional (2D) code generation method comprising:

providing the 2D code according to any one of claim 1 to claim 4, including:

based on a plurality of bijective mappings between a plurality of combinations of code parameters and a plurality of data values to provide the first 2D code arrangement, generating the 2D code for a target data value being the first data value, wherein the code parameters include the distances and the angular displacements.

9. A two-dimensional (2D) code reading method, the method comprising:

based on a first 2D code arrangement of a plurality of optically readable and distinct data indicia, being different symbols, wherein the first 2D code arrangement of the data indicia represents a first data value, wherein the data indicia include an error correction feature without adding data bit to the first 2D code arrangement,

reading the first 2D code arrangement by ascertaining cluster sizes of the data indicia therein, and detecting the data indicia of the first 2D code arrangement based on a determination that each of the ascertained cluster sizes is larger than a predetermined minimum pixel size, wherein the predetermined pixel size is based on a pixel size of a potential error,

wherein at least a first one of the data indicia includes a plurality of separate portions wherein at least one of the separate portions has a cluster size larger than the predetermined pixel size;

detecting a reference indicium of the first 2D code arrangement of the data indicia;

ascertaining a plurality of distances between the detected data indicia and the reference indicium;

ascertaining a plurality of angular displacements of the detected data indicia relative to a reference direction, wherein the reference direction is based on a predetermined one of the detected data indicia or the reference indicium which is asymmetric or includes at most one line of symmetry; and

based on the ascertained distances, the ascertained angular displacements, and a plurality of bijective mappings between a plurality of combinations of code parameters and a plurality of predetermined data values, ascertaining some data values from the plurality of predetermined data values which correspond to the detected data indicia, wherein the code parameters include the distances and the angular displacements.

10. The method of claim 9, wherein the error correction feature is further based on the cluster size of each data indicium being smaller than a predetermined maximum cluster size.

11. The method of claim 9 or claim 10, wherein the potential error includes dust and/or stray mark.

12. The method of any one of claim 9 to claim 11, based on a plurality of other 2D code arrangements of the data indicia, wherein the data indicia are re-arranged relative to the first 2D code arrangement, wherein the other 2D code arrangements of the data indicia respectively represent a plurality of other data values, wherein the data indicia include the error correction feature without adding data bit to the other data values or information to the other 2D code arrangements,

reading each of the other 2D code arrangements by ascertaining cluster sizes of the data indicia therein, and detecting the data indicia of the other 2D code arrangements based on a determination that the ascertained cluster sizes are larger than the predetermined minimum cluster size.

13. The method of claim 12, further comprising:  
for each of the other 2D code arrangements:  
detecting an other reference indicium of the each other 2D code arrangement;  
ascertaining a plurality of other distances between the detected data indicia and the other reference indicium;  
ascertaining a plurality of other angular displacements of the detected data indicia relative to the other reference direction, wherein the other reference direction is based on a predetermined one of the detected data indicia or the other reference indicium which is asymmetric or includes at most one line of symmetry; and  
based on the ascertained other distances, the ascertained other angular displacements, and the plurality of bijective mappings between the plurality of combinations of code parameters and the plurality of predetermined data values, ascertaining some other data values from the plurality of predetermined data values which correspond to the detected data indicia, wherein the code parameters include the other distances and the other angular displacements.
14. A non-transitory, computer readable medium comprising computer-executable instructions configured to direct at least one computer processor to perform the method according to any one of claim 9 to claim 13.
15. An apparatus for reading two-dimensional (2D) code, the apparatus comprising:  
at least one memory for storing computer-executable instructions; and  
at least one processor communicably coupled to the at least memory and configured to: execute the computer-executable instructions to perform the method according to any one of claim 9 to claim 13.
16. An apparatus for reading two-dimensional (2D) code, the apparatus comprising:  
at least one processor;  
at least one memory;  
an image scanning module stored in the at least one memory and executable by the at least one processor to obtain an image of the 2D code of any one of claim 1 to claim 4;  
an image processing module stored in the at least one memory and executable by the at least one processor to:  
read the first 2D code arrangement by ascertaining cluster sizes of the data indicia therein, and detecting the data indicia of the first 2D code arrangement based on a determination that each of the ascertained cluster sizes is larger than a predetermined

minimum pixel size, wherein the predetermined pixel size is based on a pixel size of a potential error.

17. The apparatus of claim 16, wherein the image processing module stored in the at least one memory and executable by the at least one processor to:

- detecting a reference indicium of the first 2D code arrangement of the data indicia;

- ascertaining a plurality of distances between the detected data indicia and the reference indicium;

- ascertaining a plurality of angular displacements of the detected data indicia relative to a reference direction, wherein the reference direction is based on a predetermined one of the detected data indicia or the reference indicium which is asymmetric or includes at most one line of symmetry;

wherein the apparatus further comprises:

- a mapping module stored in the at least one memory and having a plurality of bijective mappings between a plurality of combinations of code parameters and a plurality of data values; and

- a decoding module stored in the at least one memory and executable by the at least one processor to:

- based on the ascertained distances, the ascertained angular displacements, and a plurality of bijective mappings between a plurality of combinations of code parameters and a plurality of predetermined data values, ascertaining some data values from the plurality of predetermined data values which correspond to the detected data indicia, wherein the code parameters include the distances and the angular displacements.