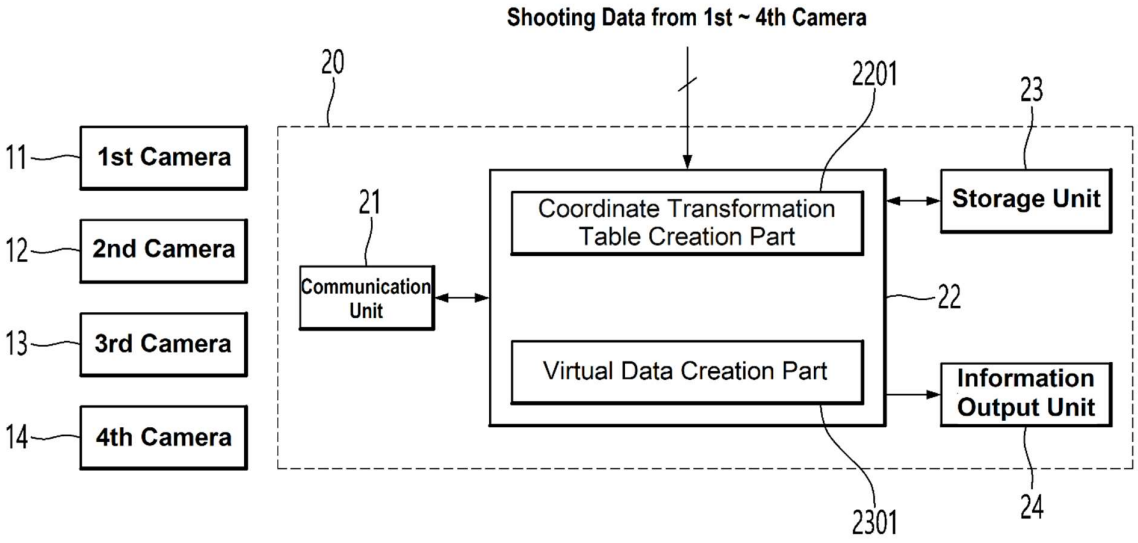
FIG.1.


Shooting Data from 1st ~ 4th Camera
20
2201
23
11
1st Camera
12
2nd Camera
13
3rd Camera
14
4th Camera
21
Communication Unit
Coordinate Transformation Table Creation Part
Virtual Data Creation Part
22
Storage Unit
Information Output Unit
2301
24

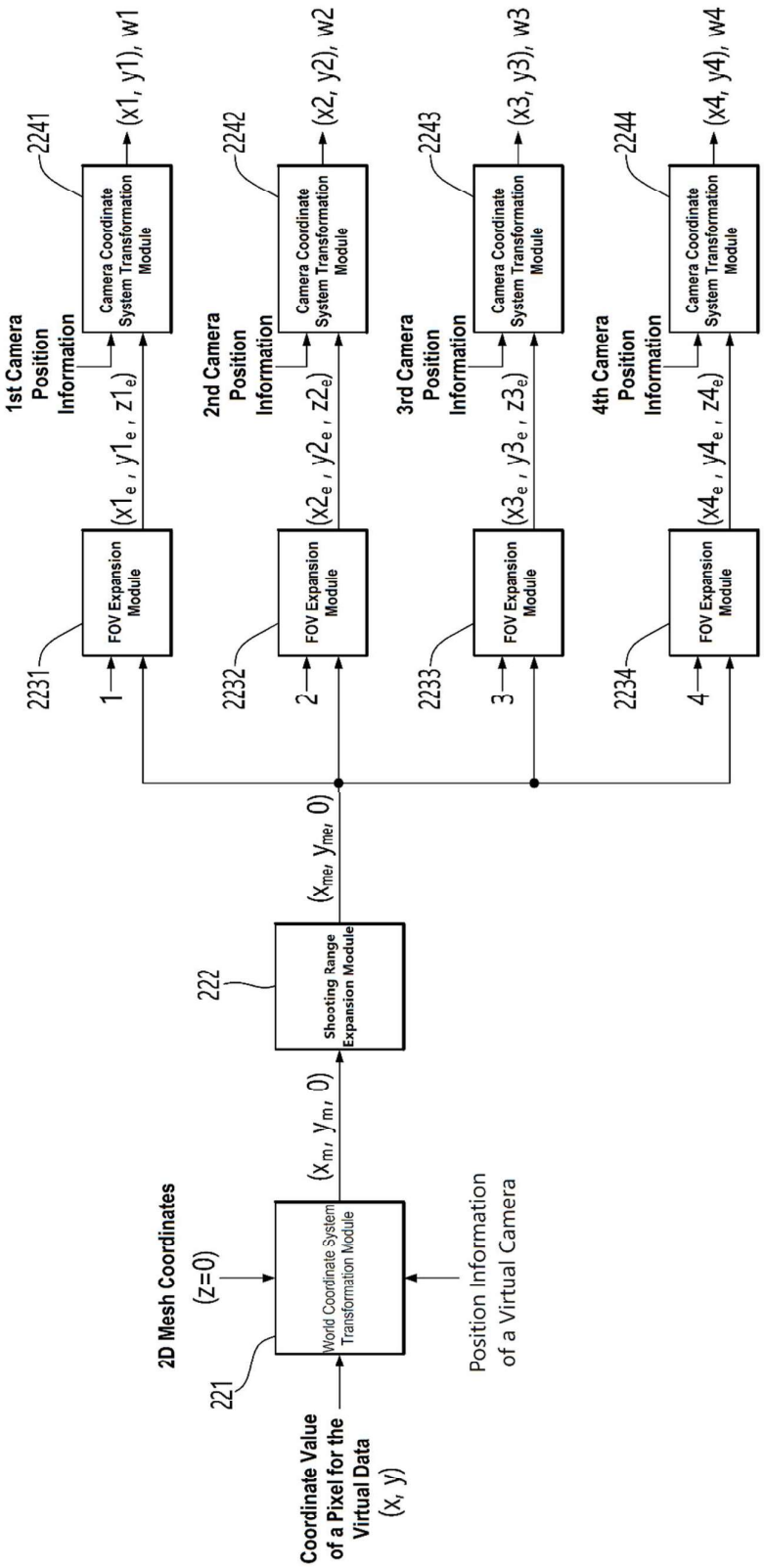

Coordinate Value of a Pixel for the Virtual Data (x, y)
2D Mesh Coordinates (z=0)
221
World Coordinate System Transformation Module
Position Information of a Virtual Camera
(xm, ym, 0)
222
Shooting Range Expansion Module
(xme, yme, 0)
1
2
3
4
2231
2232
2233
2234
FOV Expansion Module
(x1e , y1e , z1e)
(x2e , y2e , z2e)
(x3e , y3e , z3e)
(x4e , y4e , z4e)
1st Camera Position Information
2nd Camera Position Information
3rd Camera Position Information
4th Camera Position Information
2241
2242
2243
2244
Camera Coordinate System Transformation Module
(x1, y1), w1
(x2, y2), w2
(x3, y3), w3
(x4, y4), w4

FIG.2.

FIG.3.

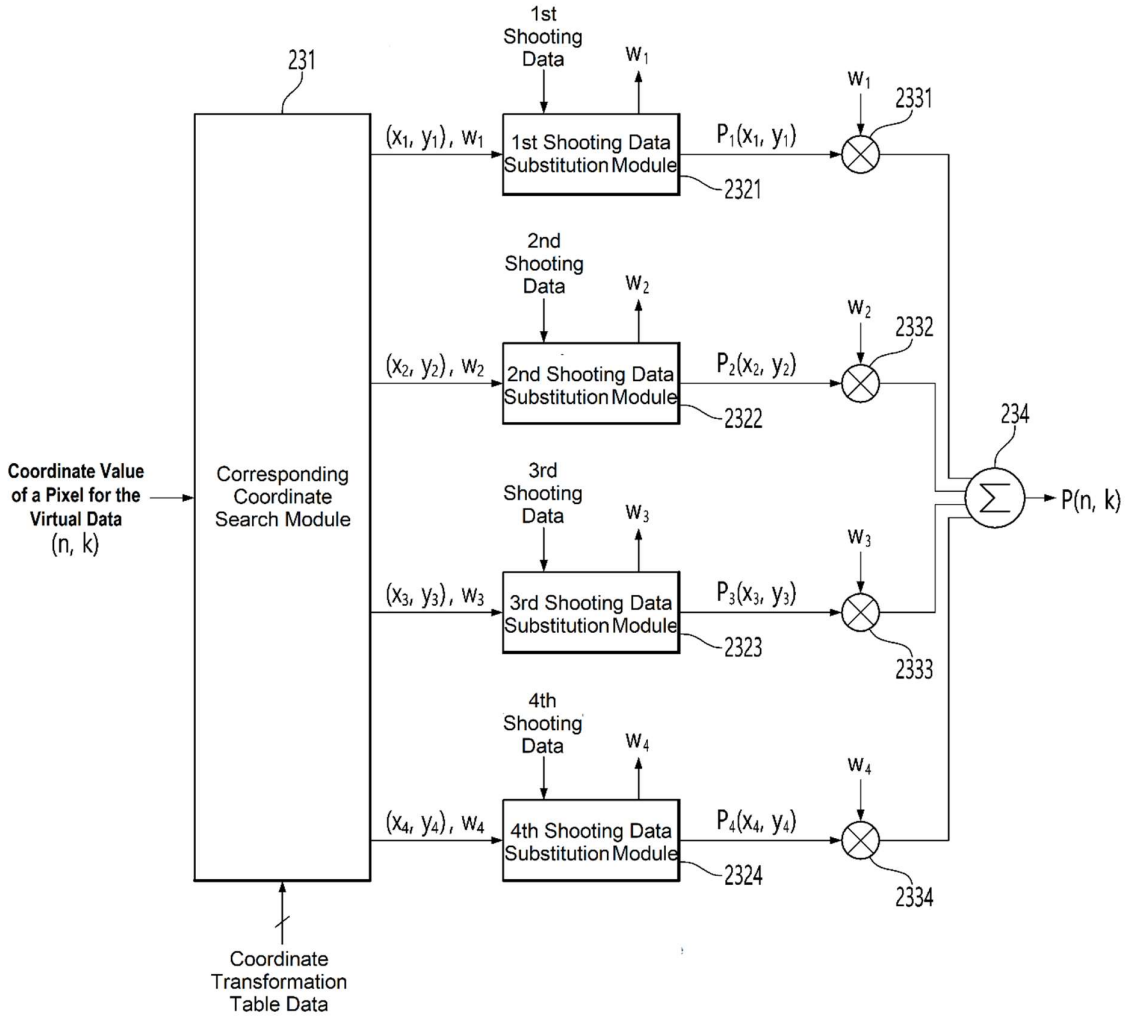

231
1st Shooting Data
W_1
W_1
2331
(x_1, y_1), W_1
1st Shooting Data Substitution Module
$P_1(x_1, y_1)$
2321
2nd Shooting Data
W_2
W_2
2332
(x_2, y_2), W_2
2nd Shooting Data Substitution Module
$P_2(x_2, y_2)$
2322
234
Coordinate Value of a Pixel for the Virtual Data
(n, k)
Corresponding Coordinate Search Module
3rd Shooting Data
W_3
W_3
(x_3, y_3), W_3
3rd Shooting Data Substitution Module
$P_3(x_3, y_3)$
2323
2333
P(n, k)
4th Shooting Data
W_4
W_4
(x_4, y_4), W_4
4th Shooting Data Substitution Module
$P_4(x_4, y_4)$
2324
2334
Coordinate Transformation Table Data

FIG.4.

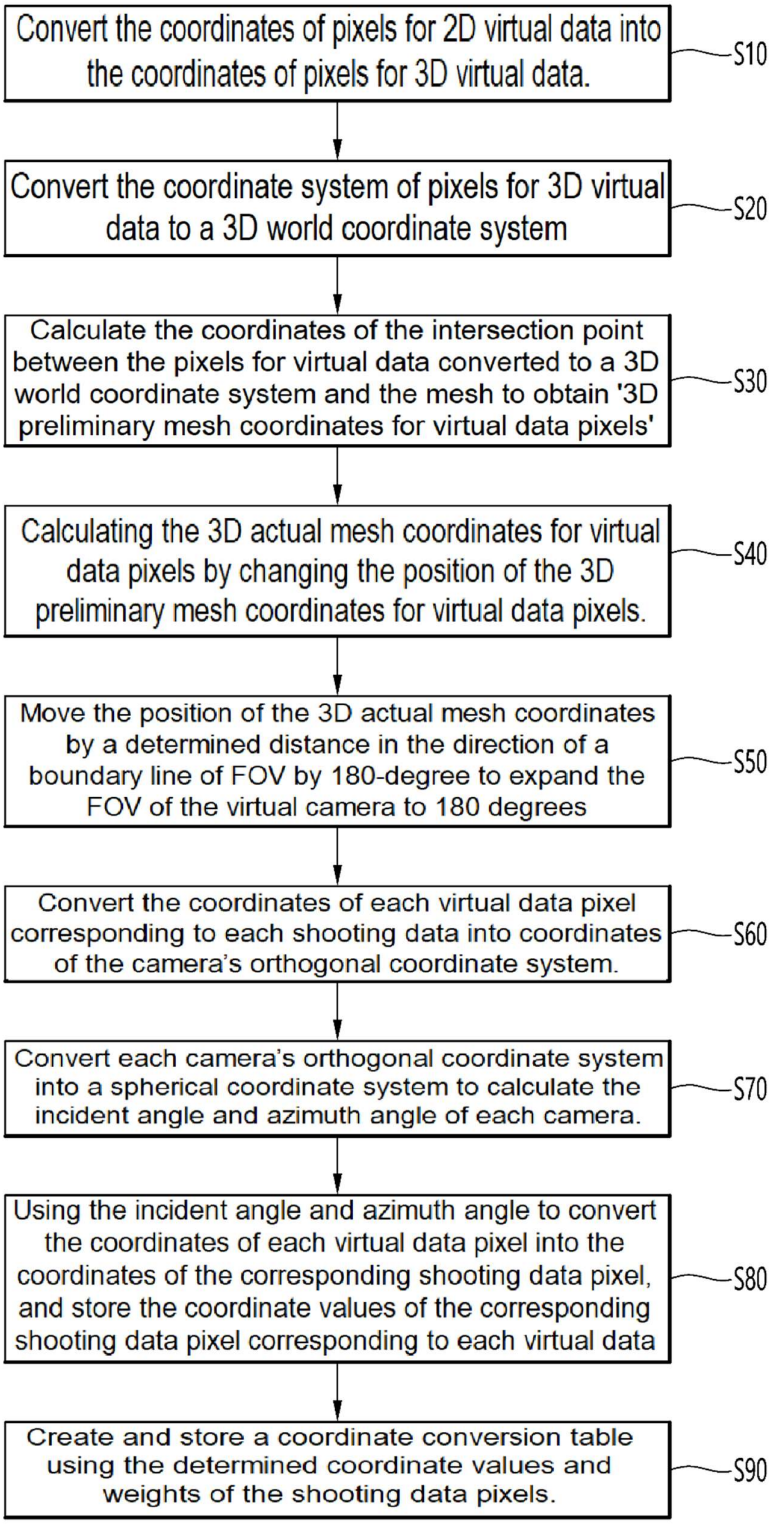

Convert the coordinates of pixels for 2D virtual data into the coordinates of pixels for 3D virtual data.
S10
Convert the coordinate system of pixels for 3D virtual data to a 3D world coordinate system
S20
Calculate the coordinates of the intersection point between the pixels for virtual data converted to a 3D world coordinate system and the mesh to obtain '3D preliminary mesh coordinates for virtual data pixels'
S30
Calculating the 3D actual mesh coordinates for virtual data pixels by changing the position of the 3D preliminary mesh coordinates for virtual data pixels.
S40
Move the position of the 3D actual mesh coordinates by a determined distance in the direction of a boundary line of FOV by 180-degree to expand the FOV of the virtual camera to 180 degrees
S50
Convert the coordinates of each virtual data pixel corresponding to each shooting data into coordinates of the camera's orthogonal coordinate system.
S60
Convert each camera's orthogonal coordinate system into a spherical coordinate system to calculate the incident angle and azimuth angle of each camera.
S70
Using the incident angle and azimuth angle to convert the coordinates of each virtual data pixel into the coordinates of the corresponding shooting data pixel, and store the coordinate values of the corresponding shooting data pixel corresponding to each virtual data
S80
Create and store a coordinate conversion table using the determined coordinate values and weights of the shooting data pixels.
S90

FIG.5.

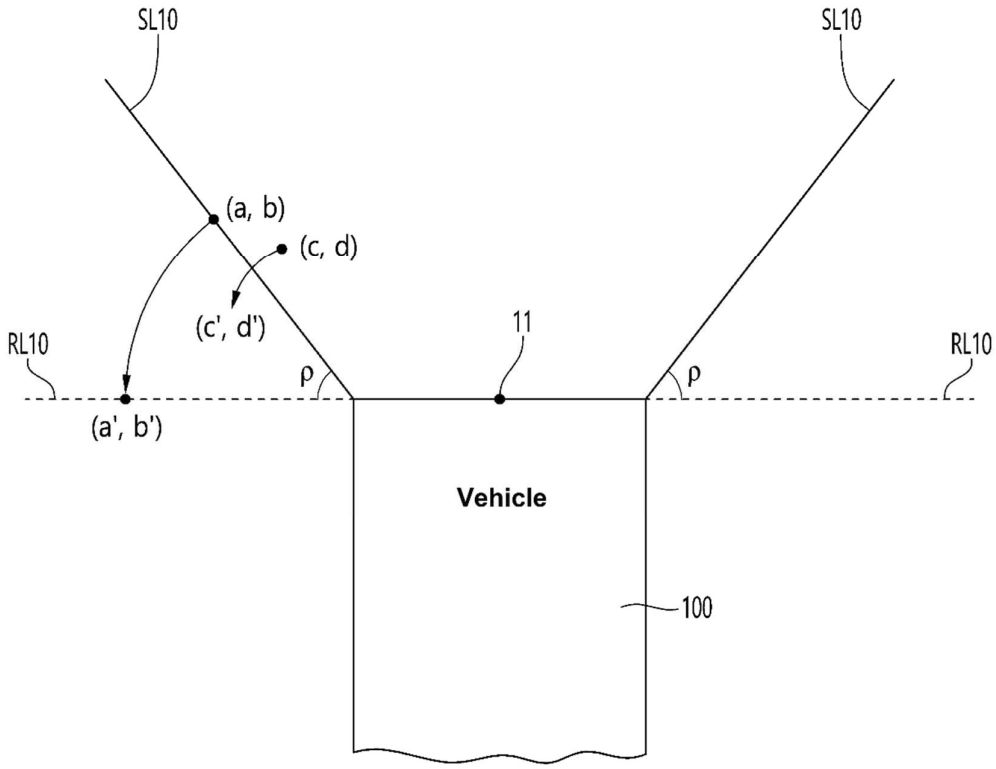

SL10
SL10
(a, b)
(c, d)
(c', d')
11
RL10
RL10
ρ
ρ
(a', b')
Vehicle
100

FIG.6.

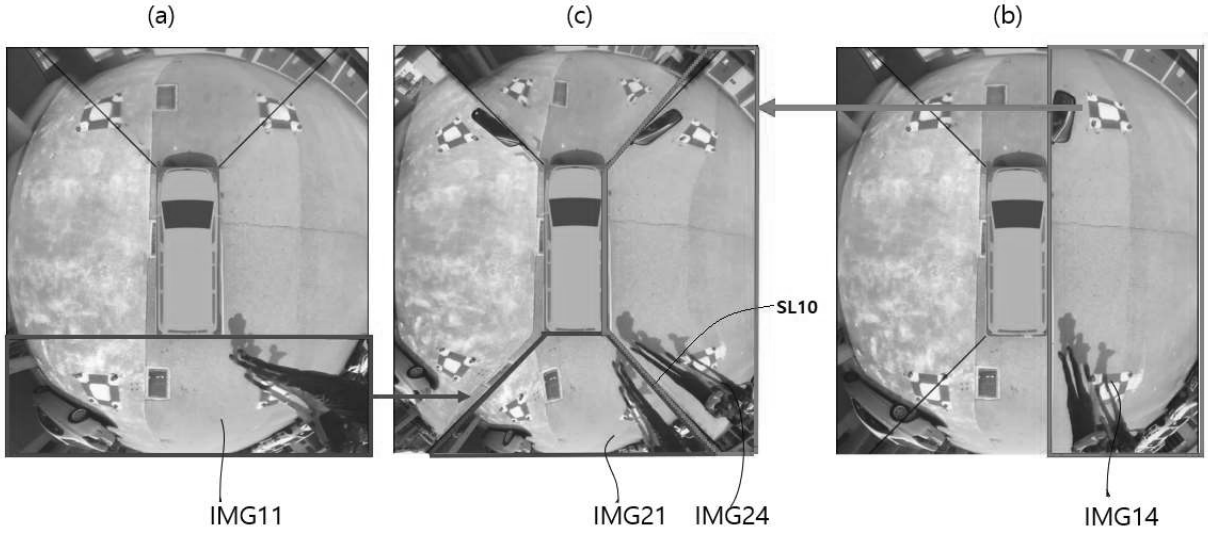

(a)
(c)
(b)
SL10
IMG11
IMG21
IMG24
IMG14

FIG.7.

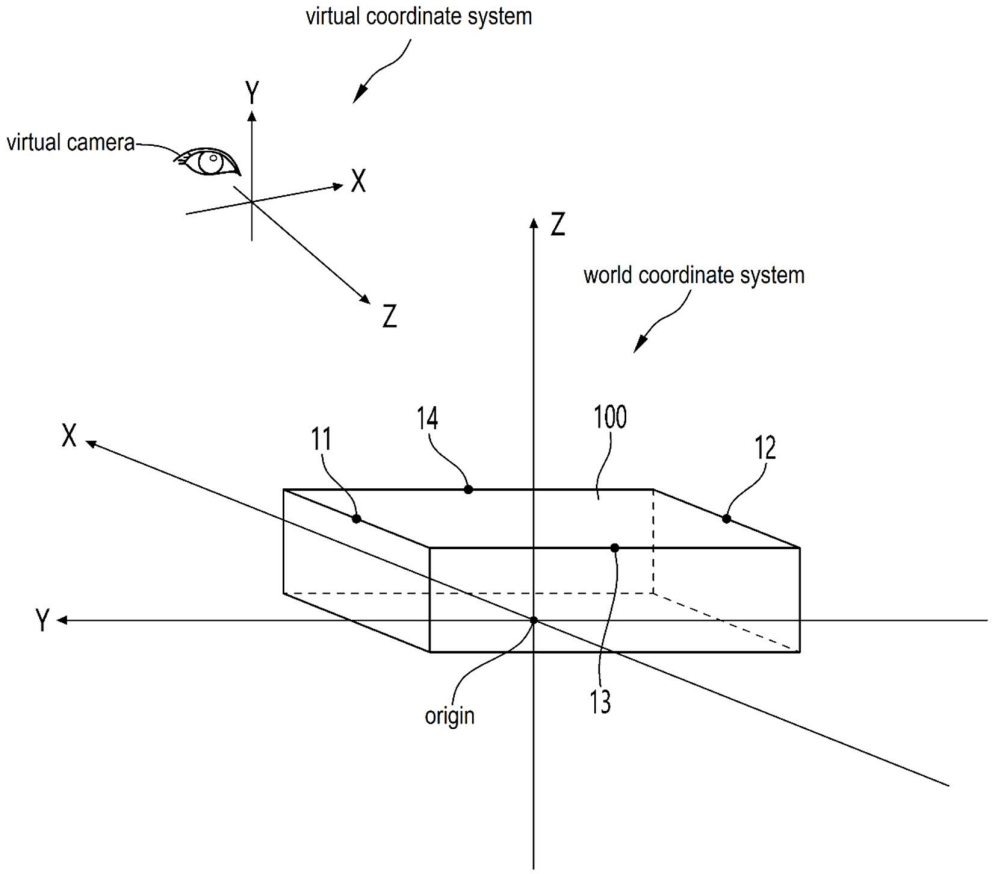

virtual coordinate system
Y
virtual camera
X
Z
Z
world coordinate system
X
11
14
100
12
Y
origin
13

FIG.8.

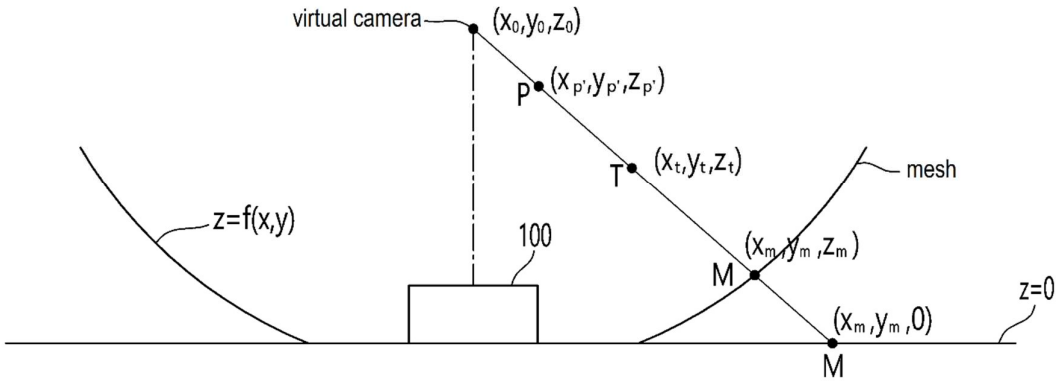

virtual camera
(x_0, y_0, z_0)
P
$(x_{p'}, y_{p'}, z_{p'})$
T
(x_t, y_t, z_t)
mesh
z=f(x,y)
100
(x_m, y_m, z_m)
M
z=0
$(x_m, y_m, 0)$
M

FIG.9.

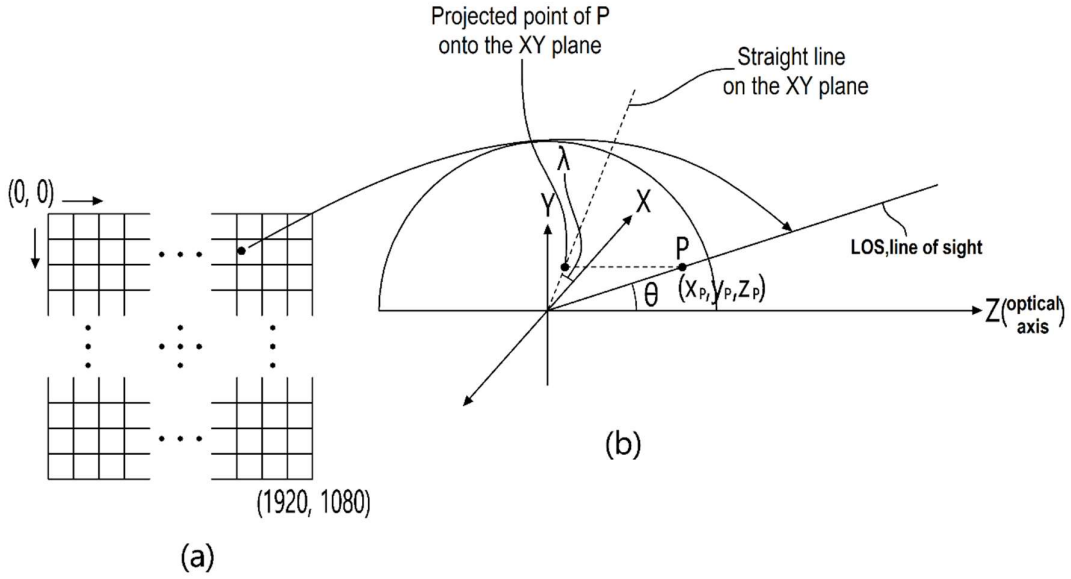

Projected point of P
onto the XY plane
Straight line
on the XY plane
λ
Y
X
P
LOS,line of sight
θ
(XP,YP,ZP)
Z(optical axis)
(b)
(0, 0)
(1920, 1080)
(a)

FIG.10.

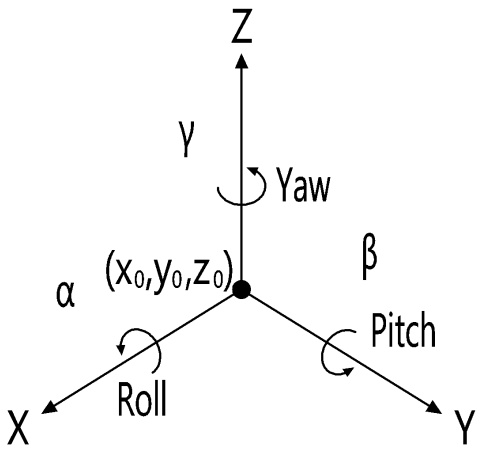

Z
γ
Yaw
β
(x0,y0,z0)
α
Pitch
Roll
X
Y

FIG.11.

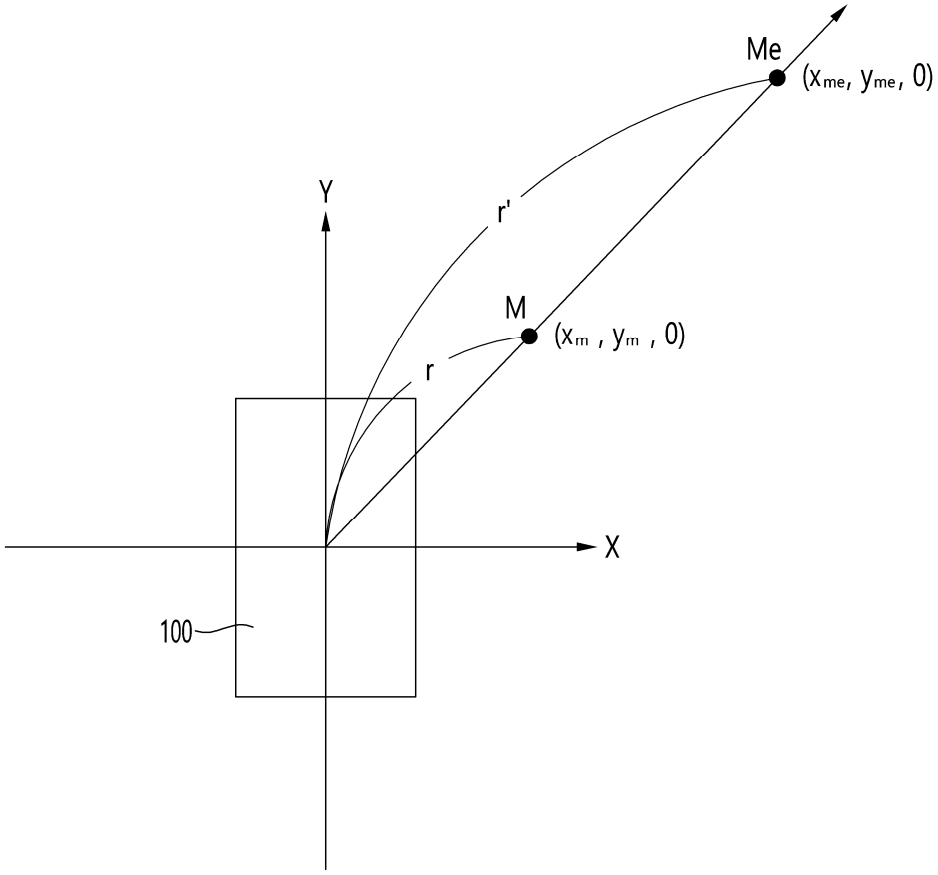


FIG.12.

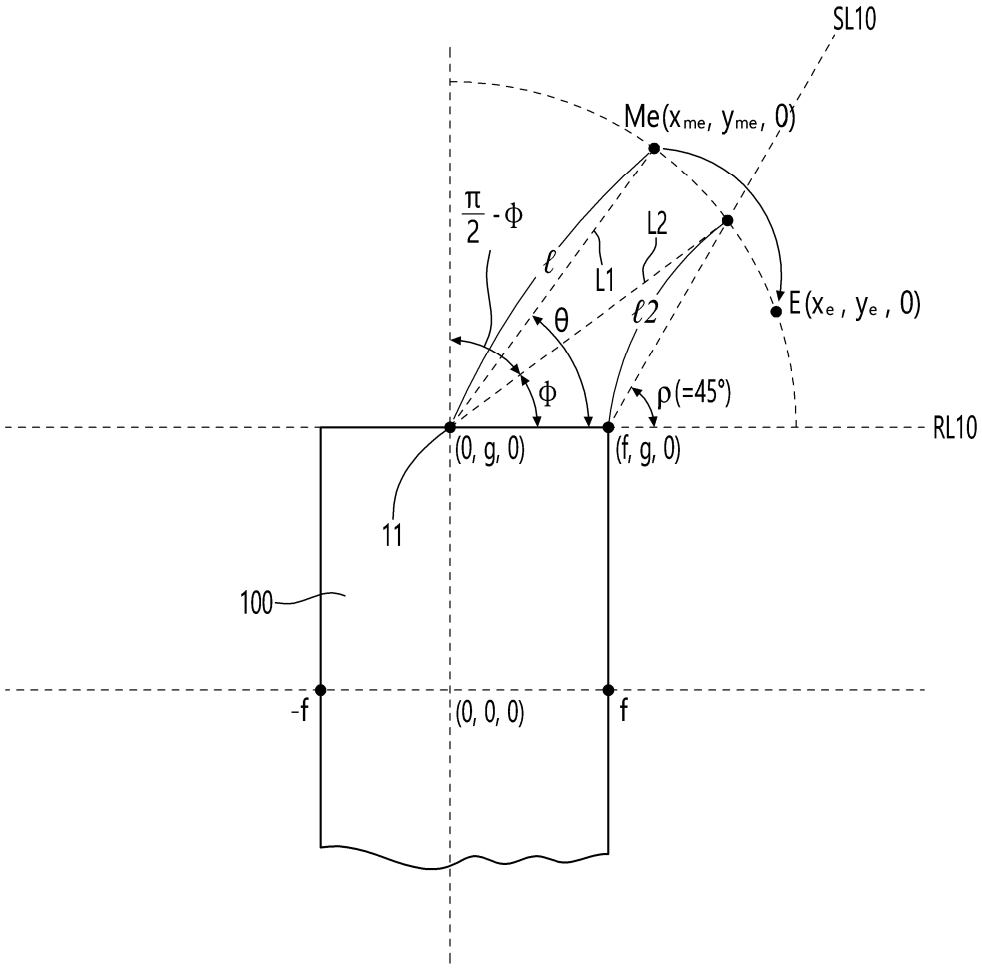
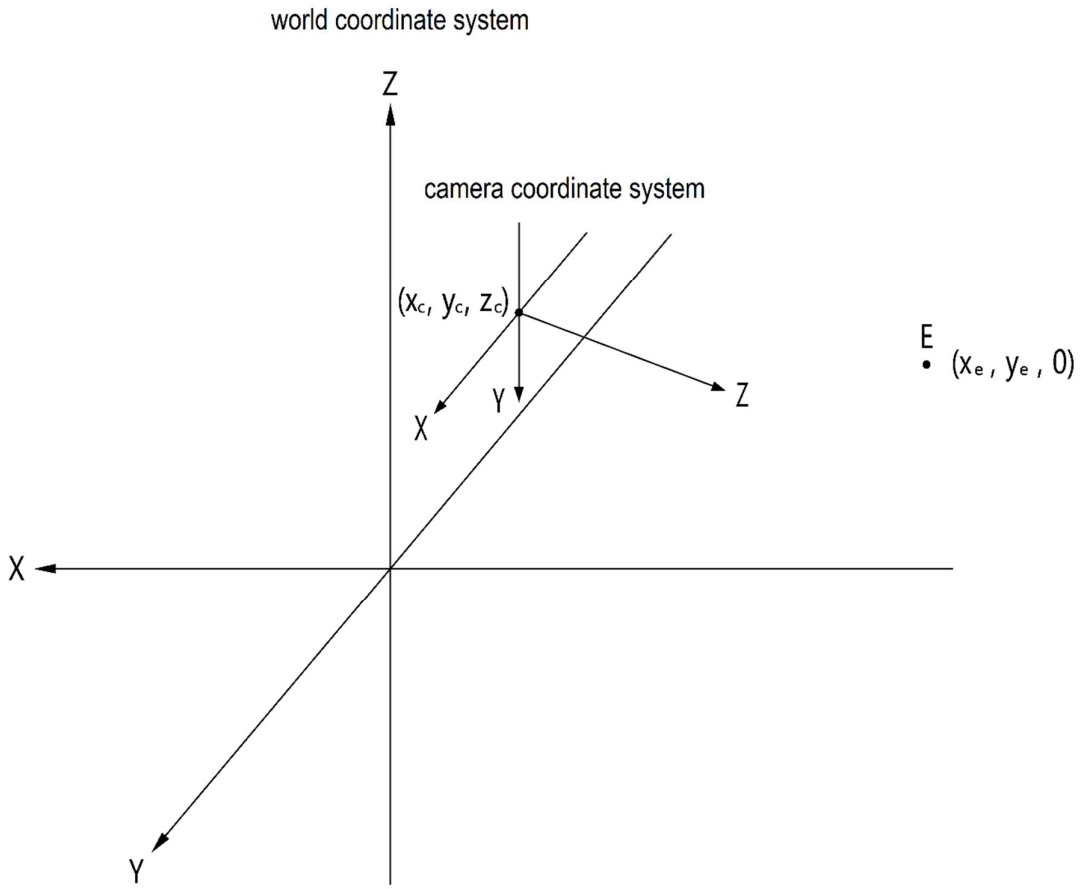

SL10
Me(xme, yme, 0)
π/2 - Φ
ℓ
L1
L2
ℓ2
θ
Φ
ρ(=45°)
E(xe , ye , 0)
RL10
(0, g, 0)
(f, g, 0)
11
100
-f
(0, 0, 0)
f

FIG.13.


world coordinate system
Z
camera coordinate system
(xc, yc, zc)
X
Y
Z
E
(xe , ye , 0)
X
Y