IMAGE PROCESSING APPARATUS MINIMIZING BLIND-SPOT BY EXPANDING FIELD OF VIEW FOR SURROUND VIEW IMAGE

[Field of the Invention]

The present invention relates to an image processing apparatus for a surround view image, and more particularly, to provide a more detailed surround view image of a vehicle, this apparatus restores lost images at the stitch line where images acquired by different cameras meet and reduces blind spots of the vehicle.

[Background of the Invention]

Automakers install rear cameras to allow drivers to see the rear of the vehicle when reversing for driver convenience and safety. Recently, a surround-view image system (Surround View System) has been developed and is used to provide drivers with more convenient and safe driving operation by checking the front, rear, left, and right sides during parking and driving by installing cameras not only at the rear but also on the front, left and right sides of the vehicle.

However, there are some problems in existing surround-view systems that quality of the combined image deteriorates due to noise or that blind spots occur where obstacles close to the camera are not visible depending on the location or direction of the installed camera, resulting in failure to provide drivers with images of obstacles located in blind spots.

[Previous Technical Citation]

[Patent Citation]

Korea Open Patent No. 10-2022-0061334 (Publication Date: 05/13/2022, Name of Invention: Vehicle Around View Provision System)

[Description of the Invention]

[Problems to be Solved]

The problem that this invention intends to solve is to minimize the loss of objects located around the vehicle by reducing blind spots in a surround view image. Another problem that this invention intends to solve is to reduce the image processing time for surround view image.

[Means for Solving the Problem]

To solve the above-mentioned problems this invention, Image Processing Apparatus for Surround View Image, includes the coordinate transformation table creation part that transforms coordinates of the pixel for virtual data corresponding to the shooting data into 3D mesh coordinates, and then the 3D mesh coordinates are transformed into 3D extended mesh coordinates that is created by a built-in field of view (FOV) expansion module that changes the position of the 3D mesh coordinate to a location expanded in the direction of the boundary line of field of view (FOV) by 180 degrees.

The FOV expansion module can transform the 3D mesh coordinates of pixels into 3D extended mesh coordinates according to [Mathematical formula 1] to [Mathematical formula 5] below.

[Mathematical formula 1]

$$l=\sqrt{x_{me}^{2}+(y_{me}-g)^{2}}$$

(Here in the formula, x_{me} and y_{me} are the actual mesh coordinate values for the X and Y axes of the each pixel, l is the distance from the center of the virtual circle with a radius equal to the distance from the position of the 3D mesh coordinates of the pixel with a coordinate value of (x_{me}, y_{me}, 0) to the center of the front of the vehicle on which the camera is mounted, connected by a straight line to the circumference of a virtual circle with a radius equal to that distance.)

[Mathematical formula 2]

$$\theta = \arctan(y_{me} - g,\ x_{me})$$

(Here in the formula, θ is the angle formed by the 180-degree boundary line of FOV and the line connecting the center of the virtual circle and a pixel with a coordinate value of (x_{me}, y_{me}, 0).)

[Mathematical formula 3]

$$w = \frac{\frac{\pi}{2}}{\frac{\pi}{2} - \phi}$$

(Here in the formula, *w* is the weight and φ is the angle formed by the line connecting the 180-degree boundary line of FOV and the stitch line from the center of a virtual circle.)

[Mathematical formula 4]

$$\delta = \frac{\pi}{2} - \theta$$
$$\delta' = w \times \delta$$
$$\theta' = \frac{\pi}{2} - \delta'$$

[Mathematical formula 5]

$$x_e = l \times \cos\theta'$$
$$y_e = l \times \sin\theta' + g$$

(Here in the formula, (x_e, y_e) is the value of the X-axis and Y-axis of the 3D extended mesh coordinate)

The coordinate transformation table creation part is connected to the world coordinate system transformation module and additionally includes a shooting range extension module. The pixel coordinate value for virtual data above is first input into the coordinate table along with 2D mesh coordinates and virtual camera position information. Then, the 2D pixel coordinates for virtual data are transformed into 3D coordinates in an orthogonal coordinate system, generating 3D coordinates for virtual data pixels. Next, the 3D coordinates of the virtual data pixel are transformed into 3D world coordinate system coordinates. The world coordinate system transformation module transforms the coordinates of the pixel in the 3D world coordinate system into the coordinates of the point where it meets the mesh to create 3D provisional mesh coordinates.

The shooting range extension module changes the position of the provisional 3D mesh coordinates authorized by the world coordinate system transformation module outward along the optical axis to produce coordinate values for the coordinates of actual 3D mesh.

The above shooting range extension module can use the following mathematical formula to calculate the position (r') of transformed coordinates:

$r' = r(1+kr^2)$

(where k is a constant and r is the distance from focus to provisional 3D mesh coordinates).

The coordinate transformation table creation part transforms the coordinates of the pixel for virtual data with the 3D extended mesh coordinates into the coordinates of the orthogonal coordinate system of the image acquisition unit.

Then, it transforms the coordinates of the orthogonal coordinate system back into spherical coordinate system coordinates to calculate the incident angle and azimuth angle of the acquired image data for the image acquisition unit that acquired the shooting data.

Then, using the calculated incidence angle and azimuth angle, the coordinate value of each pixel of the shooting data is transformed into a coordinate value of the corresponding pixel for virtual data, and the coordinate value of the pixel of the shooting data corresponding to the pixel for virtual data is determined.

Finally, it includes a camera coordinate system transformation module that uses predetermined weight and predetermined coordinate values according to virtual data pixel coordinates to create a coordinate transformation table.

The image processing apparatus for surround view image according to the above features is connected to the camera coordinate system transformation module and it includes a virtual data creation part that creates virtual data by using the coordinate transformation table to determine the pixel value of each pixel for virtual data.

The virtual data creation part is connected to a corresponding coordinate search module that, when the coordinate value of the pixel for virtual data is input, reads and outputs the coordinate value and weight of the pixel corresponding to the coordinate value of the input pixel using the coordinate transformation table.

The corresponding coordinate search module is connected to shooting data substitution modules that searches for pixels of shooting data with the above coordinate values output from the corresponding coordinate search module, reads and outputs the pixel values of the searched pixels.

The virtual data creation unit also includes a multiplier that calculates the pixel value of the pixel for virtual data by multiplying the pixel value authorized from each shooting data substitution module by the weight corresponding to the pixel for virtual data.

[The Effect of the Invention]

According to these features, by moving forward the pixel position of the photographed image acquired by different cameras toward the full boundary line of FOV, which is a 180-degree field of view, the image of the object that meant to be lost when the object is adjacent to an image stitch line and when the two images obtained from different cameras are combined can be recovered and provided as an entire surround view image.

As a result, the number of objects lost during the combination operation of two different images is greatly reduced, and the disappearance of objects adjacent to the vehicle from the entire surround view image can also be greatly reduced.

Therefore, thanks to the entire surround view image, the risk of an accident caused by not recognizing an object adjacent to the vehicle can be greatly reduced.

In addition, the driver can quickly and accurately figure out the surroundings of the front, rear, left, and right sides of the vehicle using the entire surround view image, in which almost all the photographed objects are displayed and the blind spot is greatly reduced.

Therefore, the driver can drive the vehicle more comfortably and easily, thereby greatly improving the driver's satisfaction.

Moreover, according to the present feature, the amount of image data to be processed is reduced, and thus the image processing time will be reduced.

[Brief Description of the Drawings]

FIG. 1 is a schematic block diagram of a surround view image providing system according to an embodiment of the present invention.

FIG. 2 is a schematic block diagram of a coordinate transformation table creation part of the operation unit (22) of FIG. 1.

FIG. 3 is a schematic block diagram of a virtual data creation part of the operation unit of FIG. 1.

FIG. 4 is a flowchart illustrating an operation of a method of providing image data for a surround view image according to an embodiment of the present invention.

FIG. 5 is a diagram illustrating a relationship between a stitch line and a boundary line of FOV according to an embodiment of the present invention.

FIG. 6 is an example of a total surround view image output to the information output unit, and (a) and (b) are only the rear and right images that expands to 180 degrees among all surround view images created using conventional virtual data, respectively,

(c) is a diagram illustrating an entire surround view image created using virtual data created by a surround view image providing system according to an embodiment of the present invention.

FIG. 7 is a diagram conceptually illustrating a world coordinate system and a virtual coordinate system in a system for providing image data for a surround view image according to an embodiment of the present invention.

FIG. 8 is a conceptual diagram illustrating a state in which one of pixels for virtual data is mapped to a mesh according to an embodiment of the present invention.

FIG. 9 is a diagram illustrating an incident angle and an azimuth angle for transforming 2D virtual data into a 3D orthogonal coordinate system according to an embodiment of the present invention.

FIG. 10 is a diagram illustrating a rotation angle based on respective axes of a roll, a pitch, and a yaw.

FIG. 11 is a diagram illustrating an operation concept of a shooting range expansion module of a system for providing image data for a surround view image according to an embodiment of the present invention.

FIG. 12 is a diagram illustrating an operation concept of a FOV extension module of the system for providing image data for a surround view image according to an embodiment of the present invention.

FIG. 13 is a diagram illustrating a relationship between a world coordinate system and a camera coordinate system.

[Description of the Embodiment]

Hereinafter, an embodiment disclosed herein will be described in detail with reference to the attached drawings, but regardless of the reference code, the same or similar components assign the same reference number and repeated descriptions thereof will be omitted.

In addition, in describing the embodiment disclosed in this specification, if it is determined that a specific description of the related known technology may obscure the gist of the embodiment disclosed in this specification, the detailed description will be omitted.

Terms including ordinal numbers, such as first, second, and the like, may be used to describe various components, but the components are not limited by the terms. The terms are used only

for the purpose of distinguishing one component from another component.

Singular expressions include plural expressions unless they are explicitly meant differently in the context.

In the present application, each of the steps described may be performed regardless of the order listed, except that it must be performed in the order listed by a special causal relationship.

In this application, the term "to include" or "to have" is to specify that there is a feature, number, step, operation, component, part, or combination of them described in the specification, and it should be understood that the existence or addition of one or more other features, steps, operation, component, component, or combination thereof is not excluded in advance.

Hereinafter, an image data providing system for a surround view image according to an embodiment of the present invention and a control method thereof will be described with reference to the attached drawings.

First, an image data providing system for a surround view image of the present embodiment will be described with reference to FIG. 1 to 3.

Referring to FIG. 1, it is equipped with multiple real cameras (11 to 14) mounted on the vehicle (100) and an image processing apparatus (20) that receives and processes image data transmitted from each real camera (11 to 14) in communication with each real camera (11 to 14).

Hereinafter, the actual camera (11 to 14) is referred to as a 'camera'. Also, in this example, as an example, the number of cameras (11 to 14) is four, but it is not limited to this, and the number of cameras mounted on the vehicle (100) can be increased or decreased as needed.

For example, the four cameras (11 to 14) may be mounted on the front, rear, left, and right sides of the vehicle (100), respectively.

Multiple cameras (11 to 14) are mounted on the vehicle and compose image acquisition unit that can create image data taken in a predetermined range around the vehicle and output it to the image processing apparatus (20).

These cameras (11 to 14) are for shooting an image for generating the entire surround view image.

In order to create a more efficient and accurate overall surround view image, the occurrence of blind spots should be minimized, and in order to minimize these blind spots, cameras (11 to 14) should be installed at a high location as much as possible.

For example, the mounting height of the first to fourth cameras (11 to 14) can be determined by the highest height from the ground to the corresponding section of the vehicle (e.g., the roof of the vehicle) measured through the section to which each camera (11 to 14) is attached.

That is, the highest height of the section may be the height of the vehicle measured as the line extending in a straight line along the direction perpendicular from the ground passes through the attaching position of the camera (11 to 14).

Accordingly, each camera (11 to 14) may be attached to a portion of the vehicle corresponding to 9/10 to 10/10 of the highest height of the portion of the vehicle from the ground.

The top surface of the vehicle exposed to the outside (e.g., the outer surface of the roof of the vehicle) may be 10/10 of the highest height of the part of the vehicle from the ground, and in this case, the camera (11 to 14) will be located in the corresponding part of the roof of the vehicle.

Each of these cameras (11 to 14) operates under the control of the image processing apparatus (20) to shoot a set range of areas to create multiple image data (hereinafter referred to as ‘shooting data') (e.g., first to fourth shooting data) and send each shooting data to the image processing apparatus (20) so as to be stored in the image processing apparatus (20).

Each of these cameras (11 to14) includes a lens unit equipped with at least one lens and an image sensor unit capable of generating image data by detecting incident light and converting it into an electrical signal.

In addition, each camera (11 to 14) is equipped with a communication unit (e.g., the first communication unit) for communication between the image processing apparatus (20), for example, between the camera and the image processing apparatus (20).

Each of these cameras (11 to 14) may be an optical camera, and may be equipped with a lens having a wide field of view such as a fisheye lens.

The image processing apparatus (20) is for surround view images, and can create each coordinate transformation table, which is a mapping table for mapping the coordinates of each pixel of shooting data corresponding to each pixel for virtual data. Then, the data for the entire surround view image (hereinafter, data for the entire surround view image is referred to as 'entire surround view image data') can be created by determining the pixel value of each pixel for each virtual data using the corresponding coordinate transformation table.

In this example, a virtual camera is a virtual camera having the same structure as a camera, and in this example, it is assumed that one virtual camera exists.

In this example, virtual data is basic image data for generating the entire surround view image data, and will be created corresponding to the corresponding shooting data created by each camera.

Therefore, in this example, since first to fourth cameras are mounted on the front, rear, left, and right sides, virtual data will also be first to fourth virtual data, such as front virtual data, rear virtual data, left virtual data, and right virtual data, corresponding front shooting data, rear shooting data, left shooting data, and right shooting data, respectively.

The image processing apparatus (20) in the example creates front virtual data, rear virtual data, left virtual data, and right virtual data using first to fourth shooting data, and creates full surround view image data using these front virtual data, rear virtual data, left virtual data, and right virtual data.

The virtual camera is assumed to be mounted at a designated position above the vehicle (100) up in the air, e.g., in the sky), and the surround view image, that is, the image data corresponding to the entire surround view image, which is an image obtained when looking at the vehicle from the virtual camera at the virtual location up in the air, will be the entire image data.

Each virtual data and all image data have multiple pixels arranged in a matrix structure, and the size of the virtual data is determined by the number of pixels arranged in the row direction and the number of pixels arranged in the column direction.

In this example, the sizes of each virtual data and the entire image data are the same, and have a size of 1920×1080.

The pixel value of each pixel of all virtual data is determined using the pixel value, or the pixel value and weight of the corresponding pixel of the shooting data mapped to the coordinate transformation table, and the pixel value of each pixel of the entire surround view image data is determined by adding the pixel value of the corresponding location of all virtual data.

In this example, the weight is determined for each camera and stored in the storage unit (23), and is also determined according to the location (i.e., coordinates) of the pixel. These weights have a value of 0 to 1.

As already described, the coordinate transformation table of this example is a table in which the coordinate values and weights of the corresponding pixels of the corresponding shooting data are determined to determine the pixel value of each pixel for the corresponding virtual data.

Therefore, the pixel value of each pixel for the virtual data (e.g., first virtual data) is determined based on the pixel value of the pixel corresponding to the coordinate value of the shooting data (e.g., first shooting data) mapped to the coordinate transformation table.

For this reason, the image processing apparatus (20) reads the coordinate value of the determined shooting data, reads the pixel value of the pixel of the shooting data corresponding to the read coordinate value, determines the pixel value of each pixel of the virtual data, and creates the corresponding virtual data.

For example, in the coordinate transformation table, if the coordinate value of the pixel of the shooting data corresponding to the first virtual data pixel located at (1,1) (row, column) is determined to be (1,5), the image processing apparatus (20) can determine the pixel value of the corresponding pixel for the first virtual data using the pixel value of the pixel of the shooting data having a coordinate value of (1,5).

As already described, if a weight is determined for each pixel for virtual data, the pixel value of each pixel for virtual data is determined by using an additional weight in addition to the corresponding pixel value of the photographed data.

Therefore, in this way, the pixel values of all pixels of each virtual data pixel can be determined using the pixel values of each shooting data pixel having mapped coordinate values and, if necessary, using weights additionally, and as a result, the corresponding virtual data can be generated.

The mapping value of this coordinate transformation table can change depending on the position and angle of the camera as well as the position and angle of the virtual camera.

In this example, each shooting data, as already described, refers to image data acquired by the shooting operation of the corresponding camera (11 to 14), and each virtual data is image data corresponding to the shooting data created in each camera (11 to14).

In addition, the entire image data refers to virtual image data acquired by the virtual camera, and as already described, it can be calculated as a weighted sum of each virtual data.

Therefore, each virtual data will be image data for a partial surround view video that forms the entire surround view image based on the composition of the entire surround view video represented by the entire image data. For example, it can be front surround view image data, rear surround view image data, left side surround view image data, or right side surround view image data.

As described in FIG. 1, the image processing apparatus (20) is equipped with a communication unit (e.g., a second communication unit) (21), an operation unit (22) connected to the second communication unit (21), a storage unit (23) connected to the operation unit (22), and an information output unit (24) connected to the operation unit (22).

The second communication unit (21) is for wired or wireless communication with the first communication unit of each camera (11 to 14), and can be equipped with a short-range communication module, for example.

The operation unit (22) is a processor that controls the overall operation of the image data supply system for the surround view image of the vehicle (100).

In this example, the operation unit (22) can create a coordinate transformation table and create multiple virtual data and entire surround view video data using the created coordinate transformation table.

Therefore, as described in FIG. 1, the operation unit (22) is equipped with coordinate transformation table creation part (2201) for creating the coordinate transformation table, and a virtual data creation part (2301) that creates virtual data and entire surround view video data using the coordinate transformation table created in the coordinate transformation table creation part.

The coordinate transformation table creation part (2201) creates a coordinate transformation table by setting a coordinate value of the shooting data mapped to each coordinate of each pixel for virtual data. In this case, the value of the weight is determined according to the location of the corresponding pixel of the virtual data.

Accordingly, the coordinate transformation table creation part 2201 creates a plurality of coordinate transformation tables (e.g., first to fourth coordinate transformation tables) for first to fourth virtual data.

The virtual data creation part (2301) uses the corresponding coordinate transformation table to determine the pixel value of each pixel for the virtual data (e.g., the first virtual data) based on the pixel value of the mapped location of the pixels in the shooting data. The virtual data is created by multiplying the determined pixel values by the corresponding weight. Each virtual data thus created is partial surround view image data.

In this way, first to fourth virtual data are created using each coordinate transformation table and weight.

Once the first to fourth virtual data (i.e., the first to fourth partial surround view image data) corresponding to the first to fourth shooting data are created, the virtual data creation part (2301) adds the pixel values of the pixels at the same location in the first to fourth virtual data to determine the pixel values of each pixel in the entire surround view image data and creates the entire surround view image data.

When such entire surround view image data is output to the information output unit (24), the information output unit (24) provides the driver of the vehicle (100) with a surround view image of the vehicle (100), i.e., an entire surround view video.

Therefore, the driver can quickly and accurately figure out the situation around the vehicle by checking the entire surround view video of the vehicle (100) output by the information output unit (24).

The structure and operation of the coordinate transformation table creation part (2201) and virtual data creation part (2301) of the operation unit (22) will be described in detail later.

The storage unit (23) is a storage medium for storing data necessary for operation of a surround view providing system of a vehicle (100) or data created during operation.

Such a storage unit (23) is a memory.

In this example, the storage unit (23) stores various data and information for operation of the image processing apparatus (20) and data created during operation.

For example, the storage unit (23) stores captured data (e.g., first to fourth captured data) acquired by each camera (11 to 14), virtual camera position information which is position information of a virtual camera, and camera position information which is position information of each camera.

Virtual camera position information and camera position information include at least one of a center points (=focal points), coordinates of a located point, central angle, and rotation angle of a virtual camera or camera. At this time, the rotation angle of a virtual camera or camera includes a rotation angle (α) based on a roll axis, a rotation angle (β) based on a pitch axis, and a rotation angle (γ) based on a yaw axis.

The information output unit (24) is for generating visual output and includes a display module.

The display module outputs the entire surround view image created according to the control of the operation unit (22) on the screen.

As a result, the driver can check the entire surround view image displayed on the screen of the display module.

The display module includes at least one of a liquid crystal display, an organic light emitting diode display, a flexible display, and a 3D display.

Next, referring to FIG. 2, the structure of the coordinate transformation table creation part (2201) of the operation unit (22) will be described.

As shown in FIG. 2, the coordinate transformation table creation part (2201) of the operation unit (22) includes a world coordinate system transformation module (221) into which coordinate values (x, y) of the location (i.e., coordinates) of each pixel for each virtual data and virtual camera position information are input, a shooting range expansion module (222) connected to the world coordinate system transformation module (221), a plurality of field of view (FOV) expansion modules connected to the shooting range expansion module (222), and a plurality of camera coordinate system transformation modules (2241-2244) respectively connected to each FOV expansion module (2231-2234).

The world coordinate system transformation module (221) transforms the coordinates of 2D virtual data into 3D coordinates based on a world coordinate system to create 3D mesh coordinates for all pixels for each virtual data.

The shooting range expansion module (222) can expand the shooting range, such as the shooting distance, which has been reduced due to the creation of 3D mesh coordinates using 2D mesh coordinates instead of 3D, so that the shooting effect of a fisheye lens can be exerted.

Therefore, the 3D mesh coordinates whose position has been changed by the shooting range expansion module (222) can be called true 3D mesh coordinates according to the world coordinate system (hereinafter, true 3D mesh coordinates according to the world coordinate system are referred to as '3D actual mesh coordinates').

Each FOV expansion module (2231-2234) is for transforming image data having 3D actual mesh coordinates output from the shooting range expansion module (222) into image data having a FOV of 180 degrees.

Therefore, each FOV expansion module (2231-2234) calculates the displacement position for the 3D actual mesh coordinates according to a predetermined mathematical formula and changes at least one coordinate of each pixel for each virtual data to a coordinate moved toward a FOV boundary line of 180 degrees. In this example, a FOV boundary line of 180 degrees is a line where the field of view of cameras (11 to 14) forms an angle of 180 degrees.

Also, each camera coordinate system transformation module (2241-2244) transforms the coordinates of pixels for the corresponding virtual data that have been extended at least partially to the FOV boundary line into camera coordinates based on a camera coordinate system to create each virtual data and entire surround view image data using images of captured data acquired by each camera (11 to 14).

Next, referring to FIGs. 4 to 13, the operation of the coordinate transformation table creation part (2201) having such a structure in this example will be described.

The size of each virtual data is already stored in the storage unit (23).

As already described, virtual data can include a plurality of pixels arranged in a matrix structure, and the size of virtual data is determined by the number of pixels arranged in a row direction and the number of pixels arranged in a column direction.

In addition, in this example, there are world coordinate system, virtual coordinate system and camera coordinate system.

The world coordinate system is a reference coordinate system in 3D space. As shown in FIG. 7, when a vehicle (100) is located on a flat ground, the world coordinate system includes X, Y, and Z axes extending in the X, Y, and Z directions, respectively. At this time, the X direction is the width direction of the vehicle (100), the Y direction is the length direction of the vehicle (100), and the Z direction is perpendicular to the ground or the bottom of the vehicle (100).

Also, as shown in FIG. 7, the virtual coordinate system is a local coordinate system centered on a virtual camera and includes X, Y, and Z axes extending in the X, Y, and Z directions, respectively. In this example, the direction of the Z axis in the virtual coordinate system is the direction of the optical axis.

In the case of a virtual camera assuming no lens distortion, the position where an object (e.g., vehicle (100) and its surroundings) captured through the virtual camera is focused is determined according to the incident angle at a predetermined focal distance, and another surface (e.g., corresponding plane) focused at the same focal distance as the surface where the image is focused exists. Therefore, the entire surround view image data in this example is image data including a plurality of pixels acquired from this corresponding plane.

FIG. 8 is a diagram showing an example in which virtual data is mapped to a mesh.

A mesh can be composed of minimum units such as polygons made by collecting vertices and is for displaying 3D images, and is a virtual surface where an object (e.g., subject for photograpy) is focused.

In FIG. 8, (x_0, y_0, z_0) is data related to the position of the virtual camera and is the coordinates of the focal point of the virtual camera.

In the case of FIG. 8, the graph (z) for the mesh can be represented as z=f(x, y), and if the mesh is the ground, z=0. In this example, if the mesh is set to be the ground, the coordinate value of the Z-axis coordinate (z) becomes 0 and the pixel focused on the mesh has a 2D coordinate (e.g., 2D mesh coordinate) with X and Y axis coordinates.

Therefore, the world coordinate system transformation module (221) receives input of the corresponding coordinates of each pixel for each virtual data and virtual camera position information. As already described, in this example, since the mesh is the ground and the coordinate value of the Z-axis coordinate (z) is 0 (i.e., z=0), the world coordinate system transformation module (221) can recognize that the mesh is a 2D ground by receiving input of z=0 and can set the value of the z-axis coordinate of each pixel for virtual data to '0' when transforming the coordinate system of each pixel for virtual data into a world coordinate system.

As shown in FIG. 4, the world coordinate system transformation module (221) transforms the coordinates [e.g., (n, k)] of the corresponding pixel (P) among the 2D pixels for virtual data input into coordinates (x_p, y_p, z_p) of a 3D orthogonal coordinate system to create 3D coordinates of the corresponding pixel among the pixels for virtual data (S10).

Each pixel of virtual data can exist on different lines of sight (LOS), so there can be as many lines of sight as the number of pixels composing virtual data.

Therefore, as shown in FIG. 9(a), the coordinate values (n, k) of the corresponding pixel among the 2D pixels for virtual data can be expressed as an incident angle (or latitude) (θ) and an azimuth angle (or longitude) (λ) shown in FIG. 9(b), and are transformed into coordinate values (x_p, y_p, z_p) of 3D coordinates through trigonometric functions using these incident angle and azimuth angle.

In other words, when the incident angle and azimuth angle for each pixel of virtual data are determined, the line of sight for the corresponding pixel is determined, and when the line of sight is determined, the coordinates of the point where the line of sight is extended and meets the mesh can be obtained.

Then, the world coordinate system transformation module (221) can transform the 3D coordinates of pixels for virtual data transformed into orthogonal coordinates into a 3D world coordinate system to create coordinates of pixels for virtual data transformed into a 3D world coordinate system (S20). Therefore, the coordinate values (x_p, y_p, z_p) of the corresponding pixel (P) can be changed to 3D world coordinate values (x_p', y_p', z_p') in a 3D world coordinate system (see FIG. 8).

The 3D world coordinate values (x_p', y_p', z_p') for the coordinate values (x_p, y_p, z_p) of the corresponding pixel (P) can be calculated according to the following [Mathematical Formula 1].

As shown in FIG. 10, the rotation angles based on each of the roll, pitch, and yaw axes are respectively alpha (α), beta (β), and gamma (γ), and these rotation angles (α, β, γ) are values related to the rotation angle of the virtual camera.

In [Mathematical Formula 1],

the $\begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -\sin(\beta) & 0 & \cos(\beta) \\ 0 & 1 & 0 \\ \cos(\beta) & 0 & \sin(\beta) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix}$ part is for reflecting the position rotated based on each of the roll, pitch, and yaw axes, and the $\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix}$ part is for reflecting the position moved to the focal point. In this way, after transforming the coordinate values (x_p, y_p, z_p) of the 3D orthogonal coordinate system of the corresponding pixel (P) among the pixels for virtual data into coordinate values (x_p', y_p', z_p') corresponding to coordinates in a 3D world coordinate system, the world coordinate system transformation module (221) can calculate the 3D coordinate values (x_m, y_m, 0) (e.g., 3D preliminary mesh coordinates) of the point (M) where the corresponding pixel (P) meets the mesh along the line of sight, i.e., the point (M) where the image is focused on the mesh (S30) (see FIG. 8).

The coordinate values (x_m, y_m, 0) of the 3D preliminary mesh coordinates can be calculated using the relationship between the coordinate values (x_t, y_t, z_t) of a point (T) located on the same line of sight as the corresponding pixel (P) and the graph of the mesh (z=0), and the coordinate values (x_t, y_t, z_t) of point (T) can be calculated according to [Mathematical Formula 2].

In [Mathematical Formula 2],

t is a variable.

Substituting the relationship of [Mathematical Formula 2] into the mesh function z = f(x, y) results in an equation for t, and the value of t for the point (T) where the mesh meets can be obtained from this equation. For example, in z = f(x, y), x=x0+(xp'-x0)t can be substituted for x and y=y0+(yp'-y0)t can be substituted for y. In the case of a 2D mesh where z = 0, t =z0/(z0 - zp') can be obtained.

Through such operation, 3D preliminary mesh coordinate values (x_m, y_m, 0) for at least one pixel among pixels for each virtual data are calculated and the world coordinate system transformation module (221) outputs all calculated 3D preliminary mesh coordinates for each pixel for virtual data to the shooting range expansion module (222).

The shooting range expansion module (222) is for changing the position of all 3D preliminary mesh coordinates for each pixel for virtual data input by extending the position along the line of sight.

As already described, in this example, since the mesh is the ground and the image alignment for the ground is accurate, in this example, the world coordinate system transformation module (221) can use a 2D mesh (i.e., ground) instead of a 3D mesh. Therefore, the amount of data to be processed is greatly reduced and the operation speed of the world coordinate system transformation module (221) in this example is greatly improved compared to when using a 3D mesh.

However, in this example, when cameras (11 to 14) use a fisheye lens, the field of view can be more than 180 degrees and the shooting range is greatly expanded compared to a normal lens.

Therefore, in this example, by using the shooting range expansion module (222) to change the position of the 3D preliminary mesh coordinates created along the line of sight outward, i.e., in a direction away from the focal position, it is possible to increase the shooting distance of the virtual camera and increase the shooting range of virtual data acquired by the virtual camera due to such an increase in shooting distance. In this way, 3D preliminary mesh coordinates with an increased shooting range become 3D actual mesh coordinates that are actually according to the world coordinate system.

Therefore, in this example, the shooting range expansion module (222), as shown in FIG. 11, can calculate the final distance (r') from the origin to the final 3D actual mesh coordinate (Me) that has been moved to obtain the coordinate values of the 3D actual mesh coordinate (M_e) whose position has been moved along the line of sight for the corresponding pixel (P) having coordinate values (x_m, y_m, 0) of the 3D preliminary mesh coordinate (M).

Therefore, the shooting range expansion module (222) can transform the coordinate values (x_m, y_m, 0) of the 3D preliminary mesh coordinate (M) for the corresponding pixel (P) to the position of the final distance (r') to calculate the coordinate values (x_{me}, y_{me}, 0) of the 3D actual mesh coordinate (M_e) (S40).

In FIG. 11, $r'=r(1+kr^2)$, where k is a constant and r is the distance from the focal point to the preliminary 3D mesh coordinate.

By the coordinate values (xme, yme, 0) of the 3D actual mesh coordinates (Me), the shooting distance of the corresponding virtual data can be increased, so that the effect of an image using a fisheye lens can be achieved while maintaining alignment with the ground, enabling the acquisition of a surround view image that does not deviate from the ground.

Thus, when correction for increasing the shooting distance for the coordinate values (x_m, y_m, 0) of the 3D preliminary mesh coordinates (M) is performed, the shooting range expansion module (222) can output the coordinate values (x_{me}, y_{me}, 0) of the 3D actual mesh coordinates (Me) for each virtual data pixel to a plurality of FOV expansion modules (2231-2234).

In this example, the number of FOV expansion modules (2231-2234) is equal to the number of cameras (11 to 14) mounted on the vehicle (100).

The structure of each FOV expansion module (2231-2234) is identical except for the type of coordinate values (x_{me}, y_{me}, 0) of the input 3D actual mesh coordinates.

In FIG. 2, natural numbers (1,2, 3, 4,…, N) input to each FOV expansion module (2231-2234) are identification numbers for identifying corresponding cameras (11 to 14).

Therefore, by such identification numbers, when generating a complete surround view image, the composition position of each created partial surround view image data (i.e., first to fourth virtual data) can be determined.

As a result, the first field of view(FOV) expansion module (2231) inputs 3D actual mesh coordinates for pixels for the first virtual data corresponding to the first camera (11), the second FOV expansion module (2232) inputs 3D actual mesh coordinates for pixels for the second virtual data corresponding to the second camera (12), the third FOV expansion module (2233) inputs 3D actual mesh coordinates for pixels for the third virtual data corresponding to the third camera (13), and the fourth FOV expansion module (2234) inputs 3D actual mesh coordinates for pixels for the fourth virtual data corresponding to the fourth camera (14).

However, in an alternative example where a 3D mesh located in a 3D space as a virtual surface instead of the ground is input to the world coordinate system transformation module (221), the shooting range expansion module (222) in this example can be omitted. In this case, the 3D mesh coordinates output from the world coordinate system transformation module (221) become 3D actual mesh coordinates and are input to corresponding FOV expansion modules (2231-2234).

Also, as already described, when generating an entire surround view image, due to the characteristics of a 2D mesh, only a part of an object located on a stitch line, such as the bottom part, may be visible in the created entire surround view image.

When generating entire surround view image data based on each captured data acquired by a plurality of cameras (11 to 14) and each virtual data, the entire surround view image data includes a plurality of partial surround view image data created based on captured data acquired by at least two different adjacent cameras (11 to 14).

Therefore, the entire surround view image data including a plurality of partial surround view image data can be output to the information output unit (24) so that an entire surround view image for the vehicle (100) can be displayed.

In this example, partial surround view image data is corrected to correspond to image data for a field of view of 180 degrees, so that image loss of an object at the stitch line between two different partial surround view images is prevented.

Therefore, as shown in FIG. 5, in order to make the field of view range of each camera (e.g., 11) appear to have 180 degrees, each FOV expansion module (2231-2234) corrects the values of 3D actual mesh coordinates for corresponding pixels for virtual data so that the field of view of the virtual camera is expanded up to 180 degrees, and as a result, the effect of expanding the field of view of each camera (11 to 14) up to 180 degrees can be obtained (S50).

For this purpose, in this example, each virtual data pixel can be expanded by a predetermined angle so that the corresponding pixel has coordinates that move toward the 180-degree FOV boundary line (RL10) forming a 180-degree angle of view.

In this case, by the FOV expansion module (2231-2234), pixels located on the stitch line (SL10) can have their coordinates changed to be located on the 180-degree FOV boundary line (RL10).

In this example, since there are four cameras (11 to 14), there can also be four 180-degree FOV boundary line (RL10) for each camera (11 to 14).

In FIG. 5, the stitch line (SL10) is a stitch line (seam line) that occurs when different partial surround view image data is synthesized, and this stitch line (SL10) can generally be determined as a straight line extending along the corresponding surface of the vehicle (e.g., front, rear, left or right side) and forming a predetermined angle (ρ) with the 180-degree angle line (RL10).

The storage unit (23) stores an angle (ρ) for the stitch line (SL10), and as an example, the angle (ρ) formed with the 180-degree FOV boundary line (RL10) is approximately 45 degrees.

As a result, as shown in FIG. 5, for example, a virtual data pixel located on the stitch line (SL10) and having coordinates (a, b) can be moved to coordinates (a', b') by the operation of the corresponding FOV expansion module (2231-2234) and located on the 180-degree FOV boundary line (RL10), and a virtual data pixel located inside the stitch line (SL10) and having coordinates (c, d) can be moved to coordinates (c', d') by the operation of the corresponding FOV expansion module (2231-2234) and located between the 180-degree FOV boundary line (RL10) and the stitch line (SL10).

By the operation of such a FOV expansion module (2231-2234), the range of partial surround view images created based on shooting data acquired by first to fourth cameras (11 to 14) that can be respectively located at the front, rear, left side and right side of the vehicle (10) can have an effect of being extended to a 180-degree FOV boundary line (RL10) forming a 180-degree angle of view.

An example of an operation for changing the corresponding position of each virtual data pixel to the 180-degree FOV boundary line (RL10) in such a FOV expansion module (2231-2234) is described with reference to FIG. 12.

FIG. 12 is an example in which the front of the vehicle is taken as an example, and in FIG. 12, the center is the center of the world coordinate system having a coordinate value of (0, 0, 0) as a central point of the vehicle from the road surface or ground surface.

Also, in FIG. 12, the mathematical formula for transforming the position of a pixel (P) for virtual data having coordinates (x_{me}, y_{me}, 0) (Me) to have changed coordinates (E) toward the 180-degree FOV boundary line (RL10) is as follows.

[Mathematical formula 3]

$$l=\sqrt{x_{me}^{2}+(y_{me}-g)^{2}}$$

Here, l is the distance of a straight line connecting a virtual circle with a radius equal to the distance from the center (0, g, 0) of the circle to the position of the three-dimensional real mesh coordinates of the corresponding pixel (P) having coordinates (x_{me}, y_{me}, 0) from the center (0, g, 0) of the circle at the middle point of the front of the vehicle where the camera is

mounted to the circumference of the virtual circle, and x_{me} and y_{me} are real mesh coordinate values for each axis of X and Y of the corresponding pixel.

In FIG. 12, (f, g, 0) is coordinates of the right front corner of the vehicle in world coordinates. Here, f is the coordinate value of the right front corner when located on an (x, y) plane with the center of the vehicle at the origin and with its front facing in a 12 o'clock direction (or y-axis direction).

[Mathematical formula 4]

$$\theta = \arctan(y_{me} - g,\ x_{me})$$

Here, θ is the angle formed by a straight line (L1) connecting the 180-degree FOV boundary line and the pixel (P) having coordinates (x_{me}, y_{me}, 0).

[Mathematical formula 5]

Here, *w* is the weight and φ is the angle formed by a straight line (L2) connecting the center of the circle from the 180-degree FOV boundary line to the stitch line (SL10).

[Mathematical formula 6]

$$(f + l_2 \cos\rho)^2 + (l_2 \sin\rho)^2 = l^2$$
$$l_2^2 + 2f \cdot \cos\rho\ l_2 + f^2 - l^2 = 0$$
$$l_2 = -f \cdot \cos\rho + \sqrt{f^2 \cos\rho^2 + l^2 - f^2}$$
$$\phi = \tan^{-1}\left(\frac{l_2 \sin\rho}{f + l_2 \cos\rho}\right)$$

Here, σ is the angle formed by the 180-degree FOV boundary line and the stitch line, and *f* is a coordinate value of the right front corner when located on an (x, y) plane with the center of the vehicle at the origin and with its front facing in a 12 o'clock direction (or y-axis direction). l_2 is the length of the stitch line meeting the circumference.

[Mathematical formula 7]

$$\delta = \frac{\pi}{2} - \theta$$
$$\delta' = w \times \delta$$
$$\theta' = \frac{\pi}{2} - \delta'$$

[Mathematical formula 8]

$$x_e = l \times \cos\theta'$$
$$y_e = l \times \sin\theta' + g$$

By the operation of each FOV expansion module (2231-2234), three-dimensional expanded mesh coordinates (x_e, y_e, z_e) [specifically, ($x1_e$, $y1_e$, $z1_e$), ($x2_e$, $y2_e$, $z2_e$), ($x3_e$, $y3_e$, $z3_e$), ($x4_e$, $y4_e$, $z4_e$)] moved toward the 180-degree FOV boundary line (RL10) for each virtual data (i.e., first to fourth virtual data) pixel can be output and input to each camera coordinate system transformation module (2241-2244).

Here, ($x1_e$, $y1_e$, $z1_e$) is the 3D expanded mesh coordinates for the first coordinate transformation table, ($x2_e$, $y2_e$, $z2_e$) is the 3D expanded mesh coordinates for the second coordinate transformation table, ($x3_e$, $y3_e$, $z3_e$) is the 3D expanded mesh coordinates for the third coordinate transformation table, and ($x4_e$, $y4_e$, $z4_e$) is the 3D expanded mesh coordinates for the fourth coordinate transformation table.

Next, each camera coordinate system transformation module (2241-2244) of this example transforms the coordinates of the corresponding pixel for virtual data (i.e., first to fourth virtual data) corresponding to each shooting data of the camera (11 to 14) into coordinates of the orthogonal coordinate system of the camera (S60), ultimately to convert the shooting image of each shooting data acquired by the corresponding camera (11 to 14) into a partial surround view image.

FIG. 13 is a diagram schematically illustrating the relationship between the world coordinate system and the camera coordinate system.

Therefore, [Mathematical Formula 9] for transforming the coordinate value (x_e, y_e, 0) of point (E) in FIG. 13 into the coordinate value ($x_{e'}$, $y_{e'}$, $z_{e'}$) of the orthogonal coordinate system of the

camera to transform the coordinates of the pixel located at the corresponding point (E) into coordinates of the camera orthogonal coordinate system is as follows.

[Mathematical formula 9]

$$\begin{bmatrix} x_{e'} \\ y_{e'} \\ z_{e'} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(-\alpha) & -\sin(-\alpha) \\ 0 & \sin(-\alpha) & \cos(-\alpha) \end{bmatrix} \begin{bmatrix} -\sin(-\beta) & 0 & \cos(-\beta) \\ 0 & 1 & 0 \\ \cos(-\beta) & 0 & \sin(-\beta) \end{bmatrix} \times \begin{bmatrix} \cos(-\gamma) & -\sin(-\gamma) & 0 \\ \sin(-\gamma) & \cos(-\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \left(\begin{bmatrix} x_e \\ y_e \\ 0 \end{bmatrix} - \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} \right)$$

In [Mathematical formula 9], (x_c, y_c, z_c) is the world coordinate system coordinates of the camera focus.

Next, each camera coordinate system transformation module (2241-2244) can transform the orthogonal coordinate system of the corresponding camera (11 to 14) into a spherical coordinate system (S70).

For this purpose, location information of each corresponding camera (11 to 14) can be additionally input to each camera coordinate system transformation module (2241-2244). As already described, in this example, location information of each camera (11 to 14) may be a center point, coordinates of a located point, a central angle and a rotation angle of the corresponding camera (11 to 14).

The rotation angle is a rotation angle (α) based on the roll axis, a rotation angle (β) based on the pitch axis, and a rotation angle (γ) based on the yaw axis.

Therefore, each camera coordinate system transformation module (2241-2244) can transform the input corresponding coordinates of the corresponding pixel for virtual data, i.e., coordinates [($x1_e$', $y1_e$', $z1_e$'), ($x2_e$', $y2_e$', $z2_e$'), ($x3_e$', $y3_e$', $z3_e$'), ($x4_e$', $y4_e$', $z4_e$')] based on the orthogonal coordinate system into spherical coordinates according to the spherical coordinate system and calculate the incident angle and azimuth angle of each camera (11 to 14) (S70).

And then, using the calculated incident angle and azimuth angle, transform all pixel coordinates of each shooting data, i.e., 2D coordinates [(x1, y1), (x2, y2), (x3, y3), (x4, y4)]

into coordinate values that determine the coordinate position of the corresponding pixel of the shooting data corresponding to the corresponding pixel for virtual data and store them in the storage unit (23) (S80).

Next, each camera coordinate system transformation module (2241-2244) can read weights (w1-w4) determined according to the corresponding coordinates of the corresponding pixel for virtual data from the storage unit (23), and store them in table form along with the coordinate values of the corresponding pixel of the shooting data corresponding to each pixel for virtual data to create a coordinate transformation table (S90).

At this time, each camera coordinate system transformation module (2241-2244) can store the created coordinate transformation table in the storage unit (23) (S90).

As a result, the coordinate values and weights (w1-w4) corresponding to each pixel of the corresponding shooting data corresponding to each pixel of the first to second virtual data can be respectively stored to create first to fourth coordinate transformation tables.

By the coordinate transformation table, it is possible to know which pixel of the corresponding shooting data is mapped to each pixel for virtual data in virtual data and shooting data that correspond to each other.

By each coordinate transformation table, the pixel value of each pixel for virtual data is determined by the pixel value of the corresponding pixel of the shooting data at the mapped position, and virtual data having a determined pixel value for each pixel can be created, and such created virtual data will be one of multiple surround view image data.

Thus, when a coordinate transformation table corresponding to each virtual data is created by the coordinate transformation table creation part (2201), the virtual data creation part (2301) can determine the pixel value of each pixel for virtual data using the corresponding coordinate transformation table and create each virtual data.

Next, referring to FIG. 3, the structure and operation of such a virtual data creation part (2301) will be described.

As shown in FIG. 3, the virtual data creation part (2301) includes a corresponding coordinate search module (231), at least one shooting data substitution module (2321-2324) connected to

the corresponding coordinate search module (231), at least one multiplier (2331-2334) connected to each shooting data substitution module (2321-2324), and an adder (234) connected to each multiplier (2331-2334).

The corresponding coordinate search module (231) can input the coordinate values (n, k) of the coordinates of each pixel for virtual data and read the coordinate values [(x_1,y_1), (x_2,y_2), (x_3,y_3),or (x_4,y_4)] and weights (w1, w2, w3, or w4) of the corresponding pixel of the corresponding shooting data corresponding to the coordinate value (n, k) of the position (i.e., coordinate) of the input pixel using the corresponding coordinate transformation table and input them to the shooting data substitution module (2321-2324).

At this time, the corresponding coordinate search module (231) can search for the coordinate value of the corresponding pixel of the corresponding shooting data and output the coordinate value [(x_1,y_1), (x_2,y_2), (x_3,y_3),or (x_4,y_4)] mapped to the coordinate (n, k) of the corresponding pixel for each virtual data as shown. In FIG. 3, the subscripts (1-4) of [(x_1,y_1), (x_2,y_2), (x_3,y_3), (x_4,y_4)] indicates the type of shooting data (i.e., one of first to fourth shooting data), where subscript '1' indicates first shooting data, subscript '2' indicates second shooting data, subscript '3' indicates third shooting data, and subscript '4' indicates fourth shooting data.

For example, if the coordinate (n,k) of the pixel for the first virtual data is (1,1), i.e., a pixel existing in the first row and first column, and the coordinate value of the corresponding pixel of the corresponding shooting data (i.e., first shooting data) mapped to this coordinate (1,1) in the first coordinate transformation table for the first virtual data is (1,5), i.e., a pixel existing in the first row and fifth column, the corresponding coordinate search module (231) can search for a pixel having coordinates (1,5) in the first shooting data corresponding to the first virtual data as a pixel mapped to a pixel having coordinates (1,1) of the first virtual data.

In this way, the coordinate values [(x_1,y_1), (x_2,y_2), (x_3,y_3) or (x_4,y_4)] and corresponding weights (w1, w2, w3 or w4) of pixels of the corresponding shooting data mapped to the input corresponding coordinates (n,k) of the corresponding pixel for virtual data can be input to each corresponding shooting data substitution module (2321-2324).

Each shooting data substitution module (2321-2324) is for creating corresponding virtual data using shooting data created by each camera (11 to 14).

Each shooting data substitution module (2321-2324) can search for pixels having corresponding coordinate values [(x_1,y_1), (x_2,y_2), (x_3,y_3), (x_4,y_4)] in the corresponding shooting data and read the pixel value of the searched pixel and determine the pixel value [$P_1(x_1,y_1)$, $P_2(x_2,y_2)$, $P_3(x_3,y_3)$, $P_4(x_4,y_4)$] for each pixel of the first to second preliminary virtual data and store it in the storage unit (23) and output it to the multiplier (2331-2334).

Each multiplier (e.g., first to fourth multipliers) (2331-2334) can be connected to each shooting data substitution module (2321-2324) and multiply the pixel value [$P_1(x_1,y_1)$, $P_2(x_2,y_2)$, $P_3(x_3,y_3)$, $P_4(x_4,y_4)$] of each preliminary virtual data input from the corresponding shooting data substitution module (2321-2324) by the weight (w1-w4) determined for each pixel to finally determine the pixel value of each pixel of each first to fourth virtual data and store it in the storage unit (23).

The value of the weight (w1-w4) multiplied by the pixel value [$P_1(x_1,y_1)$, $P_2(x_2,y_2)$, $P_3(x_3,y_3)$, $P_4(x_4,y_4)$] of each preliminary virtual data to determine the pixel value of each pixel of each virtual data can have a value between 0 and 1, and the value of the weight can be determined according to the position of each pixel of each preliminary virtual data.

For example, depending on whether a pixel belongs to a shooting range (e.g., front, rear, left side or right side) of a camera corresponding to each preliminary virtual data, the value of the weight will be determined. As an example, when creating first virtual data using first shooting data, if a pixel of first preliminary virtual data is a pixel located within the shooting range of first camera (11), the weight will be '1', if it is a pixel located outside the shooting range, the weight will be '0', and if it is a pixel located on a stitch line of the shooting range of first camera (11), it will have a value greater than 0 and less than 1.

In this way, each first to fourth shooting data substitution module (2321-2324) can read a matching pixel position in a corresponding shooting data (one of first to fourth shooting data) determined in a corresponding coordinate transformation table (one of first to fourth coordinate transformation tables) and read a pixel value at a read position to determine a pixel value for a corresponding position (n,k) of a corresponding preliminary virtual data (one of first to fourth preliminary virtual data).

Then, each first to fourth multiplier (2331-2334) can multiply a corresponding weight (w1,w2,w3 or w4) for a corresponding pixel of a corresponding preliminary virtual data by determining a pixel value for a corresponding position (n, k) of a corresponding virtual data.

Therefore, through such operation, all pixels values are determined for first to fourth virtual data and as a result, first to fourth virtual data can be created and stored in storage unit (23).

In this way, when first to fourth virtual data are created by first to fourth multipliers (2331-2334), first to fourth virtual data can be input to an adder (234) connected to first to fourth multipliers (2331-2334).

Therefore, the adder (234) can add all pixel values of the corresponding position (n, k) of first to fourth virtual data output from each multiplier (2331-2334) to calculate the pixel value of the corresponding pixel of the entire surround view image data having the coordinate of the corresponding position (n, k) and store it in the storage unit (23).

In this way, all pixel values of the entire surround view image data can be determined and the entire surround view image data can be created and output to the information output unit (24) to provide a complete surround view image to the driver.

FIG. 6 illustrates an example of a complete surround view image output to an information output unit.

FIG. 6(a) and (b) are images in which only a part of a complete surround view image created using virtual data of a comparative example, i.e., a rear image and a right side image, are expanded to 180 degrees, respectively, and FIG. 6(c) is a complete surround view image created using virtual data created according to the present invention.

Therefore, FIG. 6(a) shows that the rear surround view image (IMG11) created using shooting data from a camera attached to the rear of the vehicle is expanded toward the 180-degree FOV boundary line, and FIG. 6(b) shows that the right side surround view image (IMG14) created using shooting data from a camera attached to the right side of the vehicle is expanded toward the 180-degree FOV boundary line.

FIG. 6(c) shows that by the operation of the FOV expansion module (2231-2234) of the present invention, all partial surround view images, i.e., front, rear, left side and right side

surround view images are expanded toward the 180-degree FOV boundary line, so that the range displayed in the entire surround view image increases compared to before.

Therefore, when comparing the rear surround view image (IMG21) and right side surround view image (IMG24) of the vehicle with images (IMG11, IMG14) in FIG. 6(a) and (b), respectively, it can be seen that they are similar or identical and as a result, all objects are output without loss of objects in parts of stitch line (SL10) of different partial surround view images.

The technical features disclosed in each embodiment of the present invention are not limited to that embodiment and can be applied by merging with other embodiments as long as they are not mutually incompatible.

Therefore, although each technical feature is mainly described in each embodiment, it can be merged and applied with each other as long as it is not mutually incompatible.

The present invention is not limited to the above-described embodiments and accompanying drawings and various modifications and variations are possible from the perspective of a person having ordinary knowledge in the field to which the present invention belongs. Therefore, the scope of the present invention should be determined not only by the claims but also by equivalents thereof.

[Description of Symbols]

11 to 14: Cameras 20: Image processing apparatus

21: Communication unit 22: Operation unit

2201: Coordinate transformation table creation part

2301: Virtual data creation part

231: Corresponding coordinate search module

2321-2324: Shooting data substitution module

2331-2334: Multiplier 234: Adder

23: Storage unit

24: Information output unit

[Claims]

[Claim 1]

The image processing apparatus for surround view images, comprising:

a coordinate transformation table creation part including a field of view (FOV) expansion module that transforms the coordinates of virtual data pixels corresponding to shooting data into 3D mesh coordinates, and changes the position of the 3D mesh coordinates to an expanded position in the direction of a 180-degree FOV boundary line to create 3D expanded mesh coordinates.

[Claim 2]

The image processing apparatus for surround view images, as claimed in Claim 1, further comprising:

The FOV expansion module that transforms the 3D mesh coordinates of pixels into 3D expanded mesh coordinates according to [Mathematical Formula 1] to [Mathematical Formula 5].

[Mathematical Formula 1]

$$l=\sqrt{x_{me}^{2}+(y_{me}-g)^{2}}$$

(Where, x_{me} and y_{me} are the actual mesh coordinate values for the X-axis and Y-axis of the pixel, respectively, and l is the distance from the center of a virtual circle with the middle point of the front of the vehicle where the camera is mounted as the center of the circle to a straight line connecting the circumference of the circle with a radius equal to the distance from the center of the circle to the position of the 3D actual mesh coordinate of a pixel having a coordinate value of (x_{me}, y_{me}, 0))

[Mathematical Formula 2]

$$\theta = \arctan(y_{me}\text{-}g, x_{me})$$

(Where θ is the angle formed by a straight line connecting a 180-degree FOV boundary line and a pixel having a coordinate value of (x_{me}, y_{me}, 0))

[Mathematical Formula 3]

$$w = \frac{\frac{\pi}{2}}{\frac{\pi}{2} - \phi}$$

(Where *w* is a weight and φ is an angle formed by a straight line connecting a 180-degree FOV boundary line and a stitch line from the center of a circle)

[Mathematical Formula 4]

$$\delta = \frac{\pi}{2} - \theta$$
$$\delta' = w \times \delta$$
$$\theta' = \frac{\pi}{2} - \delta'$$

[Mathematical Formula 5]

$$x_e = l \times \cos\theta'$$
$$y_e = l \times \sin\theta' + g$$

(Where (x_e, y_e) are the X-axis value and Y-axis value of the 3D expanded mesh coordinate)

[Claim 3]

The image processing apparatus for surround view images, as claimed in Claim 2, further comprising:

the coordinate transformation table creation part, the part includes;

the world coordinate system transformation module, wherein

inputs coordinate values of virtual data pixels, 2D mesh coordinates and virtual camera position information to create 3D coordinates for virtual data pixels by transforming the coordinates of 2D virtual data pixels into coordinates of a 3D orthogonal coordinate system,

transforms the 3D coordinates of virtual data pixels into coordinates of a 3D world coordinate system, and

transforms the coordinates of pixels in the 3D world coordinate system into coordinates of points where they meet the mesh to generate 3D preliminary mesh coordinates; and

the shooting range expansion module,

connected to the world coordinate system transformation module,

changes the position of the 3D preliminary mesh coordinates applied from the world coordinate system transformation module in the outward direction along the optical axis to calculate the coordinate values of the 3D actual mesh coordinates.

[Claim 4]

The image processing apparatus for surround view images as claimed in Claim 3, further comprising:

the shooting range expansion module that calculates the position (r') of the transformed coordinates using the following mathematical formula.

$$r' = r(1+kr^2)$$

(Where k is a constant and r is the distance from the focus to the preliminary 3D mesh coordinate)

[Claim 5]

The image processing apparatus for surround view images as claimed in Claim 1, further comprising:

the coordinate transformation table creation part further includes a camera coordinate system transformation module for

transforming the coordinates of virtual data pixels having 3D expanded mesh coordinates into orthogonal coordinates of an image acquisition unit,

transforming orthogonal coordinates into spherical coordinates again to calculate incidence angles and azimuth angles of an image acquisition unit that acquired shooting data,

transforming all pixels of shooting data into corresponding coordinates using calculated incidence angles and azimuth angles,

determining coordinate values of corresponding pixels of shooting data corresponding to virtual data pixels according to determined weights and determined coordinate values according to coordinates of virtual data pixels, and

generating a coordinate transformation table using determined weights and determined coordinate values.

[Claim 6]

The image processing apparatus for surround view images as claimed in Claim 5, further comprising:

the virtual data creation part, connected to the camera coordinate system transformation module, determines the pixel values of each virtual data pixel to create virtual data using the coordinate transformation table.

[Claim 7]

The image processing apparatus for surround view images as claimed in Claim 6, further comprising:

the virtual data creation part includes:

the corresponding coordinate search module, wherein inputting the coordinate values of the virtual data pixels, reading and outputting the coordinate values and weights of the pixels of shooting data corresponding to the input pixel coordinates are performed;

the shooting data substitution module, wherein, connected to the corresponding coordinate search module, searching for pixels of shooting data having the coordinate values output by the corresponding coordinate search module, reading and outputting the pixel values of the searched pixels are performed; and

the multiplier, wherein, connected to the shooting data substitution module, calculation for the pixel values of virtual data pixels is performed by multiplying the pixel values applied from each shooting data substitution module by weights corresponding to virtual data pixels.