



- (51) International Patent Classification:
G06T 15/00 (2011.01)
- (21) International Application Number:
PCT/CN2013/077797
- (22) International Filing Date:
24 June 2013 (24.06.2013)
- (25) Filing Language: English
- (26) Publication Language: English
- (71) Applicant: **ADOBE SYSTEMS INCORPORATED**
[US/US]; 345 Park Avenue, San Jose, California 95110-2704 (US).
- (72) Inventor: **GOU, Dongfang**; 19/F, SP Tower B, Tsinghua Science Park, No. 1 Zhong Guan Cun East Road, Haidian District, Beijing 100084 (CN).
- (74) Agent: **KING & WOOD MALLESONS LAWYERS**; 20th Floor, East Tower, World Financial Center, No. 1 Dongsanhuan Zhonglu, Chaoyang District, Beijing 100020 (CN).
- (81) Designated States (*unless otherwise indicated, for every kind of national protection available*): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY,

BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

- (84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, 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).

Declarations under Rule 4.17:

— *of inventorship (Rule 4.17(iv))*

Published:

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

(54) Title: GRAVITY POINT DRAWING METHOD

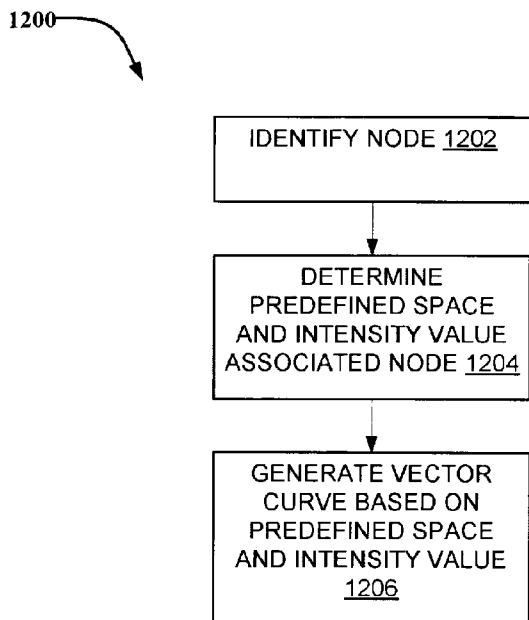


FIG. 12

(57) Abstract: A system and method of a gravity point drawing that simulate a physical model to draw or modify vector curves. A node is identified in a drawing of a graphical application. A size of a predefined space around the node and an intensity value associated with the node are determined. The intensity value affects the curvature of a vector curve in the predefined space. The vector curve is generated in the predefined space. The curvature of the vector curve is based on the size of the predefined space and the intensity value associated with the node in the drawing.



GRAVITY POINT DRAWING METHOD

TECHNICAL FIELD

5 [0001] The present application relates generally to the technical field of computer-generated drawings, and, in various embodiments, to systems and methods of gravity point drawing.

BACKGROUND

10 [0002] Computer graphics applications can generate vector curves that can be manipulated to modify a shape of the curve. Bézier curves are parametric curves commonly used in computer graphics that provide a method to transfer geometric curves into a programming model. In vector graphics, Bézier curves are used to model smooth curves that can be scaled indefinitely using anchors and handlers throughout the curves. Bézier curves are commonly defined in
15 terms of four points. Two of these points are the endpoints, also referred to as anchor points. The other two points are known as control points, also referred to as handles, one associated with each anchor point. The curve extends from anchor point to anchor point. The line between an anchor point and its control
20 point determines the slope, or derivative, of the curve at the anchor point. The further the control point is from the anchor point, the further the curve wants to go along the straight line before curving off towards the other anchor point.

[0003] As such, a user would have to manipulate multiple anchor points and control points in order to change the shape of the curves of a geometric
25 shape. For example, FIG. 1A illustrates an original computer graphic shape 100. FIG. 1B illustrates a target computer graphic shape 102. FIG. 1C illustrates the original computer graphic shape 100 with anchor points 104 and control points 106 that the user would have to manipulate, one by one, to modify the curvature of the original computer graphic shape 100. Because each anchor point and
30 control point has to be manually moved in a one by one fashion, inconsistency can be accumulated during the modification of the curvature. Further adjustment and correction of the anchor points and control points may be needed to generate the target computer graphic shape 102.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] Some embodiments of the present disclosure are illustrated by way of example and not limitation in the figures of the accompanying drawings, in which like reference numbers indicate similar elements and in which:

- 5 [0005] FIG. 1A illustrates a prior art example of an original geometric shape to be modified;
- [0006] FIG. 1B illustrates a prior art example of a target geometric shape.
- [0007] FIG. 1C illustrates a prior art example of anchor points and handlers to be manipulated.
- 10 [0008] FIG. 2A illustrates an example embodiment of a modification process of a vector curve.
- [0009] FIG. 2B illustrates another example embodiment of a modification process of a vector curve.
- [00010] FIG. 3A illustrates an example embodiment of a model for a ray
- 15 mode modification process of a vector curve.
- [00011] FIG. 3B illustrates another example embodiment of a model for a ray mode modification process of a vector curve.
- [00012] FIG. 3C illustrates an example embodiment of an algorithm for the ray mode modification process of the vector curve of FIG. 3B.
- 20 [00013] FIG. 4A illustrates an example embodiment of a model for a ray mode modification process of a vector curve.
- [00014] FIG. 4B illustrates an example embodiment of an algorithm for a ray mode modification process of the vector curve of FIG. 4A.
- [00015] FIG. 5 illustrates an example embodiment of a sequence for
- 25 generating vector curves using a ray mode modification process.
- [00016] FIG. 6A illustrates an example embodiment of a model for an arch mode modification process of a new vector curve.
- [00017] FIG. 6B illustrates an example embodiment of an algorithm for an arch mode modification process of the vector curve of FIG. 6A.
- 30 [00018] FIG. 7A illustrates another example embodiment of a model for an arch mode modification process of a new vector curve.
- [00019] FIG. 7B illustrates an example embodiment of an algorithm for an arch mode modification process of the vector curve of FIG. 7A.

[00020] FIGS. 8A-8F illustrate example embodiments of a model for an arch mode modification process of an existing vector curve.

[00021] FIGS. 9A-9C illustrate example embodiments of a model for an expand mode modification process of a new vector curve.

5 [00022] FIG. 10 illustrates a block diagram of an example embodiment of a graphical application.

[00023] FIG. 11 illustrates a block diagram of an example embodiment of a vector curve generator.

[00024] FIG. 12 is a flowchart illustrating an example embodiment of a
10 method for generating a vector curve.

[00025] FIG. 13 is a flowchart illustrating another example embodiment of a method for generating a vector curve.

[00026] FIG. 14 shows a diagrammatic representation of a machine in the example form of a computer system within which a set of instructions may be
15 executed to cause the machine to perform any one or more of the methodologies discussed herein.

DETAILED DESCRIPTION

[00027] The description that follows includes illustrative systems,
20 methods, techniques, instruction sequences, and computing machine program products that embody illustrative embodiments. In the following description, for purposes of explanation, numerous specific details are set forth in order to provide an understanding of various embodiments of the inventive subject matter. It will be evident, however, to those skilled in the art that embodiments of the
25 inventive subject matter may be practiced without these specific details. In general, well-known instruction instances, protocols, structures, and techniques have not been shown in detail.

[00028] The present disclosure describes a system and method of a gravity point drawing that simulates a physical model to draw and modify vector curves.
30 In an example embodiment, a node is identified in a drawing of a graphical application. The size of a predefined space around the node and an intensity value associated with the node are determined. The intensity value affects the curvature of a vector curve in the predefined space. The vector curve is generated in the predefined space. The curvature of the vector curve is based on

the size of the predefined space and the intensity value associated with the node in the drawing.

[00029] In another example embodiment, the identification of the node selected by a user of the graphical application is received. The size of the predefined space around the node and the intensity value associated with the node are adjusted to modify the shape of the vector curve in the predefined space. The curvature of the vector curve is adjusted in the predefined space in the drawing based on the adjusted size of the predefined space and the adjusted intensity value associated with the node. The predefined space comprises a region with the node as the center of the region.

[00030] In another example embodiment, a vector curve is generated using a physical model vector curve. This physical model vector curve has a curvature based on the intensity value associated with the node and represents the measure of gravitational forces from the node in the drawing in a ray mode.

[00031] In another example embodiment, a vector curve is generated relative to a line segment between a first intersection point found between the existing curve and a perimeter of the predefined space and a second intersection point found between the existing curve and the perimeter of the predefined space in an arch mode.

[00032] In another example embodiment, points of an existing curve are shifted in the predefined space along an axis perpendicular to a line segment between a first intersection point found between the existing curve and a perimeter of the predefined space and a second intersection point found between the existing curve and the perimeter of the predefined space in an arch mode.

[00033] In another example embodiment, points of an existing curve are shifted in the predefined space along a radial axis of the node in an expand mode.

[00034] In another example embodiment, the vector curve is part of a curve in the drawing. The vector curve comprises a portion of the curve in the predefined space. The node in the drawing does not affect portions of the curve outside the predefined space.

[00035] In another example embodiment, the intensity value associated with the node is increased to increase the curvature of the vector curve towards the node in the predefined space. The intensity value associated with the node is decreased to decrease the curvature of the vector curve towards the node in the

predefined space. The curvature of the vector curve is adjusted away from the node in the predefined space when the intensity value is negative.

[00036] In another example embodiment, a first node and a second node are identified in the drawing. A first size of a first predefined space around the first node and a first intensity value associated with the first node are determined. The first intensity value affects the curvature of a first vector curve in the first predefined space. A second size of a second predefined space around the second node and a second intensity value associated with the second node are determined. The second intensity value affects the curvature of a second vector curve in the second predefined space. The first predefined space intersects the second predefined space to form a common predefined space. A common vector curve is generated in the common predefined space. The curvature of the common vector curve is based on the size of the first and second predefined spaces and the intensity value associated with the first and second nodes.

[00037] In another example embodiment, a system comprises a memory coupled to a processor. The memory stores a graphical application. The processor comprises a node identifier, a node attribute module, and a vector curve generator. The node identifier identifies a node in a drawing of the graphical application. The node attribute module determines a size of a predefined space around the node and an intensity value associated with the node. The intensity value affects the curvature of a vector curve in the predefined space. The vector curve generator generates the vector curve in the predefined space. The curvature of the vector curve is based on the size of the predefined space and the intensity value associated with the node.

[00038] In another example embodiment, the node identifier is further configured to receive the identification of the node selected by a user of the graphical application, to adjust both the size of the predefined space around the node and the intensity value associated with the node needed to modify the shape of the vector curve in the predefined space, and to adjust the curvature of the vector curve in the predefined space in the drawing which is based on the adjusted size of the predefined space and the adjusted intensity value associated with the node.

[00039] In another example embodiment, the vector curve generator comprises a ray mode module configured to generate a physical model vector

curve having a curvature based on the intensity value associated with the node as a representation of a measure of gravitational force from the node in the drawing.

[00040] In another example embodiment, the vector curve generator comprises an arch mode module configured to generate the vector curve relative to a line segment between a first intersection point found between the existing curve and a perimeter of the predefined space and a second intersection point found between the existing curve and the perimeter of the predefined space.

[00041] In another example embodiment, the vector curve generator comprises an arch mode module configured to shift points of an existing curve in the predefined space along an axis perpendicular to a line segment between a first intersection point found between the existing curve and a perimeter of the predefined space and a second intersection point found between the existing curve and the perimeter of the predefined space.

[00042] In another example embodiment, the vector curve generator comprises an arch mode module configured to shift points of an existing curve in the predefined space along a radial axis of the node.

[00043] In another example embodiment, the node attribute module is configured to increase the intensity value associated with the node that increases a curvature of the vector curve towards the node in the predefined space, to decrease the intensity value associated with the node that decreases a curvature of the vector curve towards the node in the predefined space. The vector curve generator is configured to adjust the curvature of the vector curve away from the node in the predefined space when the intensity value is negative.

[00044] In another example embodiment, the node identifier is configured to identify a first node and a second node in the drawing. The node attribute module is configured to determine a first size of a first predefined space around the first node, and a first intensity value associated with the first node. The first intensity value affects a curvature of a first vector curve in the first predefined space. The node attribute model is also configured to determine a second size of a second predefined space around the second node, and a second intensity value associated with the second node. The second intensity value affects a curvature of a second vector curve in the second predefined space. The first predefined space intersects the second predefined space to form a common predefined space. The vector curve generator is configured to generate a common vector curve in

the common predefined space. A curvature of the common vector curve is based on the sizes of the first and second predefined spaces and the intensity value associated with the first and second nodes in the drawing.

[00045] In another example embodiment, a non-transitory machine-readable storage device may store a set of instructions that, when executed by at least one processor, causes the at least one processor to perform the method operations discussed within the present disclosure.

[00046] FIG. 2A illustrates an example embodiment of a ray mode modification process of a vector curve. The ray mode includes a single fixed point on straight line and is based on classical physics models. For example, an object 202 orbits a gravitational point (e.g., gravity point 204). As the object 202 moves closer to the gravity point 204, it may be affected. The object 202 may change its direction slowly. The path of the object 202 generates a smooth curve 206.

[00047] The variables, mass and distance, can affect the curvature of the path. For example, increasing the mass of the gravity point 204 may generate a sharp curve 208 for the object 202.

[00048] Inversely, reducing the relative distance between the object 202 and the gravity point 204 may generate an even sharper curve 210 as illustrated in FIG.2B. As such, two example parameters in the gravity point drawing method used to control the curves include the intensity of the gravity point 204 and the relative distance between the object 202 and the gravity point 204.

[00049] In a physical model, the affected range of gravity force is unlimited. In other words, as the distance between the object 202 and the gravity point 204 increases, the gravity force between the object 202 and the gravity point 204 approaches zero. In an example embodiment of the gravity point drawing method, a parameter is generated to limit the affected range. FIG. 3A illustrates a predefined space 302 that can be adjusted to limit the effect of the gravity point 204 on the object 202. As such, the path of the object 202 is not affected outside the predefined space 302. In one embodiment, the predefined space 302 comprises a region with the gravity point 204 as the center of the region. Similarly, the path of the object 202 is affected by the gravity point 204 only within the predefined space 302.

[00050] FIG. 3B illustrates a model for a ray mode modification process

of a vector curve. When the object 202 moves into the predefined space 302 at geographic location (P_x, P_y) , the object 202 starts to gain acceleration by gravity forces (G_x, G_y) from the gravity point 204. The acceleration of the object 202 at geographic location (P_x, P_y) may be separated into two directional components: a
5 first direction component a_x along the original velocity, and a second direction a_y component vertical to the first direction component a_x . The vertical acceleration a_y changes the path of the object 202. As long as the object 202 stays within the predefined space 302, the path of the object 202 generates a curve. The curvature may be affected by parameters as further described below.

10 [00051] FIG. 3C illustrates an example embodiment of an algorithm for the ray mode modification process of the vector curve of FIG. 3B. An example of the ray mode modification process of a vector curve is illustrated in FIG. 3C where the square of the mass “m” of the object 202 makes the acceleration non-linear. Increasing the mass “m” of the object 202 adds acceleration “a” evenly.
15 Reducing the relative distance “d” between the object 202 and the gravity point 204 adds different acceleration based on the position of each point along the path of the object 202.

[00052] The object 202 may move into an area covered by multiple gravity points. For example, FIG. 4A illustrates a model for a ray mode
20 modification process of a vector curve for the object 202. The object 202 is affected by two gravity points 204 and 402 when the object 202 is in the corresponding predefined space 302 and 404 of gravity points 204 and 402. The acceleration (a and a') from each gravity point 204 and 402 is broken down into horizontal direction (a_x and a_x') and vertical direction (a_y and a_y'). The
25 composite direction is then calculated based on the combined original directions (a_x and a_x') and vertical direction (a_y and a_y').

[00053] In another embodiment, the intensity associated with the gravity point 204 may be reversed. This reverse gravity point method may be referred to as an inverse option. For example, the distribution of gravity force can be
30 inversed inside the predefined space 302. In other words, points with more distance from the gravity point 204 receive more gravity force than points closer to the gravity point 204. FIG. 4B illustrates an example embodiment of an algorithm for the reverse gravity point method.

[00054] FIG. 5 illustrates an example embodiment of a sequence 500 for

generating vector curves using a ray mode modification process as previously described. In drawing 502, three straight lines 503 are drawn. In drawing 504, gravity points 510 and 512 are placed within their respective predefined spaces 514, 516. The shape or area of the predefined spaces 514, 516 can be adjusted to
5 change the affected range. The mass of (or intensity associated with) the gravity points 510 and 512 may also be adjusted. For example, a positive mass would represent attractive force while a negative mass would represent a repulsive force. Drawing 506 illustrates the effects of the gravity point drawing method on the three lines 503. Drawing 508 illustrates an example of an end result of a
10 computer graphics drawing using the gravity point drawing method.

[00055] In contrast to the traditional anchors and handles based methods, the gravity point drawing method is easier to use. Adjusting one gravity point (e.g., 510, 512) in the gravity point drawing method would be the equivalent of adjusting multiple anchor points and handlers as illustrated in the example of
15 FIG. 1C. Furthermore, the relative curvature of the paths is consistently maintained between the three lines 503 in FIG. 5 when adjusting the gravity points 510 and 512.

[00056] In accordance with another example embodiment, the gravity point drawing method includes an arch mode. FIG. 6A illustrates an example of
20 an arch mode where two points P_0 and P_1 are fixed on a single straight line 602. The portion 604 of the straight line 602 is within a predefined space 606 and is therefore affected by a gravity point 608 associated with the predefined space 606.

[00057] In the arch mode, the acceleration as previously described is not a
25 parameter in this case. The arch mode produces a vector curve 610 defined by three points: P_0 , P_1 , and P_2 . P_0 and P_1 are two points of intersection between the straight line 602 and the predefined space 606 which are associated with the gravity point 608. P_2 is a point along the vector curve 610 closest to the gravity point 608. In other words, P_0 , P_1 , and P_2 make up a Quadratic Bézier curve.

30 FIG. 6B illustrates an example embodiment of an algorithm for the arch mode modification process of the vector curve of FIG. 6A where P_{2x} and P_{2y} represent geographic coordinates of the point P_2 , $B(t)$ represents the offset relative to the portion 604 of the straight line 602 to generate the vector curve 610.

[00058] FIG. 7A illustrates another example embodiment of a model for an arch mode modification process of a new vector curve 710. The arch mode produces a new vector curve 710 defined by three points: P_0 , P_1 , and P_2 . FIG. 7A illustrates an example of an arch mode where two points P_0 and P_1 are fixed on a single straight line 702. The portion 704 of the line 702 between P_0 and P_1 is within a predefined space 706 and is, therefore, affected by a gravity point 708 associated with the predefined space 706. Because points P_0 and P_1 are user-defined and do not correspond to intersection points between the straight line 702 and the predefined space 706, the arch mode in FIG. 7A illustrates an asymmetric mode where the intersection points P_0 and P_1 are not defined by the predefined space 706, and the horizontal movement of the gravity point 708 distorts the arch shape of the new vector curve 710 between intersection points P_0 and P_1 .

[00059] FIG. 7B illustrates an example embodiment of an algorithm for the arch mode modification process of the vector curve 710 of FIG. 7A where P_{2x} and P_{2y} represent geographic coordinates of the point P_2 , $B(t)$ represents the offset relative to the portion 704 of the straight line 702 to generate the vector curve 710.

[00060] FIGS. 8A-8E illustrate example embodiments of a model for an arch mode modification process for existing vector curves. FIG. 8A illustrates an example of a vector curve 802 with multiple anchor points P_0 , P_1 , P_2 , P_3 , and P_4 , and respective handles 804, 806, 808, 810, and 812. In order to change the shape of the vector curve 802, anchor points P_0 , P_1 , P_2 , P_3 , P_4 , and respective handles 804, 806, 808, 810, and 812 would all have to be manipulated in a traditional method.

[00061] FIG. 8B illustrates where the example vector curve 802 intersects a predefined space 814 between two intersection points: P_5 and P_6 . The two intersection points P_5 and P_6 are added on the vector curve 802 as new anchor points P_0 , P_1 , P_2 , P_3 , and P_4 . A straight line 816 between the two intersection points P_5 and P_6 forms the direction of a horizontal line. Anchor points P_0 , P_1 , P_2 , P_3 , and P_4 are located inside the predefined space 814.

[00062] FIG. 8C illustrates how the relative horizontal distances between each anchor point to the gravity point 818 associated with the predefined space 814 decides the weight of vertical movement. For example, anchor point P_1 has

a horizontal distance d1 from the gravity point 818. Anchor point P₂ has a horizontal distance d2 from the gravity point 818. Anchor point P₃ has a horizontal distance d3 from the gravity point 818. Because the horizontal distance d1 is greater than the horizontal distance d2, the vertical displacement of anchor point P₂ is greater than the vertical displacement of anchor point P₁. Similarly, because the horizontal distance d3 is greater than the horizontal distance d2, the vertical displacement of anchor point P₂ is still greater than the vertical displacement of anchor point P₃.

[00063] FIG. 8D illustrates how the relative horizontal distance between each control point to the gravity point 818 decides the weight of vertical movement. For example, control point 820 has a horizontal distance a1 from the gravity point 818. Control point 822 has a horizontal distance a2 from the gravity point 818. Because the horizontal distance a1 is greater than the horizontal distance a2, the vertical displacement of control point 822 is much greater than the vertical displacement of control point 820.

[00064] FIG. 8E illustrates a mathematical model for vertical displacement of points along the existing vector curve 802. The maximum vertical displacement of point P2 is represented by MaxP2x and MaxP2y. The minimum vertical displacement of point P2 is represented by MinP2x and MinP2y.

[00065] The original Bezier curve may be represented as:

$$B(t)=(P_0, P_1, P_2 \dots P_{n-2}, P_{n-1}, P_n).$$

[00066] The intersection points may be represented by P_{n+1}, P_{n+2}.

[00067] The direction of the horizontal line may be represented as:

$$\text{tag}(a): (P_{n+1x} - P_{n+2x}) / (P_{n+1y} - P_{n+2y})$$

[00068] The anchor points and control points vertical movement may be represented as:

$$\text{New } P_{m_x} = P_{m_x} - (P_{m_x} - \min P_{m_x}) * (m^2 * 100\%) \text{ when } m > 0$$

$$\text{New } P_{m_x} = P_{m_x} - (\text{Max } P_{m_x} - P_{m_x}) * (m^2 * 100\%) \text{ when } m < 0$$

$$\text{New } P_{m_y} = P_{m_y} - (P_{m_y} - \min P_{m_y}) * (m^2 * 100\%) \text{ when } m > 0$$

$$\text{New } P_{m_y} = P_{m_y} - (\text{Max } P_{m_y} - P_{m_y}) * (m^2 * 100\%) \text{ when } m < 0$$

where P_m represents an amount of vertical displacement of a point P along the vector curve 802.

[00069] In this arch mode, the vector curve 802 is bent away or closer to

the horizontal direction line as previously defined. However, the curvature of the original vector curve 802 will be preserved since the vertical displacement is adjusted respectively along each point of the vector curve 802 in the predefined space 814.

5 [00070] FIG. 8F illustrates the original vector curve 802 modified using the arch mode of the gravity point method drawing to generate a second curve 824.

[00071] FIG. 9A illustrates an example embodiment of a model for an expand mode modification process of an existing vector curve. The expand mode is similar to the previously described arch mode. Similarly, an existing curve 902 intersects with a predefined space 906 associated with a gravity point 904. The existing curve 902 intersects the edge of the predefined space 906 at two intersection points: P5 and P6. The two points P5 and P6 are added to the existing curve 902 as new anchor points.

15 [00072] In the present expand mode, there is no horizontal line. Instead, the respective distance between the anchor points and control points inside the predefined space 906 and the gravity point 904 decide the amount of radial displacement. In this mode, the displacement movement is toward or away from the gravity point 904 along a radial axis of the gravity point 904.

20 [00073] For example, anchor point P1 has a radial distance of d1 from the gravity point 904. Anchor point P2 has a radial distance of d2 from the gravity point 904 and anchor point P3 has a radial distance of d3 from the gravity point 904. Because the radial distance d2 of P2 is less than the radial distance d1 of P1, the displacement of P2 is greater than P1.

25 [00074] FIG. 9B illustrates a mathematical model for radial displacement of points of the existing curve 902. The maximum vertical displacement of point P2 is represented by MaxP2x and MaxP2y. The minimum vertical displacement of point P2 is represented by MinP2x and MinP2y.

[00075] The original Bezier curve may be represented as:

30 $B(t)=(P_0, P_1, P_2 \dots P_{n-2}, P_{n-1}, P_n).$

[00076] The intersection points may be represented by $P_{n+1}, P_{n+2}.$

[00077] The direction movement (towards or away from the gravity point) may be represented as:

$\text{tag}(a): (G_x - P_{m_x}) / (G_y - P_{m_y})$

[00078] The anchor points and control points radial displacements may be represented as:

$$\text{New } P_{m_x} = P_{m_x} - (P_{m_x} - \min P_{m_x}) * (m^2 * 100\%) \text{ when } m > 0$$

$$\text{New } P_{m_x} = P_{m_x} - (\text{Max} P_{m_x} - P_{m_x}) * (m^2 * 100\%) \text{ when } m < 0$$

5 $\text{New } P_{m_y} = P_{m_y} - (P_{m_y} - \min P_{m_y}) * (m^2 * 100\%) \text{ when } m > 0$

$$\text{New } P_{m_y} = P_{m_y} - (\text{Max} P_{m_y} - P_{m_y}) * (m^2 * 100\%) \text{ when } m < 0$$

where P_m represents an amount of radial displacement of a point P along the existing curve 902.

[00079] FIG. 9C illustrates the existing curve 902 modified using the arch mode of the gravity point drawing method to generate a second curve 908. In 10 this mode, the distribution of anchor points may be changed. This mode contracts points and curves towards the gravity point 904 or expand them to the edge of the predefined space 906.

[00080] As such, by changing the size of the predefined space, a user can 15 choose which part of a curve needs to receive gravity force. Furthermore, by moving a gravity point, a user can choose which part of the curve receives effects.

[00081] FIG. 10 illustrates a block diagram of an example embodiment of a graphical application 1000 used to implement the gravity point drawing 20 method as previously described. The graphical application 1000 may be used to generate new vector curves and modify existing vector curves in computer generated drawings.

[00082] In one embodiment, the graphical application 1000 comprises a gravity point drawing module 1002, a node identifier module 1004, a node 25 attribute module 1006, and a vector curve generator 1008.

[00083] The node identifier module 1004 identifies a node in a drawing of the graphical application 1000. For example, the node identifier module 1004 identifies the gravity point 204 of FIG. 3B as the node.

[00084] The node attribute module 1006 determines a size of a predefined 30 space around the node and an intensity value associated with the node. The intensity value affects a curvature of a vector curve in the predefined space. For example, the node attribute module 1006 determines the size of the predefined space 302 around the gravity point 204 in FIG. 3B. The predefined space 302 may be a two-dimensional area or a three-dimensional space. In another

embodiment, the predefined space 302 may be a circle or sphere or any user-defined enclosed shape. In other words, the predefined space 302 does not have to be a circle, but can also include other shapes such as a box, a rectangle, or any user-drawn shapes. The intensity value may include the gravitational value associated with the gravity point 204 (e.g., G_x and G_y) in FIG. 3B.

[00085] The vector curve generator 1008 generates the vector curve in the predefined space (e.g., 302). The curvature of the vector curve may be based on the size of the predefined space (e.g., 302) and the intensity value associated with the node. For example, the curvature of the vector curve 206 is based on the size of the predefined space 302 and the gravitational value associated with the gravity point 204 (e.g., G_x and G_y) in FIG. 3B.

[00086] FIG. 11 illustrates a block diagram of an example embodiment of the vector curve generator 1008. The vector curve generator 1008 comprises a ray mode module 1102, an arch mode module 1104, and an expand mode module 1106.

[00087] The ray mode module 1102 may generate a physical model vector curve having a curvature based on the intensity value associated with the node as a representation of a measure of gravitational force from the node in the drawing. For example, the curvature of the vector curve 206 is based on the gravitational value of the gravity point 204 and the size of the predefined space 302 in FIG. 3A.

[00088] The arch mode module 1104 may generate the vector curve relative to a line segment between a first intersection point found between the existing curve and a perimeter of the predefined space and a second intersection point found between the existing curve and the perimeter of the predefined space. For example, the arch mode module 1104 generates the vector curve 610 relative to portion of the straight-line segment 604 in FIG. 6A.

[00089] The expand mode module 1106 may shift points of an existing curve in the predefined space along an axis perpendicular to a line segment between a first intersection point found between the existing curve and a perimeter of the predefined space and a second intersection point found between the existing curve and the perimeter of the predefined space. For example, the expand mode module 1104 generates the second curve 824 in FIG. 8F.

[00090] The expand mode module 1106 may also shift points of an

existing curve in the predefined space along a radial axis of the node. For example, the expand mode module 1106 generates the second curve 908 in FIG. 9C.

[00091] FIG. 12 is a flowchart illustrating an example embodiment of a method 1200 for generating a vector curve. The operations of method 1200 may be performed by a computer system of the graphical application 1000 or modules of the graphical application 1000. At operation 1202, a node is identified in a drawing of a graphical application 1000. The node includes the gravitational point as previously described. In one embodiment, the node identifier module 1004 of FIG. 10 performs this operation.

[00092] At operation 1204, the size of a predefined space around the node and an intensity value associated with the node are determined. The intensity value affects the curvature of a vector curve in the predefined space. In one embodiment, the node attribute module 1006 of FIG. 10 performs this operation.

[00093] At operation 1206, the vector curve is generated in the predefined space. The curvature of the vector curve is based on the size of the predefined space and the intensity value associated with the node in the drawing. In one embodiment, the vector curve generator 1008 of FIG. 10 performs this operation.

[00094] FIG. 13 is a flowchart illustrating another example embodiment of a method for generating a vector curve. At operation 1302, an identification of the node selected by a user of the graphical application is received. In one embodiment, the node identifier module 1004 of FIG. 10 performs this operation.

[00095] At operation 1304, the size of the predefined space around the node and the intensity value associated with the node are adjusted to modify a shape of the vector curve in the predefined space. In one embodiment, the vector curve generator 1008 of FIG. 10 performs this operation.

[00096] At operation 1306, the curvature of the vector curve is adjusted in the predefined space in the drawing based on the adjusted size of the predefined space and the adjusted intensity value associated with the node. The predefined space comprises a region with the node as a center of the region. In one embodiment, the vector curve generator 1008 of FIG. 10 performs this operation.

MODULES, COMPONENTS AND LOGIC

[00097] Certain embodiments are described herein as including logic or a number of components, modules, or mechanisms. Modules may constitute either software modules (e.g., code embodied on a machine-readable medium or in a transmission signal) or hardware modules. A hardware module is a tangible unit capable of performing certain operations and may be configured or arranged in a certain manner. In example embodiments, one or more computer systems (e.g., a standalone, client, or server computer system) or one or more hardware modules of a computer system (e.g., a processor or a group of processors) may be configured by software (e.g., an application or application portion) as a hardware module that operates to perform certain operations as described herein.

[00098] In various embodiments, a hardware module may be implemented mechanically or electronically. For example, a hardware module may comprise dedicated circuitry or logic that is permanently configured (e.g., as a special-purpose processor, such as a field programmable gate array (FPGA) or an application-specific integrated circuit (ASIC)) to perform certain operations. A hardware module may also comprise programmable logic or circuitry (e.g., as encompassed within a general-purpose processor or other programmable processor) that is temporarily configured by software to perform certain operations. It will be appreciated that the decision to implement a hardware module mechanically, in dedicated and permanently configured circuitry, or in temporarily configured circuitry (e.g., configured by software) may be driven by cost and time considerations.

[00099] Accordingly, the term “hardware module” should be understood to encompass a tangible entity, be that an entity that is physically constructed, permanently configured (e.g., hardwired) or temporarily configured (e.g., programmed) to operate in a certain manner and/or to perform certain operations described herein. Considering embodiments in which hardware modules are temporarily configured (e.g., programmed), each of the hardware modules need not be configured or instantiated at any one instance in time. For example, where the hardware modules comprise a general-purpose processor configured using software, the general-purpose processor may be configured as respective different hardware modules at different times. For example, software may accordingly configure a processor to constitute a particular hardware module at

one instance of time and to constitute a different hardware module at a different instance of time.

[000100] Hardware modules can provide information to, and receive information from, other hardware modules. Accordingly, the described hardware modules may be regarded as being communicatively coupled. Where multiple of such hardware modules exist contemporaneously, communications may be achieved through signal transmission (e.g., over appropriate circuits and buses) that connect the hardware modules. In embodiments in which multiple hardware modules are configured or instantiated at different times, communications between such hardware modules may be achieved, for example, through the storage and retrieval of information in memory structures to which the multiple hardware modules have access. For example, one hardware module may perform an operation and store the output of that operation in a memory device to which it is communicatively coupled. A further hardware module may then, at a later time, access the memory device to retrieve and process the stored output. Hardware modules may also initiate communications with input or output devices and can operate on a resource (e.g., a collection of information).

[000101] The various operations of example methods described herein may be performed, at least partially, by one or more processors that are temporarily configured (e.g., by software) or permanently configured to perform the relevant operations. Whether temporarily or permanently configured, such processors may constitute processor-implemented modules that operate to perform one or more operations or functions. The modules referred to herein may, in some example embodiments, comprise processor-implemented modules.

[000102] Similarly, the methods described herein may be at least partially processor-implemented. For example, at least some of the operations of a method may be performed by one or more processors or processor-implemented modules. The performance of certain of the operations may be distributed among the one or more processors, not only residing within a single machine, but deployed across a number of machines. In some example embodiments, the processor or processors may be located in a single location (e.g., within a home environment, an office environment or as a server farm), while in other embodiments the processors may be distributed across a number of locations.

[000103] The one or more processors may also operate to support performance of the relevant operations in a “cloud computing” environment or as a “software as a service” (SaaS). For example, at least some of the operations may be performed by a group of computers (as examples of machines including
5 processors), these operations being accessible via a network and via one or more appropriate interfaces (e.g., APIs).

ELECTRONIC APPARATUS AND SYSTEM

[000104] Example embodiments may be implemented in digital electronic
10 circuitry or in computer hardware, firmware, software, or in combinations of them. Example embodiments may be implemented using a computer program product, (e.g., a computer program tangibly embodied in an information carrier, e.g., in a machine-readable medium for execution by, or to control the operation of, data processing apparatus, e.g., a programmable processor, a computer, or
15 multiple computers).

[000105] A computer program can be written in any form of programming language, including compiled or interpreted languages, and it can be deployed in any form, including as a stand-alone program or as a module, subroutine, or other unit suitable for use in a computing environment. A computer program can
20 be deployed to be executed on one computer or on multiple computers at one site or distributed across multiple sites and interconnected by a communication network.

[000106] In example embodiments, operations may be performed by one or more programmable processors executing a computer program to perform
25 functions by operating on input data and generating output. Method operations can also be performed by, and apparatus of example embodiments may be implemented as, special purpose logic circuitry (e.g., a FPGA or an ASIC).

[000107] A computing system can include clients and servers. A client and a server are generally remote from each other and typically interact through a
30 communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other. In embodiments deploying a programmable computing system, it will be appreciated that both hardware and software architectures merit consideration. Specifically, it will be appreciated that the

choice of whether to implement certain functionality in permanently configured hardware (e.g., an ASIC), in temporarily configured hardware (e.g., a combination of software and a programmable processor), or a combination of permanently and temporarily configured hardware may be a design choice.

- 5 Below, are set out hardware (e.g., machine) and software architectures that may be deployed in various example embodiments.

EXAMPLE MACHINE ARCHITECTURE AND MACHINE-READABLE MEDIUM

- 10 [000108] FIG. 14 is a block diagram of a machine in the example form of a computer system 1400 within which instructions (e.g., 1424) for causing the machine to perform any one or more of the methodologies discussed herein may be executed. In alternative embodiments, the machine operates as a standalone device or may be connected (e.g., networked) to other machines. In a networked
15 deployment, the machine may operate in the capacity of a server or a client machine in a server-client network environment, or as a peer machine in a peer-to-peer (or distributed) network environment. The machine may be a personal computer (PC), a tablet PC, a set-top box (STB), a Personal Digital Assistant (PDA), a cellular telephone, a web appliance, a network router, switch or bridge,
20 or any machine capable of executing instructions (e.g., 1424- sequential or otherwise) that specify actions to be taken by that machine. Further, while only a single machine is illustrated, the term “machine” shall also be taken to include any collection of machines that individually or jointly execute a set (or multiple sets) of instructions (e.g., 1424) to perform any one or more of the
25 methodologies discussed herein.

- [000109] The example computer system 1400 includes a processor 1402 (e.g., a central processing unit (CPU), a graphics processing unit (GPU) or both), a main memory 1404 and a static memory 1406, which communicate with each other via a bus 1408. The computer system 1400 may further include a video
30 display unit 1410 (e.g., a liquid crystal display (LCD) or a cathode ray tube (CRT)). The computer system 1400 also includes an alphanumeric input device 1412 (e.g., a keyboard), a user interface (UI) navigation (or cursor control) device 1414 (e.g., a mouse), a disk drive unit 1416, a signal generation device 1418 (e.g., a speaker) and a network interface device 1420.

MACHINE-READABLE MEDIUM

[000110] The disk drive unit 1416 includes a machine-readable medium 1422 on which is stored one or more sets of data structures and instructions 1424 (e.g., software) embodying or utilized by any one or more of the methodologies or functions described herein. The instructions 1424 may also reside, completely or at least partially, within the main memory 1404 and/or within the processor 1402 during execution thereof by the computer system 1400, the main memory 1404 and the processor 1402 also constituting machine-readable media. The instructions 1424 may also reside, completely or at least partially, within the static memory 1406.

[000111] While the machine-readable medium 1422 is shown in an example embodiment to be a single medium, the term "machine-readable medium" may include a single medium or multiple media (e.g., a centralized or distributed database, and/or associated caches and servers) that store the one or more instructions 1424 or data structures. The term "machine-readable medium" shall also be taken to include any tangible medium that is capable of storing, encoding or carrying instructions (e.g., 1424) for execution by the machine and that cause the machine to perform any one or more of the methodologies of the present embodiments, or that is capable of storing, encoding or carrying data structures utilized by or associated with such instructions (e.g., 1424). The term "machine-readable medium" shall accordingly be taken to include, but not be limited to, solid-state memories, and optical and magnetic media. Specific examples of machine-readable media include non-volatile memory, including by way of example semiconductor memory devices (e.g., Erasable Programmable Read-Only Memory (EPROM), Electrically Erasable Programmable Read-Only Memory (EEPROM), and flash memory devices); magnetic disks such as internal hard disks and removable disks; magneto-optical disks; compact disc-read-only memory (CD-ROM) and digital versatile disc (or digital video disc) read-only memory (DVD-ROM) disks.

TRANSMISSION MEDIUM

[000112] The instructions 1424 may further be transmitted or received over a communications network 1426 using a transmission medium. The instructions

1424 may be transmitted using the network interface device 1420 and any one of a number of well-known transfer protocols (e.g., HTTP). Examples of communication networks 1426 include a LAN, a WAN, the Internet, mobile telephone networks, POTS networks, and wireless data networks (e.g., WiFi and 5 WiMax networks). The term "transmission medium" shall be taken to include any intangible medium capable of storing, encoding, or carrying instructions (e.g., 1424) for execution by the machine, and includes digital or analog communication signals or other intangible media to facilitate communication of such software.

10 [000113] Although an embodiment has been described with reference to specific example embodiments, it will be evident that various modifications and changes may be made to these embodiments without departing from the broader spirit and scope of the present disclosure. Accordingly, the specification and drawings are to be regarded in an illustrative rather than a restrictive sense. The 15 accompanying drawings that form a part hereof, show by way of illustration and not of limitation, specific embodiments in which the subject matter may be practiced. The embodiments illustrated are described in sufficient detail to enable those skilled in the art to practice the teachings disclosed herein. Other embodiments may be utilized and derived therefrom, such that structural and 20 logical substitutions and changes may be made without departing from the scope of this disclosure. Therefore, this Detailed Description is not to be taken in a limiting sense, and the scope of various embodiments is defined only by the appended claims along with the full range of equivalents to which such claims are entitled.

25 [000114] Such embodiments of the inventive subject matter may be referred to herein, individually and/or collectively, by the term "invention" merely for convenience and without intending to voluntarily limit the scope of this application to any single invention or inventive concept if more than one is in fact disclosed. Although specific embodiments have been illustrated and 30 described herein, it should be appreciated that any arrangement calculated to achieve the same purpose may be substituted for the specific embodiments shown. This disclosure is intended to cover any and all adaptations or variations of several embodiments. Combinations of the above embodiments and other

embodiments not specifically described herein, will be apparent to those of skill in the art upon reviewing the above description.

[000115] The Abstract of the Disclosure is provided to comply with 37 C.F.R. §1.72(b), requiring an abstract that will allow the reader to quickly
5 ascertain the nature of the technical disclosure. It is submitted with the understanding that it will not be used to interpret or limit the scope or meaning of the claims. In addition, in the foregoing Detailed Description, it can be seen that various features are grouped together in a single embodiment for the purpose of streamlining the disclosure. This method of disclosure is not to be
10 interpreted as reflecting an intention that the claimed embodiments require more features than are expressly recited in each claim. Rather, as the following claims reflect, the inventive subject matter lies in less than all features of a single disclosed embodiment. Thus, the following claims are hereby incorporated into the Detailed Description with each claim standing on its own as a separate
15 embodiment.

WHAT IS CLAIMED IS:

1. A method comprising:
identifying a node in a drawing of a graphical application embodied in a
5 processor of a machine;
determining a size of a predefined space around the node and an intensity
value associated with the node, the intensity value affecting a curvature of a
vector curve in the predefined space; and
generating the vector curve in the predefined space, a curvature of the
10 vector curve based on the size of the predefined space and the intensity value
associated with the node in the drawing.
2. The method of claim 1, further comprising:
receiving an identification of the node selected by a user of the graphical
15 application;
adjusting the size of the predefined space around the node and the
intensity value associated with the node to modify a shape of the vector curve in
the predefined space; and
adjusting the curvature of the vector curve in the predefined space in the
20 drawing based on the adjusted size of the predefined space and the adjusted
intensity value associated with the node.
3. The method of claim 1, wherein the predefined space comprises a region
with the node as a center of the region.
25
4. The method of claim 1, further comprising:
generating the vector curve using a physical model vector curve having a
curvature based on the intensity value associated with the node as a
representation of a measure of gravitational force from the node in the drawing
30 in a ray mode.
5. The method of claim 4, further comprising:
generating the vector curve relative to a line segment between a first
intersection point found between an existing curve and a perimeter of the

predefined space and a second intersection point found between the existing curve and the perimeter of the predefined space in an arch mode.

6. The method of claim 4, further comprising:
- 5 shifting points of an existing curve in a predefined space along an axis perpendicular to a line segment between a first intersection point found between the existing curve and a perimeter of the predefined space and a second intersection point found between the existing curve and the perimeter of the predefined space in an arch mode.
- 10
7. The method of claim 4, further comprising:
- shifting points of an existing curve in a predefined space along a radial axis of the node in an expand mode.
- 15
8. The method of claim 4, wherein the vector curve is part of a curve in a drawing, the vector curve comprising a portion of the curve in a predefined space, portions of the curve outside the predefined space are not affected by the node in the drawing.
- 20
9. The method of claim 1, further comprising:
- increasing the intensity value associated with the node to increase the curvature of the vector curve towards the node in the predefined space;
- decreasing the intensity value associated with the node to decrease the curvature of the vector curve towards the node in the predefined space; and
- 25 adjusting the curvature of the vector curve away from the node in the predefined space when the intensity value is negative.
10. The method of claim 1, further comprising:
- identifying a first node and a second node in the drawing;
- 30 determining a first size of a first predefined space around the first node and a first intensity value associated with the first node, the first intensity value affecting a curvature of a first vector curve in the first predefined space;
- determining a second size of a second predefined space around the second node and a second intensity value associated with the second node, the

second intensity value affecting a curvature of a second vector curve in the second predefined space, the first predefined space intersecting the second predefined space to form a common predefined space; and

5 generating a common vector curve in the common predefined space, a curvature of the common vector curve based on the size of the first and second predefined spaces and the intensity value associated with the first and second nodes.

11. A system comprising:
10 a memory configured to store a graphical application; and
a processor coupled to the memory, the processor comprising:
a node identifier configured to identify a node in a drawing of the graphical application;
a node attribute module configured to determine a size of a
15 predefined space around the node and an intensity value associated with the node, the intensity value affecting a curvature of a vector curve in the predefined space; and
a vector curve generator configured to generate the vector curve
20 in the predefined space, the curvature of the vector curve based on the size of the predefined space and the intensity value associated with the node.

12. The system of claim 11, wherein the node identifier is further configured to:
25 receive an identification of the node selected by a user of the graphical application;
adjust the size of the predefined space around the node and the intensity value associated with the node to modify a shape of the vector curve in the predefined space; and
30 adjust the curvature of the vector curve in the predefined space in the drawing based on the adjusted size of the predefined space and the adjusted intensity value associated with the node.

13. The system of claim 11, wherein the predefined space comprises a region with the node as a center of the region.
14. The system of claim 11, wherein the vector curve generator comprises a
5 ray mode module configured to generate a physical model vector curve having a curvature based on the intensity value associated with the node as a representation of a measure of gravitational force from the node in the drawing.
15. The system of claim 14, wherein the vector curve generator comprises an
10 arch mode module configured to generate a vector curve relative to a line segment between a first intersection point found between an existing curve and a perimeter of a predefined space and a second intersection point found between the existing curve and the perimeter of the predefined space.
16. The system of claim 14, wherein the vector curve generator comprises an
15 arch mode module configured to shift points of an existing curve in a predefined space along an axis perpendicular to a line segment between a first intersection point found between the existing curve and a perimeter of the predefined space and a second intersection point found between the existing curve and the
20 perimeter of the predefined space.
17. The system of claim 14, wherein the vector curve generator comprises an
25 arch mode module configured to shift points of an existing curve in a predefined space along a radial axis of the node.
18. The system of claim 11, wherein the vector curve generator is configured to:
- increase the intensity value associated with the node to increase the curvature of the vector curve towards the node in the predefined space;
 - 30 decrease the intensity value associated with the node to decrease the curvature of the vector curve towards the node in the predefined space; and
 - adjust the curvature of the vector curve away from the node in the predefined space when the intensity value is negative.

19. The system of claim 11, wherein the node identifier is configured to identify a first node and a second node in the drawing,

wherein the node attribute module is configured to determine a first size of a first predefined space around the first node and a first intensity value
5 associated with the first node, the first intensity value affecting a curvature of a first vector curve in the first predefined space, to determine a second size of a second predefined space around the second node and a second intensity value associated with the second node, the second intensity value affecting a curvature of a second vector curve in the second predefined space, the first predefined
10 space intersecting the second predefined space to form a common predefined space, and

wherein the vector curve generator is configured to generate a common vector curve in the common predefined space, a curvature of the common vector curve based on the size of the first and second predefined spaces and the
15 intensity value associated with the first and second nodes in the drawing.

20. A non-transitory machine-readable storage medium storing instructions which, when executed by at least one processor, performs operations comprising:

identifying a node in a drawing of a graphical application;
20 determining a size of a predefined space around the node, and an intensity value associated with the node, the intensity value affecting a curvature of a vector curve in the predefined space; and

generating the vector curve in the predefined space, the curvature of the vector curve based on the size of the predefined space and the intensity value
25 associated with the node in the drawing.

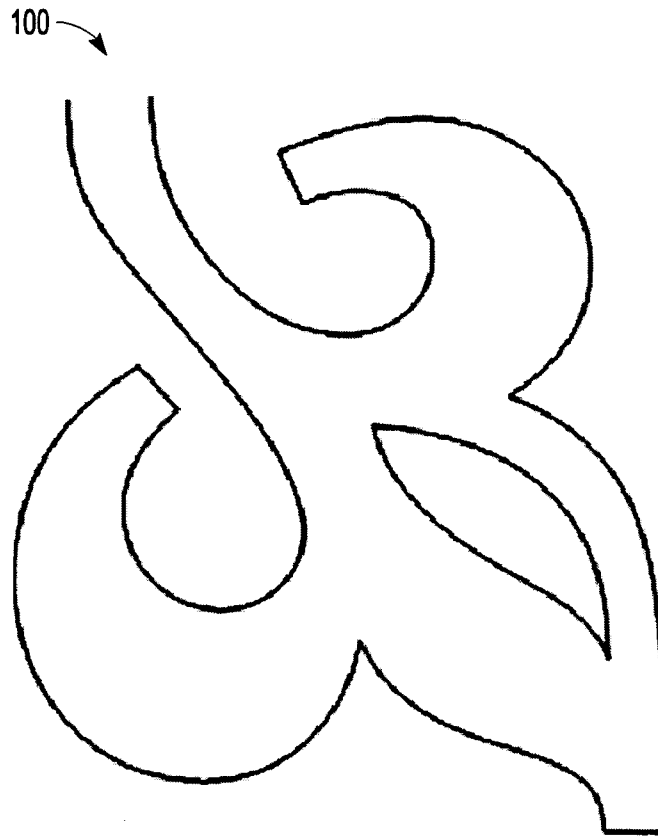


FIG. 1A

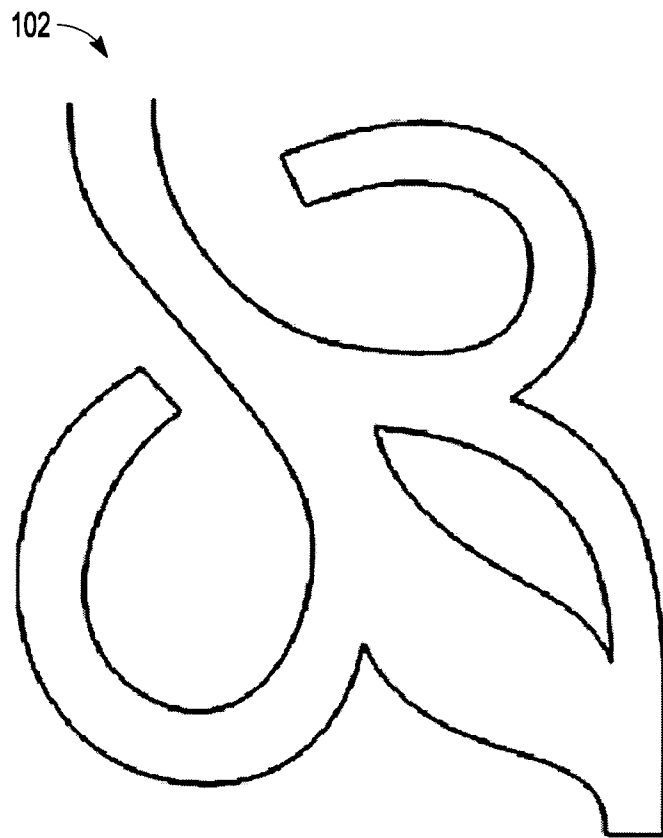


FIG. 1B

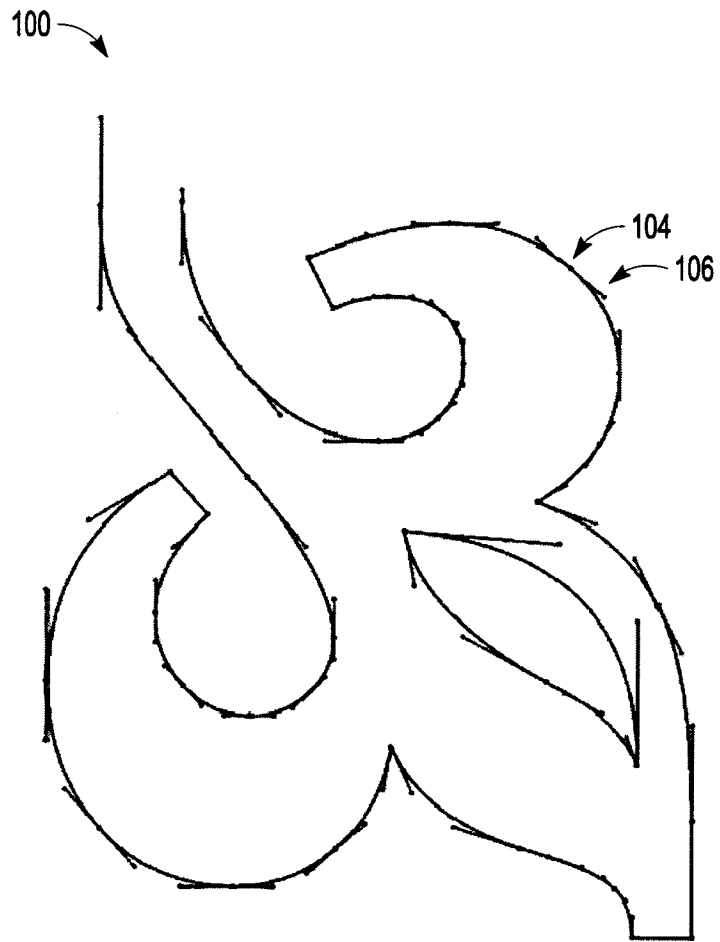


FIG. 1C
(PRIOR ART)

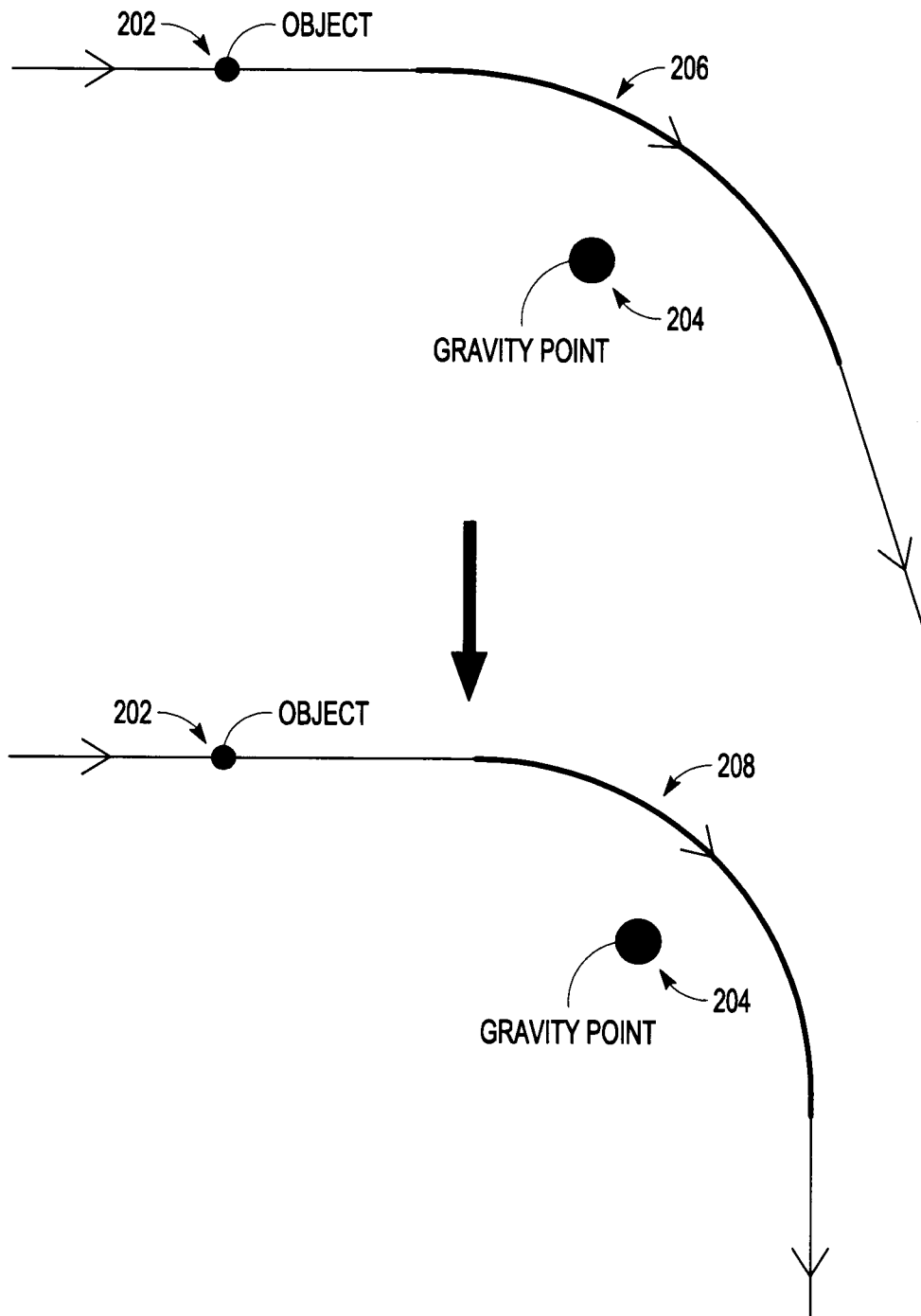


FIG. 2A

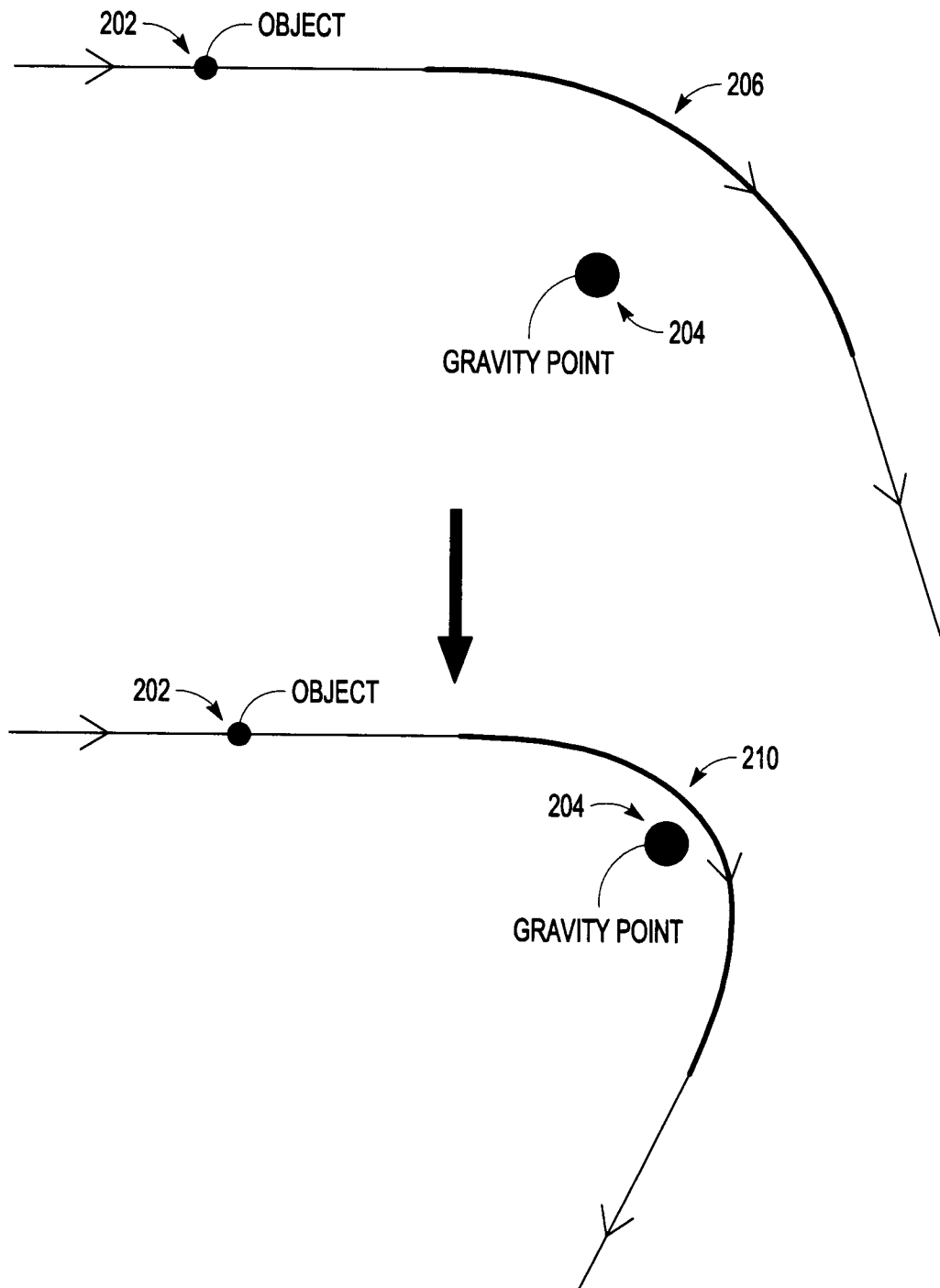


FIG. 2B

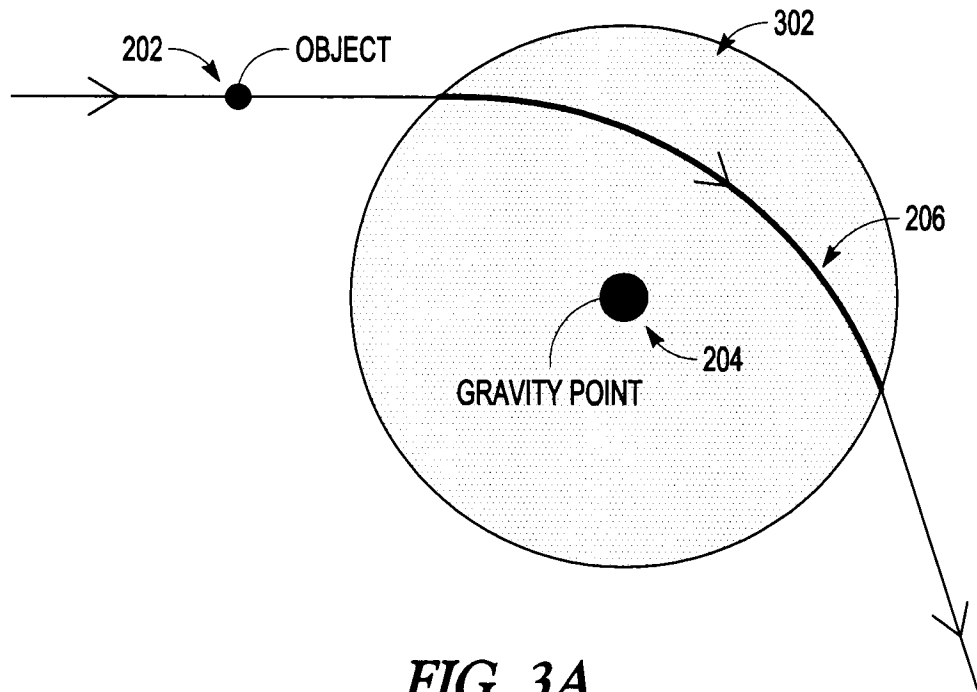


FIG. 3A

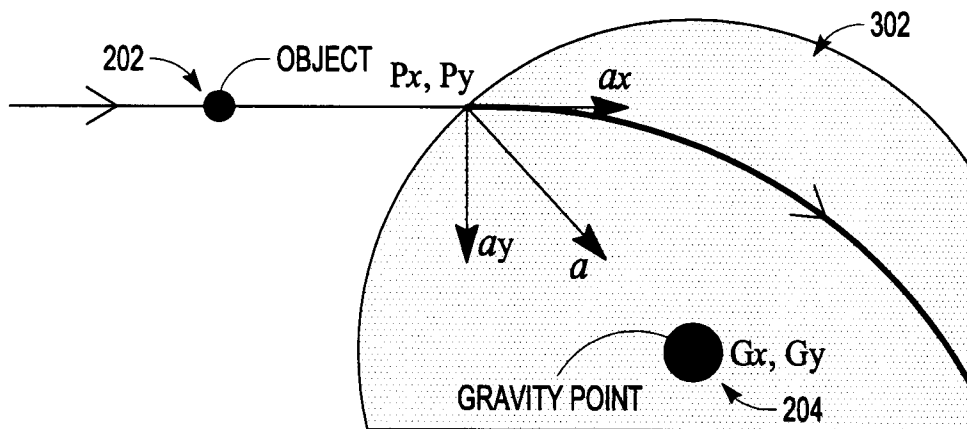


FIG. 3B

IN PHYSICAL MODEL

$$F = G \frac{m_1 m_2}{r^2}$$

$$F_x = F * \sin(a)$$

$$F_y = F * \cos(a)$$

$$a_x = F_x/m$$

$$a_y = F_y/m$$



CONVERSION TO GRAVITY POINT MODEL

$$d = \sqrt{(Px - Gx)^2 + (Py - Gy)^2}$$

$$a = \frac{m}{\frac{d}{r} * 100\%}$$

$$a_x = a * \sin(a)$$

$$a_y = a * \cos(a)$$

FIG. 3C

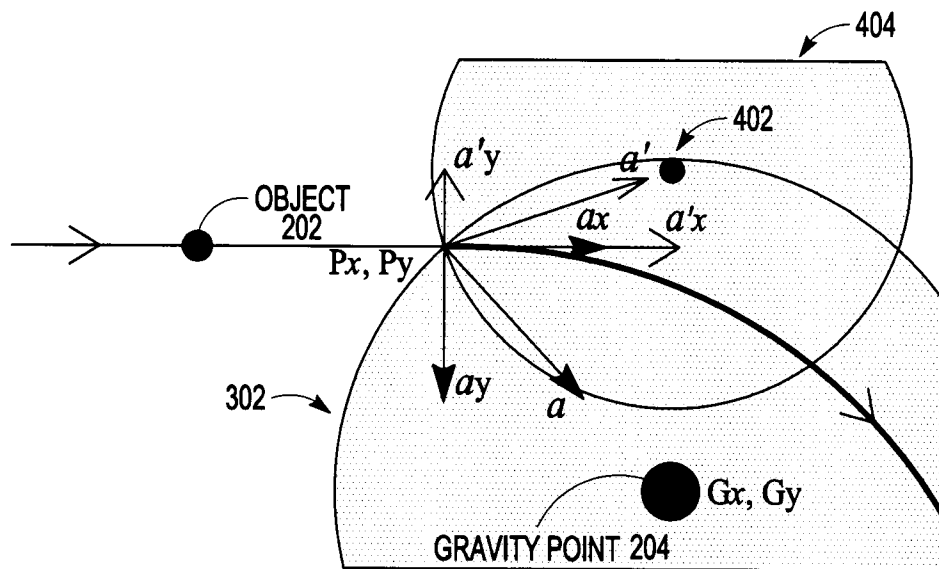


FIG. 4A

AFTER INVERSE OPTION ON

$$d = \sqrt{(Px - Gx)^2 + (Py - Gy)^2}$$

$$a = \frac{m}{\frac{r-d}{r} * 100\%}$$

$$ax = a * \sin(a)$$

$$ay = a * \cos(a)$$

FIG. 4B

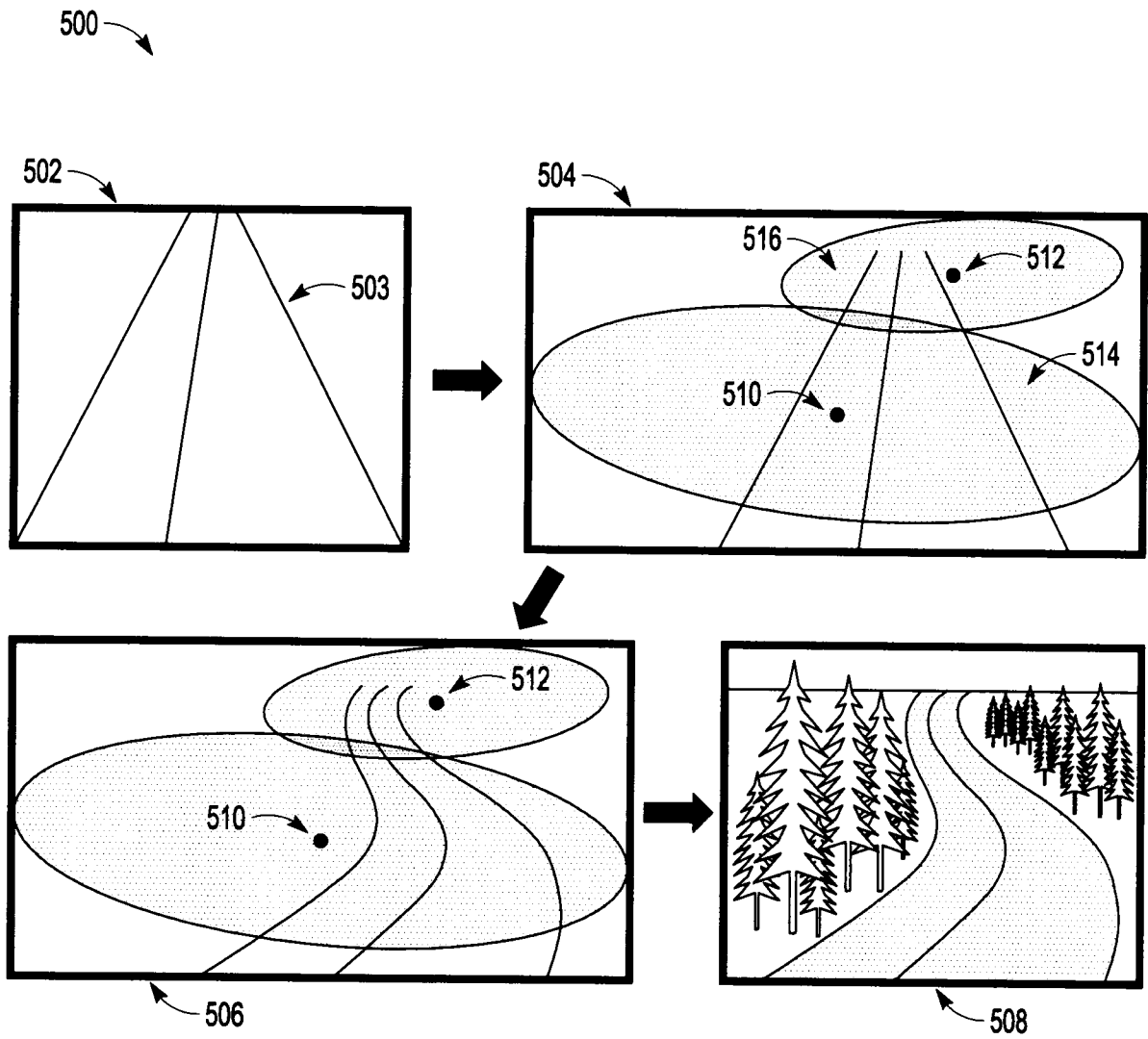


FIG. 5

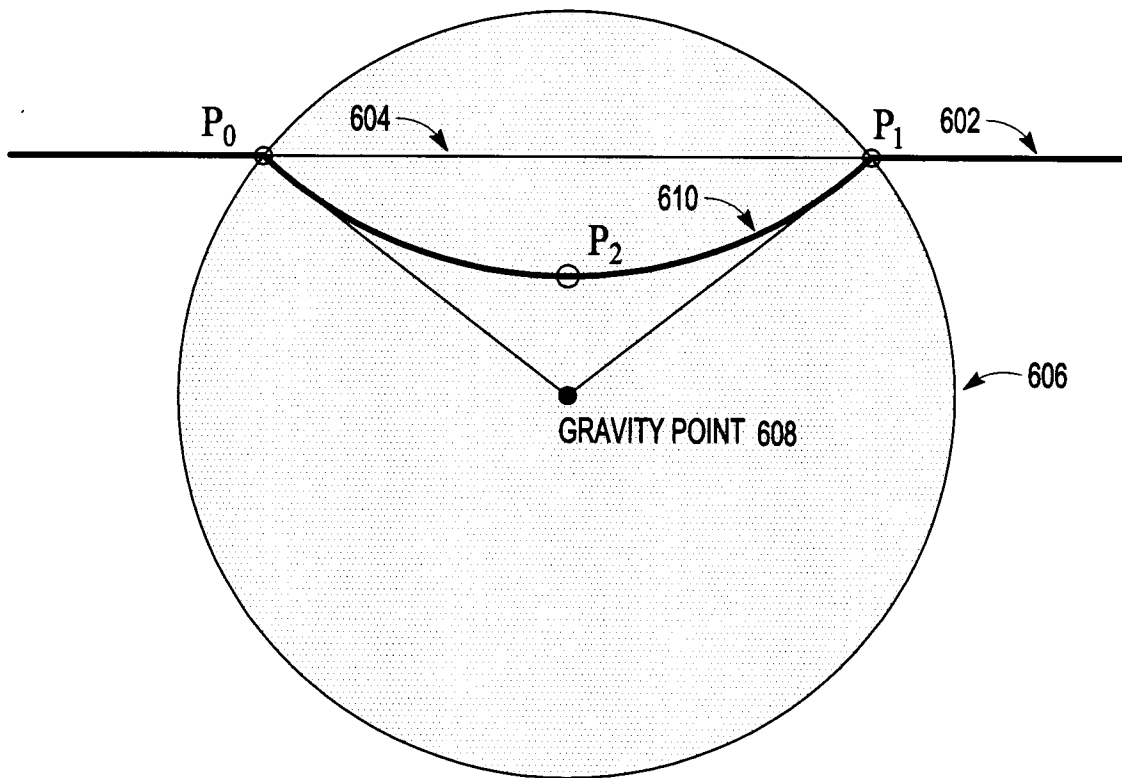


FIG. 6A

INTERSECT POINTS: P_0, P_1

$$P_{2x} = P_{0x}$$

$$P_{2y} = G_y - \frac{G_y - P_{0y}}{m * 100\%}$$

$$B(t) = (1-t)^2 P_0 + 2t(1-t) P_1 + t^2 P_2, t \in [0, 1]$$

FIG. 6B

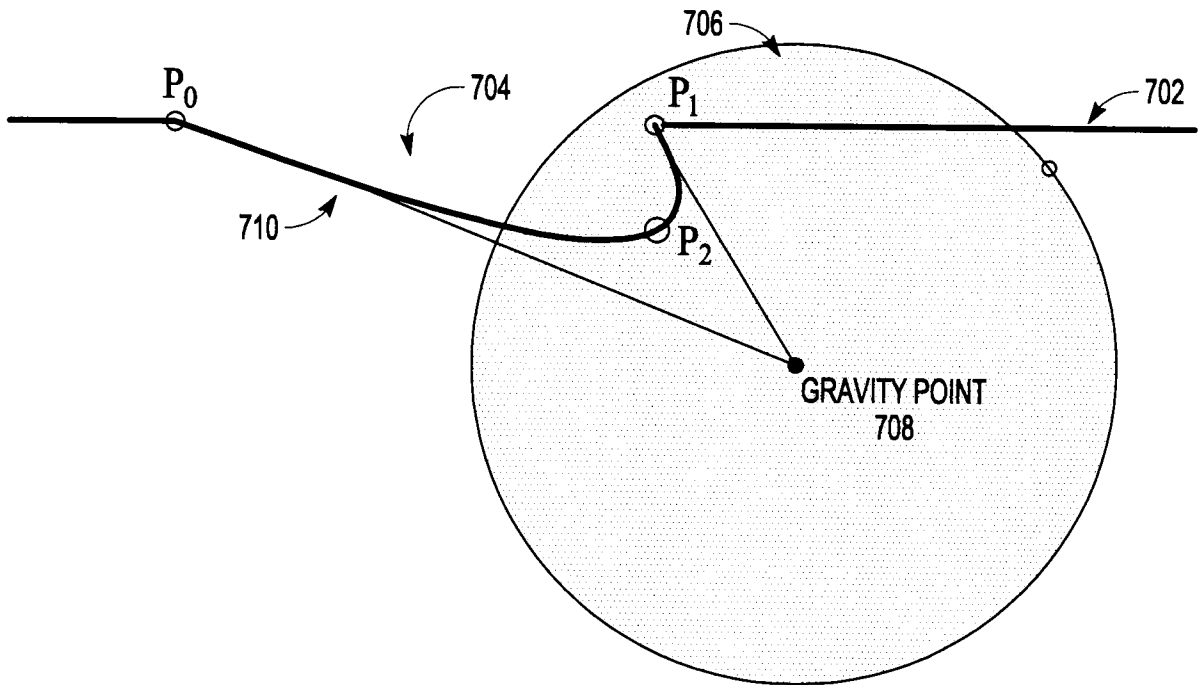


FIG. 7A

FIXED INTERSECT POINTS: P₀, P₁

$$P_{2x} = G_x - \frac{G_x - \frac{P_{0x} + P_{1x}}{2}}{m * 100\%}$$

$$P_{2y} = G_y - \frac{G_y - P_{0y}}{m * 100\%}$$

$$B(t) = (1-t)^2 P_0 + 2t(1-t) P_1 + t^2 P_2, t \in [0, 1]$$

FIG. 7B

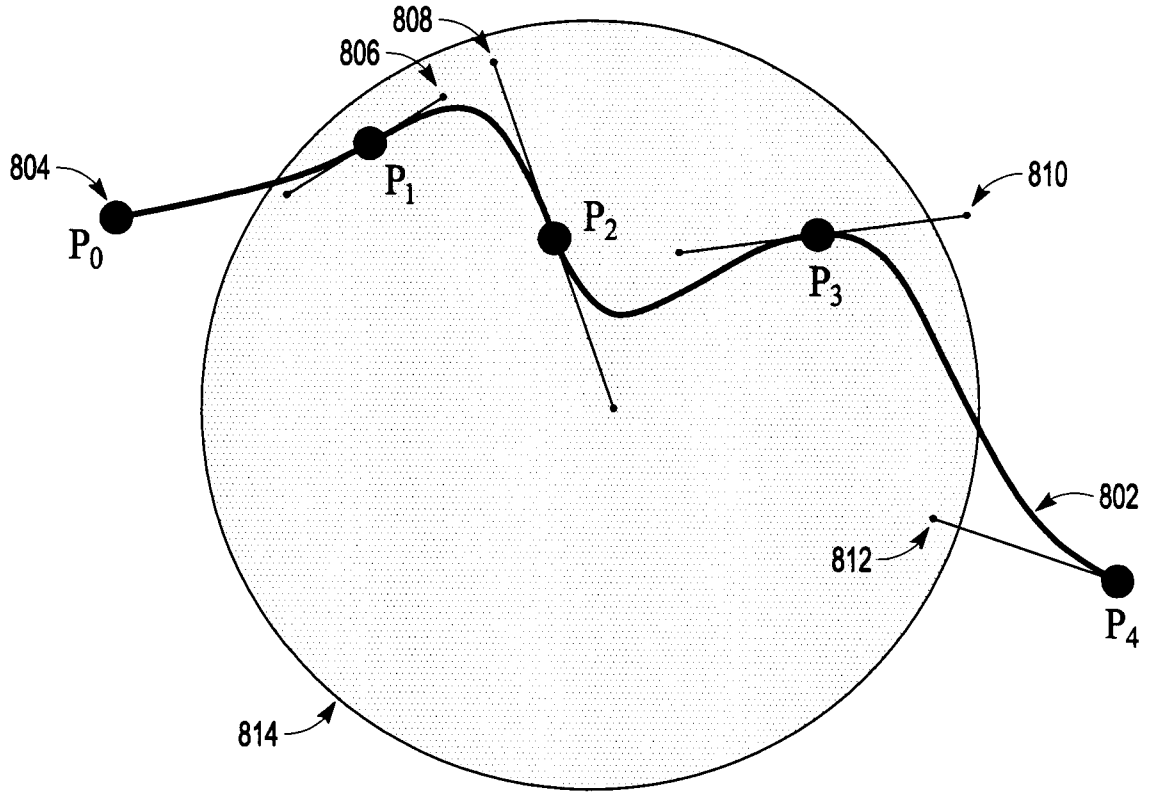


FIG. 8A

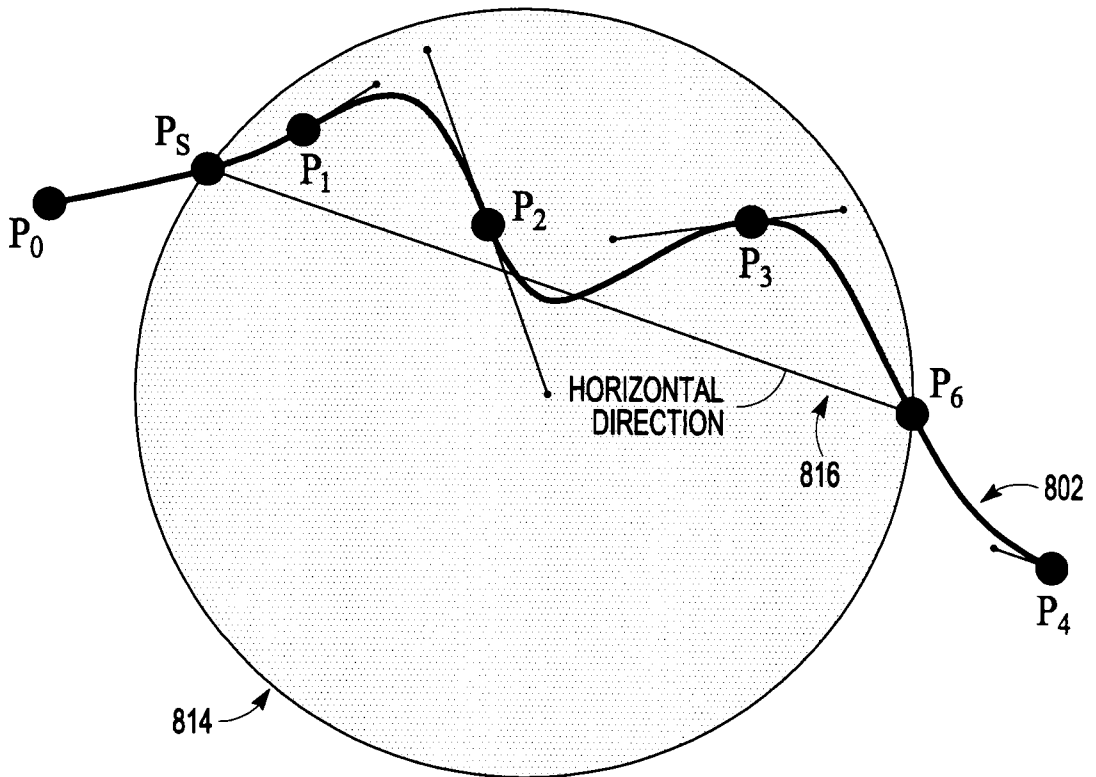


FIG. 8B

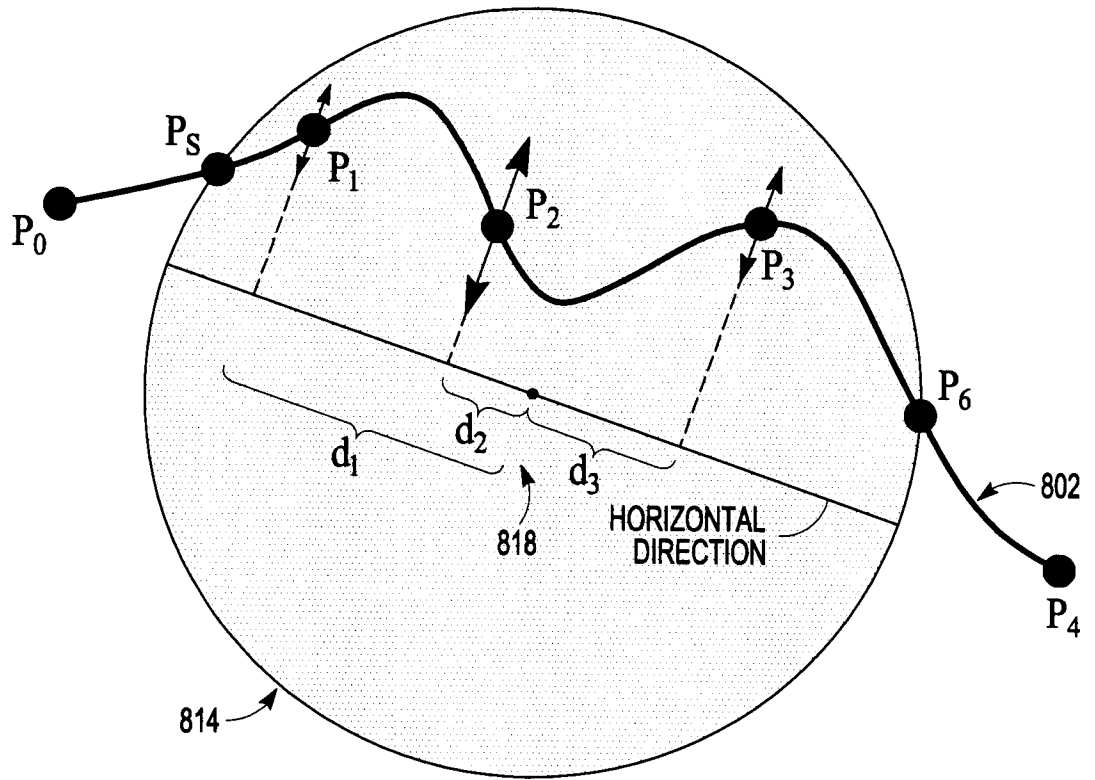


FIG. 8C

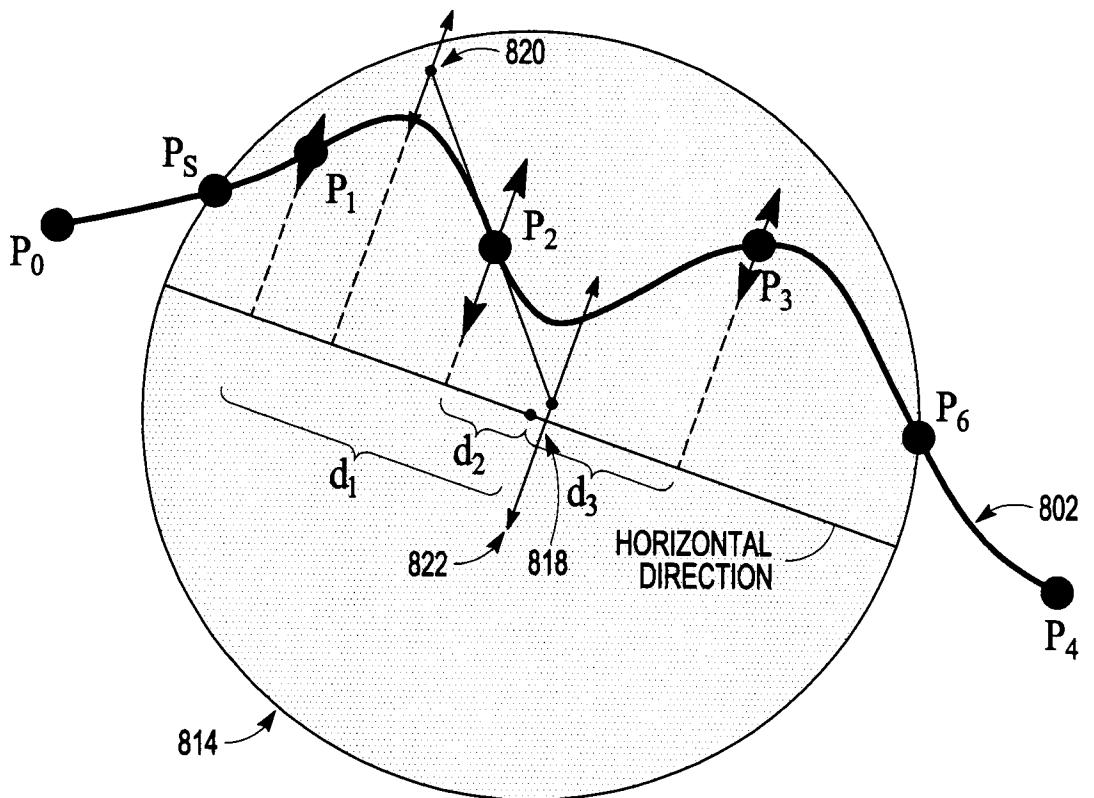


FIG. 8D

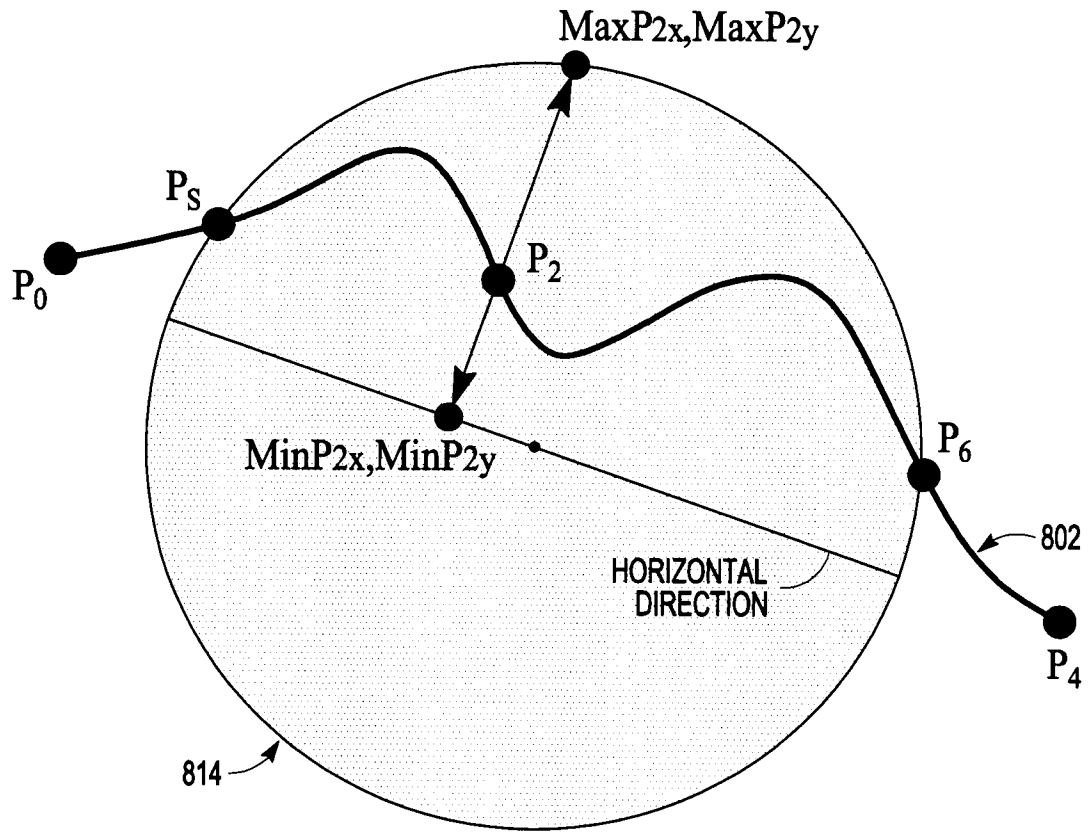


FIG. 8E

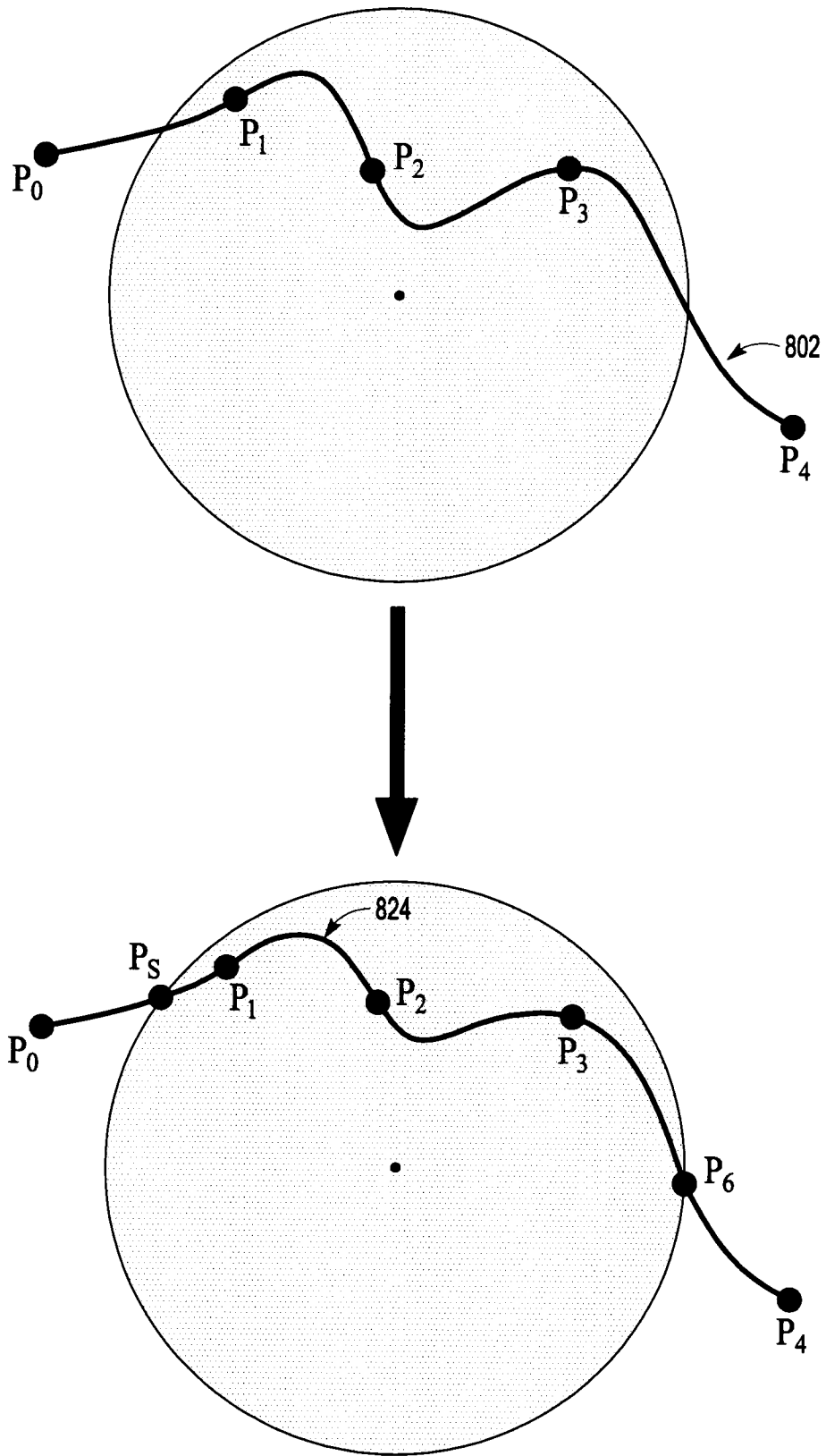


FIG. 8F

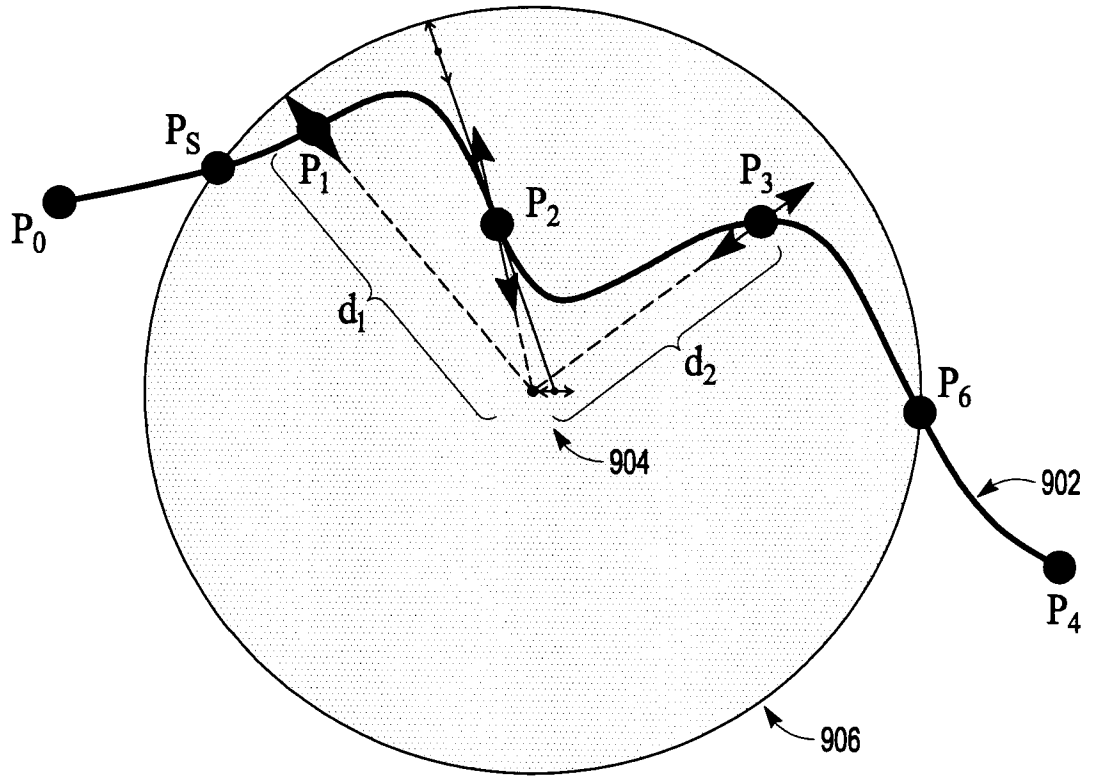


FIG. 9A

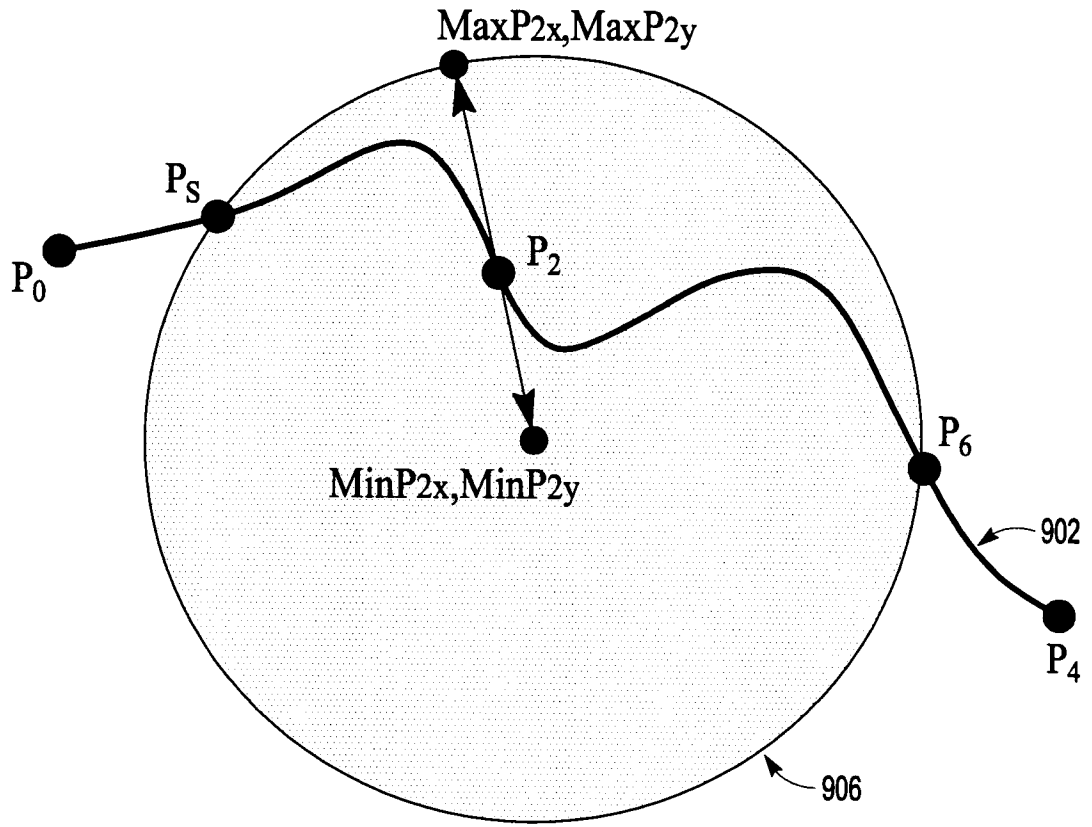


FIG. 9B

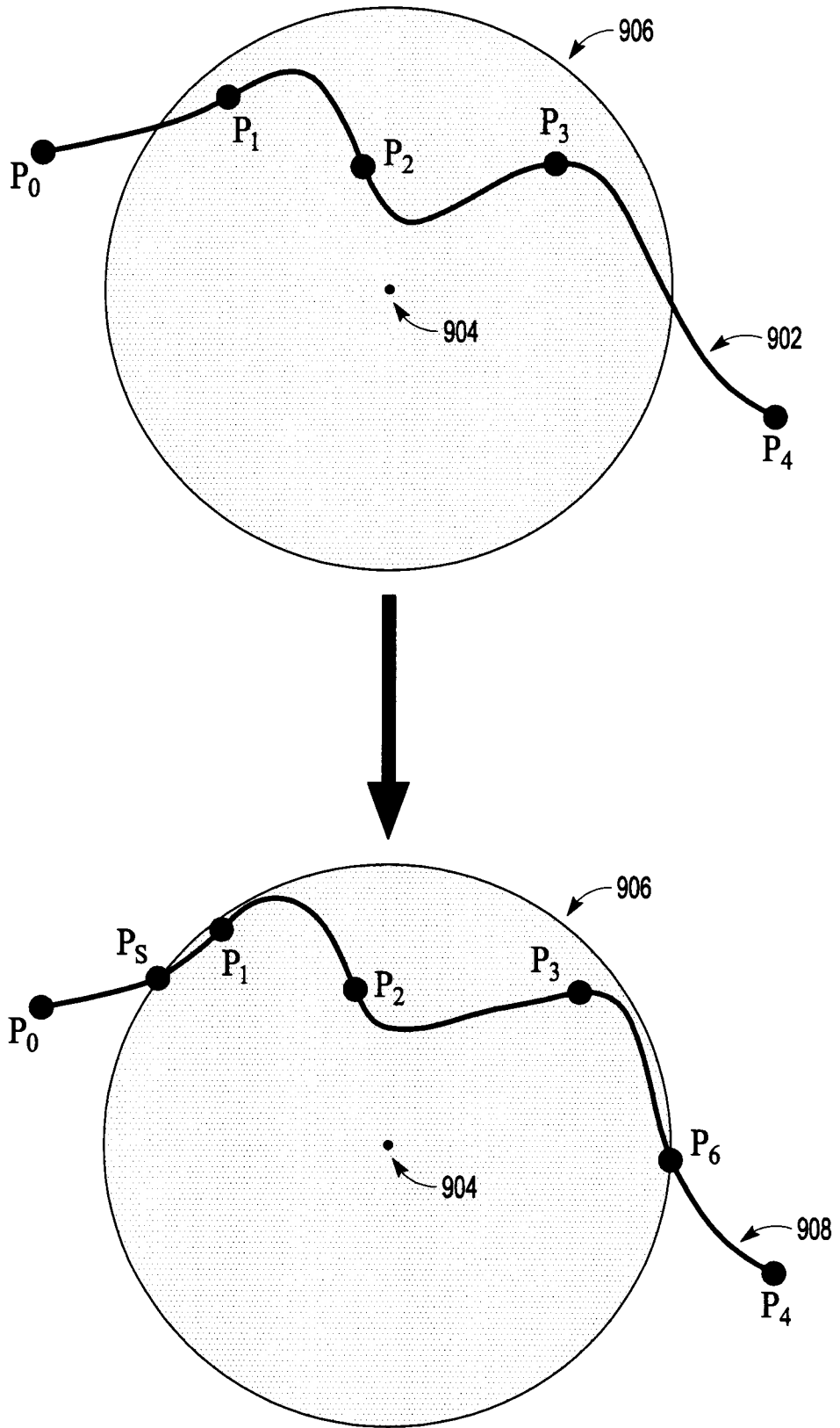


FIG. 9C

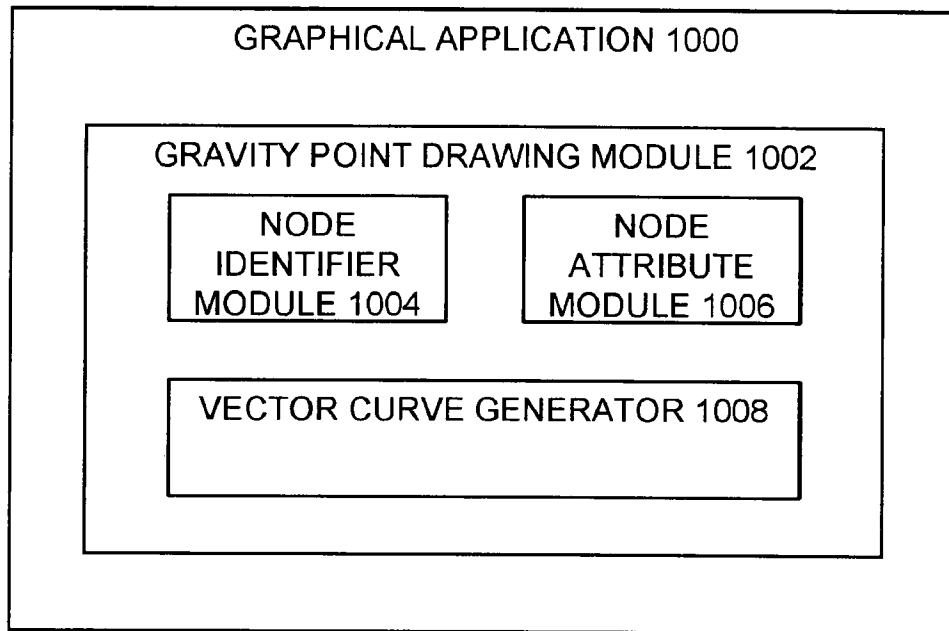


FIG. 10

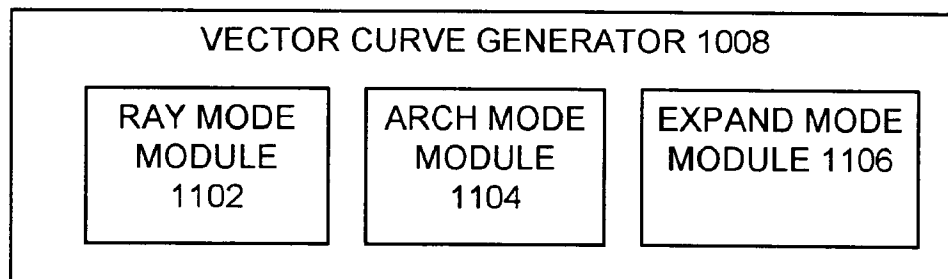


FIG. 11

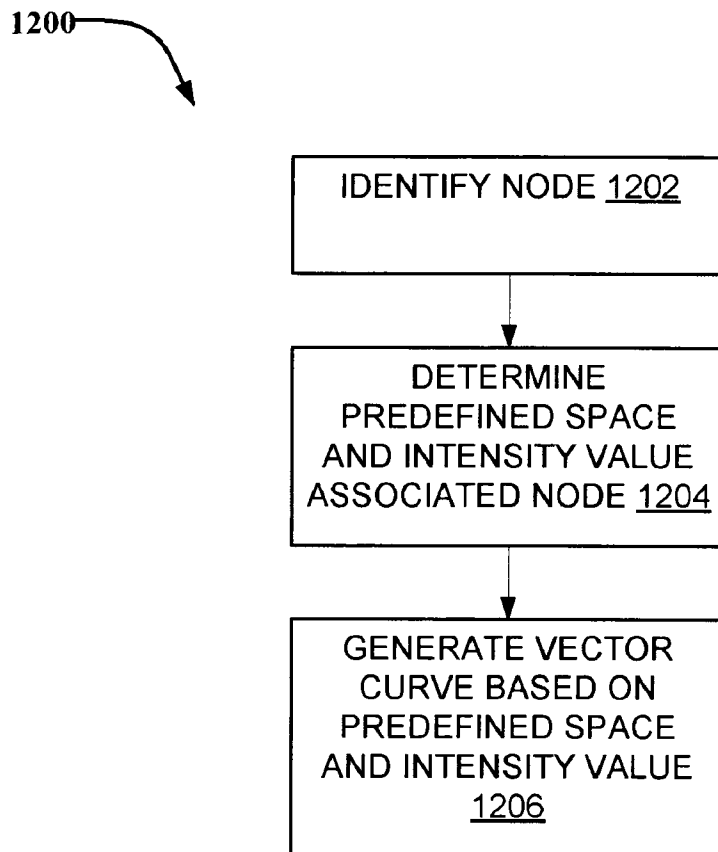



FIG. 12

1300 

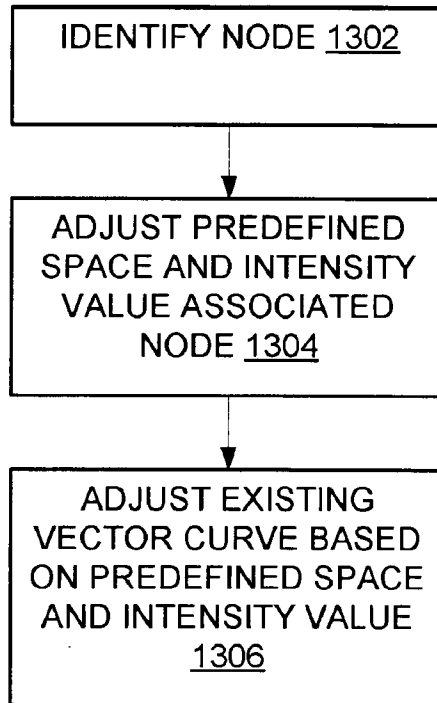


FIG. 13

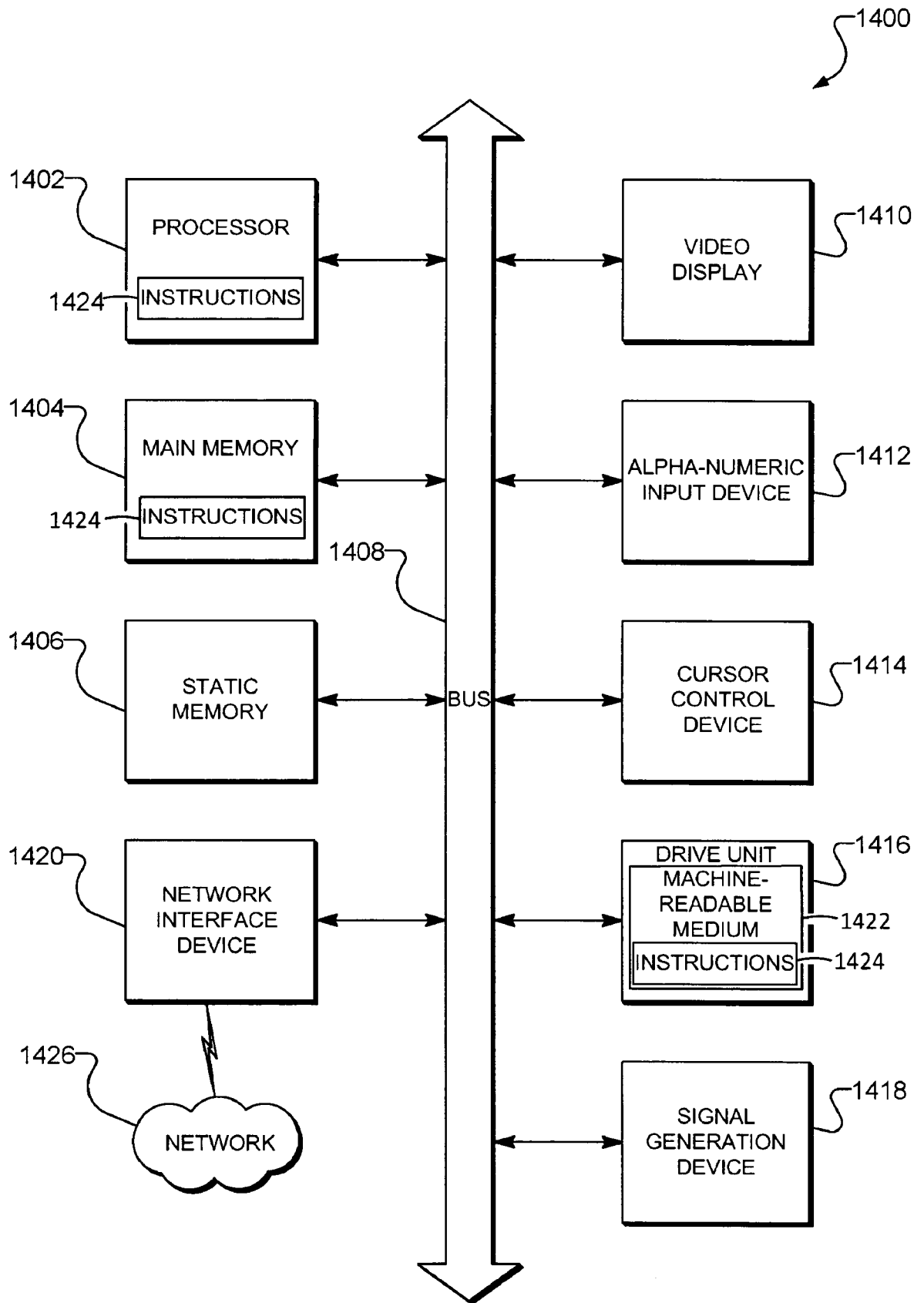


FIG. 14

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2013/077797

A. CLASSIFICATION OF SUBJECT MATTER

G06T 15/00 (2011.01) 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)

IPC: G06T, G06K, G06F

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

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

EPODOC, WPI, CNKI, CNPAT: gravity point, gravity center, curve, curvature, size, shape, intensity, vector

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2009/0051680 A1 (MITSUBISHI HEAVY INDUSTRIES, LTD. et al.) 26 February 2009 (26.02.2009) description, paragraphs [0033]-[0064] and figures 2-10	1-20
A	CN 101996515 A (JILIN UNIVERSITY) 30.03.2011 (30.03.2011) the whole document	1-20
A	CN 102880593 A (HARBIN INSTITUTE OF TECHNOLOGY) 16.01.2013 (16.01.2013) the whole document	1-20

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

* Special categories of cited documents:	“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
“A” document defining the general state of the art which is not considered to be of particular relevance	“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
“E” earlier application or patent but published on or after the international filing date	“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
“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)	“&” document member of the same patent family
“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	

Date of the actual completion of the international search 07 March 2014 (07.03.2014)	Date of mailing of the international search report 27 Mar. 2014 (27.03.2014)
---	--

Name and mailing address of the ISA/CN
The State Intellectual Property Office, the P.R.China
6 Xitucheng Rd., Jimen Bridge, Haidian District, Beijing, China
100088
Facsimile No. 86-10-62019451

Authorized officer

LI, Jia

Telephone No. (86-10)82245495

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2013/077797

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
US 2009/0051680 A1	26.02.2009	WO 2006101080 A1	28.09.2006
		JP 2006268188 A	05.10.2006
		EP 1868120 A1	19.12.2007
		INKOLNP 200703545 E	18.01.2008
		KR 20070103783 A	24.10.2007
		CN 101147149 A	19.03.2008
CN 101996515 A	30.03.2011	None	
CN 102880593 A	16.01.2013	None	