AROUND VIEW IMAGE GENERATION METHOD, APPARATUS, STORAGE MEDIUM, AND ELECTRONIC DEVICE

TECHNICAL FIELD

[0001] The present disclosure relates to image processing technologies, and in particular, to an around view image generation method, an apparatus, a storage medium, and an electronic device.

BACKGROUND

[0002] A vehicle around view monitor (AVM) is a commonly used functional monitor on an intelligent driving vehicle. The vehicle around view monitor uses images captured by cameras with multiple viewing angles installed on the vehicle to generate an around view image, so as to visually display a three-dimensional scene in which the vehicle is located.

[0003] In the related art, a preset three-dimensional (3D) model is usually adopted to render and obtain an around view image. However, because the preset 3D model is inconsistent with an actual scene, the visualization effect of the around view image is poor.

SUMMARY

[0004] The present disclosure provides an around view image generation method, an apparatus, a storage medium, and an electronic device, so as to improve the visualization effect of an around view image.

[0005] A first aspect of embodiments of the present disclosure provides an around view image generation method, including: determining environment images of multiple viewing angles; reusing voxel depth data of surroundings of a vehicle, output by an occupancy network model pre-configured in the vehicle according to the environment images; determining, based on the voxel depth data, depth maps respectively corresponding to the environment images of the multiple viewing angles; determining Gaussian point cloud data based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images; and generating an around view image based on the Gaussian point cloud data.

[0006] A second aspect of embodiments of the present disclosure provides an around view image generation apparatus, including: a determination module, configured to determine environment images of multiple viewing angles; a first processing module, configured to reuse voxel depth data of surroundings of a vehicle, output by an occupancy network model

pre-configured in the vehicle according to the environment images; a second processing module, configured to determine, based on the voxel depth data, depth maps respectively corresponding to the environment images of the multiple viewing angles; a third processing module, configured to determine Gaussian point cloud data based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images; and a fourth processing module, configured to generate an around view image based on the Gaussian point cloud data.

[0007] A third aspect of embodiments of the present disclosure provides a computer-readable storage medium storing a computer program thereon, wherein the computer program, when executed by a processor, is configured to implement the around view image generation method according to any one of the embodiments of the present disclosure.

[0008] A fourth aspect of embodiments of the present disclosure provides an electronic device, including: a memory, configured to store executable instructions for a processor; and a processor, configured to read the executable instructions from the memory and execute the executable instructions to implement the around view image generation method according to any one of the embodiments of the present disclosure.

[0009] A fifth aspect of embodiments of the present disclosure provides a computer program product, wherein instructions in the computer program product, when executed by a processor, perform the around view image generation method provided by any one of the embodiments of the present disclosure.

[0010] The around view image generation method, apparatus, storage medium, and electronic device provided by the above embodiments of the present disclosure, when generating an around view image, may determine environment images of multiple viewing angles; reuse voxel depth data of surroundings of the vehicle, output by an occupancy network model pre-configured in the vehicle according to the environment images; determine, based on the voxel depth data, depth maps respectively corresponding to the environment images of the multiple viewing angles; determine Gaussian point cloud data based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images; and generate the around view image based on the Gaussian point cloud data. It can be seen that the embodiments of the present disclosure reuse the voxel depth data output by the occupancy network model already existing in the vehicle to determine the depth maps of the environment images, thereby avoiding additionally occupying computing resources of the vehicle and effectively reducing computing requirements of the around view monitor. Moreover, the voxel depth data output by the occupancy network model has relatively high accuracy, and using the

voxel depth data to determine the depth maps of the environment images helps to improve the accuracy of the depth maps. Further, the Gaussian point cloud data is determined based on the environment images and the high-accuracy depth maps, and the around view image is generated by real-time rendering based on the Gaussian point cloud data, so that the around view image better conforms to an actual environment scene. By avoiding rendering the around view image using the preset 3D model, occurrences of distortion, seam misalignment, and other similar situations are avoided or reduced, thereby helping to improve the visualization effect of the around view image.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] FIG. 1 is an exemplary application scenario of an embodiment of the present disclosure.

[0012] FIG. 2 is a flowchart of an around view image generation method provided by some exemplary embodiments of the present disclosure.

[0013] FIG. 3 is a flowchart of an around view image generation method provided by other exemplary embodiments of the present disclosure.

[0014] FIG. 4 is a visualization diagram of voxel depth data and a depth map provided by some exemplary embodiments of the present disclosure.

[0015] FIG. 5 is a flowchart of an around view image generation method provided by still other exemplary embodiments of the present disclosure.

[0016] FIG. 6 is a flowchart of an around view image generation method provided by yet other exemplary embodiments of the present disclosure.

[0017] FIG. 7 is a schematic structural diagram of an around view image generation apparatus provided by some exemplary embodiments of the present disclosure.

[0018] FIG. 8 is a schematic structural diagram of an around view image generation apparatus provided by other exemplary embodiments of the present disclosure.

[0019] FIG. 9 is a schematic structural diagram of an around view image generation apparatus provided by still other exemplary embodiments of the present disclosure.

[0020] FIG. 10 is a schematic structural diagram of an around view image generation apparatus provided by yet other exemplary embodiments of the present disclosure.

[0021] FIG. 11 is a structural diagram of an electronic device provided by an embodiment of the present disclosure.

DETAILED DESCRIPTION

[0022] Exemplary embodiments of the present disclosure are described below in detail with reference to the accompanying drawings to illustrate the present disclosure. Apparently, the described embodiments are merely a part, rather than all, of the embodiments of the present disclosure. It should be understood that the present disclosure is not limited by the exemplary embodiments described herein.

[0023] Unless otherwise specifically stated, relative arrangements of components and steps, numerical expressions, and values described in these embodiments shall not limit the scope of the present disclosure.

Overview of the Application

[0024] In the related art for generating an around view image by a vehicle around view monitor (AVM), a preset three-dimensional (3D) model is usually adopted to render and obtain the around view image, so as to visually display a three-dimensional scene in which the vehicle is located. However, because the preset 3D model is a preset model, it is often inconsistent with an actual scene, resulting in a poor visualization effect of the around view image.

Exemplary Overview

[0025] FIG. 1 is an exemplary application scenario of an embodiment of the present disclosure. As shown in FIG. 1, when a vehicle (or ego-vehicle) 11 is traveling on a road, environment images of the surroundings of the vehicle may be collected by sensors disposed on the vehicle 11, and then an around view image of the surroundings of the vehicle 11 is generated using the around view image generation method of the present disclosure. The sensors on the vehicle 11 include, but are not limited to, cameras that collect the environment images, and may further include lidars, millimeter-wave radars, ultrasonic radars, etc. Optionally, the generated around view image may be displayed, for example, by any display device on the vehicle 11. The display device includes, for example but is not limited to, an instrument panel, a central control screen, a head-up display (HUD), an entertainment screen, etc.

[0026] The around view image generation method of the present disclosure may be executed by an in-vehicle computing platform or an in-vehicle terminal of the vehicle 11. For example, the in-vehicle computing platform or the in-vehicle terminal may include at least one processor and a memory. The memory is configured to store executable instructions for the processor. The processor is configured to read the executable instructions from the memory and execute the read executable instructions, so as to implement the around view image generation method of the present disclosure.

[0027] Specifically, the processor may determine environment images of multiple viewing angles; reuse voxel depth data of surroundings of the vehicle, output by an occupancy network

model pre-configured in the vehicle according to the environment images; determine, based on the voxel depth data, depth maps respectively corresponding to the environment images of the multiple viewing angles; determine Gaussian point cloud data based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images; and generate an around view image based on the Gaussian point cloud data.

[0028] It can be seen that the embodiments of the present disclosure reuse the voxel depth data output by the occupancy network model already existing in the vehicle to determine the depth maps of the environment images, thereby avoiding additionally occupying computing resources of the vehicle and effectively reducing computing requirements of the around view monitor. Moreover, the voxel depth data output by the occupancy network model has relatively high accuracy, and using the voxel depth data to determine the depth maps of the environment images helps to improve the accuracy of the depth maps. Further, the Gaussian point cloud data is determined based on the environment images and the high-accuracy depth maps, and the around view image is generated by real-time rendering based on the Gaussian point cloud data, so that the around view image better conforms to the actual environment scene. By avoiding rendering the around view image using the preset 3D model, occurrences of distortion, seam misalignment, and other similar situations are avoided or reduced, thereby helping to improve the visualization effect of the around view image.

Exemplary Method

[0029] FIG. 2 is a flowchart of an around view image generation method provided by some exemplary embodiments of the present disclosure. The around view image generation method provided by the embodiments of the present disclosure may be applied to an electronic device, an in-vehicle computing platform, or a system on chip (SoC) on an in-vehicle computing platform, or a microprogrammed control unit (MCU), or hardware and/or software applied to the electronic device or the system on chip. For example, the electronic device includes, but is not limited to, an in-vehicle terminal, an embodied agent, a robot, and the like, and the system on chip includes, but is not limited to, a system on chip on the in-vehicle terminal, the embodied agent, or the robot, etc. As shown in FIG. 2, the around view image generation method provided by the embodiments of the present disclosure may include the following steps 210 to 250.

[0030] At step 210, environment images of multiple viewing angles are determined.

[0031] The environment images of the multiple viewing angles include an environment image corresponding to each of the multiple viewing angles; that is, the environment images of the multiple viewing angles include multiple images. The vehicle may be provided with cameras (or still cameras) with the multiple viewing angles, and the environment images of the multiple

viewing angles may be collected by the cameras with the multiple viewing angles. The multiple viewing angles include, but are not limited to, for example, multiple viewing angles among a front view, a front-left view, a front-right view, a left view, a rear-left view, a rear view, a rear-right view, a right view, and the like. Cameras with different viewing angles may collect environment images of different viewing angle ranges around the vehicle.

[0032] Optionally, the environment images of the multiple viewing angles may be pre-processed images. For example, the environment images of the multiple viewing angles are obtained by pre-processing images collected by the cameras with the multiple viewing angles. The pre-processing includes, but is not limited to, for example, distortion correction, noise reduction filtering, size scaling, etc.

[0033] At step 220, voxel depth data of the surroundings of the vehicle, which is output by an occupancy network model pre-configured in the vehicle according to the environment images, is reused.

[0034] The occupancy network model may be referred to as an OCC model for short. The occupancy network model is a neural network model pre-configured on the vehicle. The occupancy network model is configured to generate voxel depth data according to input sensor data, and the voxel depth data may be referred to as voxel occupancy data.

[0035] The sensor data may include, but is not limited to, the environment images collected by the cameras. For example, the sensor data may further include one or more of point cloud data collected by a lidar, distance data collected by a distance sensor, etc. That is, the input data of the occupancy network model may be single-modal sensor data or multi-modal sensor data.

[0036] The voxel depth data may include one or more of spatial position information of voxels around the vehicle, an occupancy state of voxels, an occupancy probability of voxels, and semantic information of voxels, and the like. The voxels (or voxel grids) refer to three-dimensional grids obtained by dividing a three-dimensional space around the vehicle. The size of a voxel may be set as required; the size of the voxel is, for example, but not limited to, 20 cm × 20 cm × 20 cm, that is, the length, width, and height of the voxel are all 20 cm. In practical applications, the spatial shape of a voxel is not limited to a cube with equal length, width, and height, and may also be a cuboid, a tetrahedron, a sphere, etc.

[0037] The spatial position information of a voxel is a three-dimensional coordinate of the voxel in a three-dimensional coordinate system. Optionally, the three-dimensional coordinate system may be a vehicle coordinate system (VCS) or another three-dimensional coordinate system. The other three-dimensional coordinate system includes, for example, but is not limited to, a world coordinate system, a VCS coordinate system of the vehicle at an initial position, etc.

Various coordinate systems may be converted to each other. For example, the vehicle coordinate system may be represented as an O-XYZ coordinate system, where O is an origin of the coordinate system, for example, O is a center of a rear axle of the vehicle, X represents an X-axis in a forward direction of a vehicle head, Y represents a Y-axis pointing from the coordinate origin to a left side of the vehicle, and Z represents a Z-axis in a vertically upward direction. In practical applications, the vehicle coordinate system is not limited to this example coordinate system.

[0038] The occupancy state represents whether a voxel is occupied by an object in the three-dimensional space. The occupancy state may include at least two states: occupied and free. Occupied means that an object exists in the voxel space. Free means that the voxel space is not occupied, that is, no object exists. The occupancy state of any voxel may be occupied or free, etc.

[0039] The occupancy probability represents a confidence of a voxel being occupied by an object. The greater the occupancy probability is, the higher the credibility that an object exists in the voxel space.

[0040] The semantic information represents a classification of an object occupying a voxel. The semantic information includes, for example, but is not limited to, a motor vehicle, a non-motor vehicle, a pedestrian, a guardrail, a road edge, a green belt, a road surface, etc.

[0041] In some embodiments, the voxel depth data of the surroundings of the vehicle may be generated using the occupancy network model according to only the environment images of the multiple viewing angles. Alternatively, the voxel depth data of the surroundings of the vehicle may be generated using the occupancy network model according to the environment images of the multiple viewing angles and environment point clouds collected by a lidar. The specific input data on which the occupancy network model depends is determined by a configuration of the existing occupancy network model on the vehicle, which is not limited by the embodiments of the present disclosure. In the embodiments of the present disclosure, the voxel depth data, output by the occupancy network model of the vehicle, is directly reused. For example, the existing occupancy network model on the vehicle is used for a trajectory planning function or other functions, and the present disclosure directly reuses the voxel depth data output by the occupancy network model without affecting the application of the occupancy network model in the other functions. For example, the voxel depth data output by the occupancy network model may be requested from a functional module performing inference of the occupancy network model, or according to a protocol, the functional module performing the inference of the occupancy network model may actively transmit the voxel depth data to a functional module or apparatus executing the method of the present disclosure after obtaining the voxel depth data. A

specific manner of obtaining the voxel depth data is not limited.

[0042] In the embodiments of the present disclosure, reusing the voxel depth data output by the occupancy network model pre-configured in the vehicle can be implemented by using a characteristic that all data of the vehicle is capable of being acquired by a central processor, a cockpit processor, a cockpit-driving integrated processor, etc., on the vehicle.

[0043] In some embodiments, the method of the present disclosure may be executed by at least one of processors such as a central processing unit, a cockpit processing unit, a cockpit-driving integrated processing unit on the vehicle; during execution of the method, the reuse of the voxel depth data is implemented using the characteristic that the processor is capable of acquiring all data of the vehicle. All the data of the vehicle that may be acquired by the processor includes, but is not limited to, intelligent driving data. The intelligent driving data includes various data related to an intelligent driving function of the vehicle, such as sensor data, data obtained by processing the sensor data based on various intelligent driving algorithms, etc. The data obtained based on the intelligent driving algorithms includes, for example, but is not limited to, the voxel depth data obtained based on the occupancy network model, a target detection result obtained based on a target detection model, and a trajectory prediction result obtained based on a trajectory prediction model.

[0044] In other embodiments, the method of the present disclosure may be performed by another electronic device or another processor other than the central processor, the cockpit processor, and the cockpit-driving integrated processor. While performing the method, the other electronic device or the other processor may acquire the voxel depth data from the central processor, the cockpit processor, or the cockpit-driving integrated processor, etc., to implement the reuse of the voxel depth data. In this way, the other electronic device or the other processor does not need to regenerate the voxel depth data.

[0045] At step 230, depth maps respectively corresponding to the environment images of the multiple viewing angles are determined based on the voxel depth data.

[0046] A depth map represents an image formed by depth values respectively corresponding to pixels of an environment image; that is, a depth map corresponding to an environment image of any one of the multiple viewing angles includes depth values respectively corresponding to pixels of the environment image of the viewing angle. For example, the environment image is an RGB three-channel image of H*M, and the depth map may be a single-channel image of H*M. RGB represents a red channel, a green channel, and a blue channel. Optionally, the depth map may be represented as an RGBD four-channel image of H*M, where D represents a depth channel, H represents a height of the image, and M represents a width of the image. This is

merely an example of the environment image and the depth map shown here, which is not intended to limit the present disclosure; and in practical applications, the environment image is not limited to an RGB three-channel image.

[0047] Optionally, taking a three-dimensional coordinate system of the voxel depth data as the vehicle coordinate system as an example, for an environment image of any one of the multiple viewing angles, depth values respectively corresponding to pixels in the environment image may be determined based on the voxel depth data, the environment image of the viewing angle, and a conversion relationship between the vehicle coordinate system and an image pixel coordinate system of the viewing angle, and the depth map corresponding to the environment image is obtained based on the depth values respectively corresponding to the pixels.

[0048] At step 240, Gaussian point cloud data is determined based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images.

[0049] The Gaussian point cloud data is explicitly structured data that adopts multiple Gaussian ellipsoids to characterize a three-dimensional scene. The Gaussian point cloud data may also be referred to as a 3D Gaussian Splatting (3DGS) point cloud. Each point in the Gaussian point cloud data (or a Gaussian primitive, or a Gaussian point) is a 3D Gaussian ellipsoid with a spatial volume, rather than a discrete coordinate point (x, y, z). The Gaussian point cloud data may include parameters of Gaussian ellipsoids respectively corresponding to multiple Gaussian primitives, and the parameters of a Gaussian ellipsoid include, for example, but are not limited to, a position, a shape, opacity, and appearance of the Gaussian ellipsoid.

[0050] The position of the Gaussian ellipsoid may be three-dimensional center coordinates of the Gaussian ellipsoid. The three-dimensional center coordinates represent three-dimensional coordinates of a center point of the Gaussian ellipsoid in the vehicle coordinate system.

[0051] The shape of the Gaussian ellipsoid represents an orientation and scaling of the Gaussian ellipsoid. Optionally, the shape of the Gaussian ellipsoid may include scale parameters and rotation parameters. The scale parameters are used for controlling stretching magnitudes of the Gaussian ellipsoid along XYZ axes of the vehicle coordinate system, thereby determining a spatial range covered by the Gaussian ellipsoid. The rotation parameters represent a spatial orientation of the Gaussian ellipsoid. Optionally, the shape of the Gaussian ellipsoid may be represented as a covariance matrix $\Sigma \in R^{3\times3}$.

[0052] The opacity represents a degree of light transmission of the Gaussian ellipsoid.

[0053] The appearance of the Gaussian ellipsoid represents a color and spherical harmonic coefficients of the Gaussian ellipsoid. The spherical harmonic coefficients represent processing

of view-dependent lighting effects.

[0054] Optionally, the Gaussian point cloud data may further include a semantic category of the Gaussian ellipsoid. The semantic category represents a type of object within the Gaussian ellipsoid.

[0055] Optionally, based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images, the Gaussian point cloud data may be generated in any implementable manner of generating a Gaussian point cloud. The manner of generating the Gaussian point cloud includes, for example, but is not limited to, a Gaussian splatting algorithm series.

[0056] Optionally, for characteristics of an around view task, different accuracies/densities of Gaussian ellipsoids may be set based on distances from the vehicle. For example, Gaussian ellipsoids within a 10-meter range are prioritized for learning, so that more Gaussian ellipsoids can be adopted for more accurate expression within the 10-meter range, while fewer Gaussian ellipsoids are adopted for expression at farther distances, thereby reducing computational load.

[0057] At step 250, an around view image is generated based on the Gaussian point cloud data.

[0058] The Gaussian point cloud data uses a large number of Gaussian ellipsoids to represent a three-dimensional scene of the surroundings of the vehicle. When the around view image is generated, the three-dimensional scene may be rendered into the around view image based on the Gaussian point cloud data.

[0059] The around view image generation method provided by the embodiment, when generating the around view image, may determine environment images of multiple viewing angles; reuse voxel depth data of surroundings of the vehicle, output by an occupancy network model pre-configured in the vehicle according to the environment images; determine, based on the voxel depth data, depth maps respectively corresponding to the environment images of the multiple viewing angles; determine Gaussian point cloud data based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images; and generate the around view image based on the Gaussian point cloud data. It can be seen that the embodiments of the present disclosure reuse the voxel depth data output by the occupancy network model already existing in the vehicle to determine the depth maps of the environment images, and there is no need to deploy the occupancy network model in the around view monitor, thereby avoiding additionally occupying computing resources of the vehicle and effectively reducing computing requirements of the around view monitor. Moreover, the voxel depth data output by the occupancy network model has relatively high accuracy, and using the voxel depth data to determine the depth maps of the environment images helps to

improve the accuracy of the depth maps. Further, the Gaussian point cloud data is determined based on the environment images and the high-accuracy depth maps, and the around view image is generated by real-time rendering based on the Gaussian point cloud data, so that the around view image better conforms to an actual environment scene. By avoiding rendering the around view image using the preset 3D model, occurrences of distortion, seam misalignment, and other similar situations are avoided or reduced, thereby helping to improve the visualization effect of the around view image.

[0060] FIG. 3 is a flowchart of an around view image generation method provided by other exemplary embodiments of the present disclosure.

[0061] In some alternative embodiments, based on the embodiment shown in FIG. 2 above, the voxel depth data includes depth values of voxel grids occupied by an object around the vehicle.

[0062] The object refers to a person or a physical object around the vehicle; the object includes, for example, but is not limited to, a pedestrian, a rider, another vehicle, an animal, a fence, a road edge, a sign, etc., around the vehicle.

[0063] As shown in FIG. 3, the step 230 of determining, based on the voxel depth data, the depth maps respectively corresponding to the environment images of the multiple viewing angles includes the following steps 2310 to 2340.

[0064] At step 2310, for any one of the environment images of the multiple viewing angles, a ray from an optical center of a camera to a target pixel in the environment image is determined based on camera parameters corresponding to the environment image.

[0065] The camera parameters may include extrinsic parameters and intrinsic parameters of the camera. The camera parameters corresponding to the environment image refer to parameters of the camera that captures the environment image. The environment images of different viewing angles have corresponding camera parameters, and the camera parameters are pre-calibrated parameters. The target pixel may be any pixel in the environment image. The ray from the optical center of the camera to the target pixel in the environment image may be a ray in the vehicle coordinate system.

[0066] Optionally, each pixel in the environment image may be used as the target pixel, or a part of pixels in the environment image may be used as target pixels. For example, a part of pixels are sampled from all pixels of the environment image as the target pixels according to a preset sampling manner.

[0067] In some alternative embodiments, for a target pixel in any environment image, a ray from the optical center of the camera to the target pixel in a camera coordinate system may be determined based on pixel coordinates of the target pixel and the intrinsic parameters in the

camera parameters corresponding to the environment image; and further, the ray may be converted from the camera coordinate system to the vehicle coordinate system based on the extrinsic parameters in the camera parameters corresponding to the environment image, to obtain a ray from the optical center of the camera to the target pixel in the environment image in the vehicle coordinate system.

[0068] At step 2320, an occupied target voxel grid in the voxel depth data that intersects with the ray corresponding to the target pixel is determined based on the voxel depth data.

[0069] The ray corresponding to the target pixel refers to a ray from the optical center of the camera to the target pixel. When the ray from the optical center of the camera to the target pixel extends in a three-dimensional space, if the ray enters or passes through the occupied target voxel grid in the voxel depth data, it indicates that the target voxel grid intersects with the ray corresponding to the target pixel. That is, an occupancy state of the target voxel grid is occupied, and the target voxel grid intersects with the ray corresponding to the target pixel.

[0070] Optionally, if the ray corresponding to the target pixel intersects with multiple occupied voxel grids in the voxel depth data, a voxel grid closest to the optical center of the camera among the multiple occupied voxel grids is used as the target voxel grid.

[0071] At step 2330, a depth value of the target voxel grid is used as a depth value of the target pixel.

[0072] In some embodiments, the depth value of the target voxel grid may be determined based on spatial position information of the target voxel grid. For example, a coordinate value of the target voxel grid in a Z_1 direction of a camera coordinate system $\mathrm{O}_1 - \mathrm{X}_1\mathrm{Y}_1\mathrm{Z}_1$ may be determined based on the spatial position information of the target voxel grid and the extrinsic parameters of the camera that captures the environment image where the target pixel is located, where O_1 represents an origin of the camera coordinate system, and X_1, Y_1, and Z_1 represent a horizontal axis, a vertical axis, and an optical axis of the camera coordinate system, respectively; the coordinate value in the Z_1 direction is used as the depth value of the target voxel grid, and then the depth value of the target voxel grid is used as the depth value of the target pixel.

[0073] In some alternative embodiments, the depth value of the target voxel grid may be a depth value of an intersection point between the ray and the target voxel grid, that is, a Z-direction coordinate of the intersection point in the camera coordinate system.

[0074] In some alternative embodiments, the pixel coordinates of the target pixel may be converted into three-dimensional coordinates (X_c, Y_c, Z_c) in the camera coordinate system through the following formula (1):

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = Z_c \cdot K^{-1} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \quad \text{Formula (1)}$$

[0075] where $\begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix}$ represents a vector representation of the three-dimensional coordinates (X_c, Y_c, Z_c) in the camera coordinate system; $K^{-1} = \begin{bmatrix} 1/f_x & 0 & -c_x/f_x \\ 0 & 1/f_y & -c_y/f_y \\ 0 & 0 & 1 \end{bmatrix}$ represents an inverse matrix of a camera intrinsic matrix K; u and v represent pixel coordinates of the target pixel; $\begin{pmatrix} u \\ v \\ 1 \end{pmatrix}$ represents homogeneous pixel coordinates; f_x, and f_y represent pixel focal lengths in an x direction and a y direction, respectively; and c_x and c_y represent an image principal point. In this case, (X_c, Y_c, Z_c) is a ray expression, and $K^{-1}\begin{pmatrix} u \\ v \\ 1 \end{pmatrix}$ is a spatial ray direction starting from the optical center of the camera; by extending along the ray direction, the target voxel grid in the voxel depth data that intersects with the ray may be determined, and a Z_c value of the intersection point between the ray and the target voxel grid is used as the depth value of the target pixel.

[0076] At step 2340, a depth map corresponding to the environment image is determined based on depth values of target pixels.

[0077] In a case where each pixel in the environment image is used as the target pixel, a depth value of each pixel of the environment image may be obtained through steps 2310 to 2330. Then, when the depth map corresponding to the environment image is determined based on the depth values of the target pixels, the depth map corresponding to the environment image may be obtained based on the depth values of the respective pixels of the environment image.

[0078] In a case where a part of pixels in the environment image are respectively used as the target pixels, obtained depth values of the target pixels include depth values of these pixels. In this case, when the depth map corresponding to the environment image is determined based on the depth values of the target pixels, the sparse depth values of these pixels may first be densified to obtain depth values of respective pixels in the environment image, and then the depth map of the environment image is obtained based on the depth values of the respective pixels. Examples of the densification manner include, but are not limited to, densification by interpolation, densification by deep learning, and the like.

[0079] In some examples, FIG. 4 is a schematic visualization diagram of voxel depth data and a depth map provided by some exemplary embodiments of the present disclosure. As shown in

FIG. 4, for the same object in a three-dimensional space (e.g., a rabbit in FIG. 4), visualization of the voxel depth data of the object has a three-dimensional stereoscopic effect, while the depth map is a two-dimensional image without a three-dimensional stereoscopic effect.

[0080] In the embodiments of the present disclosure, by determining the ray from the optical center of the camera to the target pixel in the environment image based on the camera parameters corresponding to the environment image, determining the target voxel grid according to an intersection situation between the ray and the occupied voxel grids in the voxel depth data, and using the depth value of the target voxel grid as the depth value of the target pixel, determining the depth values on a pixel-by-pixel basis is achieved, so that dense depth values of pixels can be effectively obtained. Because pixels are dense and voxels are sparse, if the voxels are projected onto the environment image, only sparse depth values of pixels can be obtained, and the depth value of each pixel cannot be obtained. In the embodiments of the present disclosure, by emitting the ray from the optical center of the camera to the pixel to determine the depth value of the pixel, the depth value of each pixel can be obtained.

[0081] FIG. 5 is a flowchart of an around view image generation method provided by still other exemplary embodiments of the present disclosure.

[0082] In some alternative embodiments, as shown in FIG. 5, before the step 2310 of determining the ray from the optical center of the camera to the target pixel in the environment image based on the camera parameters corresponding to the environment image, the method further includes the following steps 310 to 330:

[0083] At step 310, an object segmentation result corresponding to the environment image is determined.

[0084] The object segmentation result may include one or more segmentation regions of objects, semantic types, and the like.

[0085] Optionally, a pre-trained semantic segmentation model may be used to perform semantic segmentation on the environment image, to obtain the object segmentation result corresponding to the environment image. The semantic segmentation model may be any implementable neural network model, and the present disclosure does not limit the implementation manner of obtaining the object segmentation result.

[0086] At step 320, a sampling density corresponding to each of the segmentation regions is determined based on the object segmentation result.

[0087] A sampling density is a parameter used for controlling a degree of sparsity or density of sampling pixels from a segmentation region. Optionally, the sampling density may be represented as a number of pixels sampled per unit pixel area in the segmentation region.

Exemplarily, sampling density = the number of sampled pixels / the total number of pixels in the segmentation region. For the same segmentation region, the larger the sampling density is, the more and denser the sampled pixels are; the smaller the sampling density is, the fewer and sparser the sampled pixels are.

[0088] In an example, a segmentation region of an object A includes 1000 pixels, and the sampling density is 1/5; one pixel is sampled from every 5 pixels in the segmentation region of the object A, and 200 pixels are sampled in total.

[0089] In some embodiments, the sampling density corresponding to the segmentation region of the object may be determined according to one or more of the following: a semantic type of the object, a distance between the object and the vehicle, and other factors. For example, a smaller sampling density may be used for a segmentation region of a relatively stable road surface, and a larger sampling density may be used for a segmentation region of a nearby pedestrian.

[0090] At step 330, the target pixels are determined from the environment image based on the sampling density corresponding to each of the segmentation regions.

[0091] After the sampling density corresponding to each of the segmentation regions is obtained, pixel sampling may be performed in each segmentation region based on the sampling density corresponding to each of the segmentation regions to obtain sampled pixels corresponding to each segmentation region, and the sampled pixels corresponding to each segmentation region are determined as the target pixels.

[0092] In the embodiments of the present disclosure, by determining the object segmentation result corresponding to the environment image, determining the sampling density corresponding to each of the segmentation regions based on the object segmentation result, and then sampling and obtaining the small number of target pixels from the environment image based on the sampling density corresponding to each of the segmentation regions, on one hand, a computational amount of calculating the depth values of the target pixels according to rays can be reduced; on the other hand, a sparsity/density degree of the target pixels sampled from the segmentation regions of different objects can be controlled, thereby improving accuracy and effectiveness of the target pixels, so that the depth values of the remaining pixels can be quickly obtained by interpolation or the like after the depth values corresponding to the target pixels are calculated, to ensure accuracy and effectiveness of the depth values of the depth map as a whole.

[0093] In some alternative embodiments, based on any one of the above embodiments, the step 2340 of determining the depth map corresponding to the environment image based on the depth values of the target pixels includes: for any segmentation region, determining depth values of

other pixels in the segmentation region based on the depth values of the target pixels in the segmentation region, where the other pixels are pixels in the segmentation region other than the target pixels; and determining the depth map corresponding to the environment image based on the depth values of the target pixels and the depth values of the other pixels in the segmentation regions.

[0094] For any segmentation region, the target pixel includes a part of pixels sampled in the segmentation region. After the depth values of the target pixels of the segmentation region are obtained, the depth values of the other pixels in the segmentation region may be determined by densification based on the depth values of the target pixels. For example, a segmentation region 1 has 1000 pixels in total, 200 pixels are sampled as the target pixels, and the depth values of the 200 target pixels are obtained according to an intersection situation between the rays from the optical center of the camera to each of the target pixels and the occupied voxels in the depth voxel data; and then, the depth values of the 800 remaining pixels are obtained by densification based on the depth values of the 200 target pixels. Examples of the densification manner include, but are not limited to, densification by interpolation, densification by deep learning, and the like.

[0095] Densification by interpolation refers to performing interpolation calculation based on the obtained depth values of the target pixels, to obtain the depth values of other pixels between the target pixels. Examples of the interpolation manner include, but are not limited to, bilinear interpolation, inverse distance weighted interpolation, and the like.

[0096] In some embodiments, taking bilinear interpolation as an example, for any pixel with an unknown depth (which may be referred to as a first pixel), multiple neighboring target pixels with known depth values are selected from surroundings of the first pixel; and further, a depth value of the first pixel is obtained through horizontal linear interpolation and vertical linear interpolation based on the depth values of the respective target pixels.

[0097] In other embodiments, taking inverse distance weighted interpolation as an example, for any first pixel with an unknown depth, multiple neighboring target pixels with known depth values may be selected from surroundings of the first pixel; distance weights of the respective target pixels are determined according to distances between the respective target pixels and the first pixel, where the distance weights decrease as the distances increase; and further, thae depth value of the first pixel is obtained by weighted summation of the respective target pixels based on the distance weights of the respective target pixels.

[0098] Densification by deep learning refers to performing inference based on the depth values of the target pixels and the environment image by using a pre-trained depth densification network model (or a depth completion network model), to obtain the depth values of other pixels,

or directly outputting a depth map by using a depth densification network model. The depth densification network model adopts, for example, but is not limited to, a model based on a U-shaped network (U-Net), a model based on GuideNet (guided convolution network), and the like.

[0099] Optionally, a sparse depth map may be generated based on depth values of target pixels, and the depth map corresponding to the environment image may be generated based on the environment image and the sparse depth map by using the depth densification network model. Optionally, a validity mask map may also be generated based on the sparse depth map; and the validity mask map includes masks of whether respective pixels in the sparse depth map have known depth values. For any pixel, mask = 1 indicates that the pixel has a known depth value; and mask = 0 indicates that the pixel does not have a known depth value, that is, the pixel is a pixel to be completed. The depth map of the environment image is generated based on the environment image, the sparse depth map, and the validity mask map by using the depth densification network model.

[00100] In the embodiments of the present disclosure, in a case where a small number of pixels are sampled as the target pixels according to the sampling density of the segmentation region, after the depth values of the target pixels are determined according to the intersection situation of the rays from the optical center of the camera to the respective target pixels with the voxel depth data, depth values of other pixels are quickly obtained by performing densification based on the depth values of the target pixels, which helps to improve depth calculation efficiency.

[00101] In some alternative embodiments, based on any one of the above embodiments, the step 240 of determining Gaussian point cloud data based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images in may include: determining the Gaussian point cloud data by using a pre-configured Gaussian splatting algorithm based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images.

[00102] The Gaussian splatting algorithm refers to a 3D Gaussian Splatting (3DGS) algorithm, which is a three-dimensional scene representation and rendering algorithm that uses three-dimensional anisotropic Gaussian ellipsoids as scene primitives and adopts differentiable tile-based rasterization rendering to achieve inverse optimization. The algorithm is capable of initializing Gaussian primitives according to the environment images of multiple viewing angles and the depth maps of the respective environment images, and further iteratively optimizing parameters of each Gaussian primitive to obtain Gaussian point cloud data conforming to an actual three-dimensional scene.

[00103] In the embodiments of the present disclosure, the Gaussian point cloud data is determined by using the Gaussian splatting algorithm based on the environment images of the multiple viewing angles and the depth maps of the respective environment images, thereby realizing fitting of geometry and textures of a real scene by using a massive number of three-dimensional Gaussian ellipsoids, facilitating real-time rendering of images from arbitrary viewing angles, and providing an effective three-dimensional scene representation for generation of around view images of the around view system.

[00104] FIG. 6 is a flowchart of an around view image generation method provided by yet other exemplary embodiments of the present disclosure.

[00105] In some alternative embodiments, based on any one of the above embodiments, as shown in FIG. 6, the step 240 of determining Gaussian point cloud data based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images may include the following steps 2410 to 2420:

[00106] At step 2410, an initial Gaussian point cloud is determined based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images.

[00107] The initial Gaussian point cloud includes parameters of multiple initialized Gaussian ellipsoids. The parameters of a Gaussian ellipsoid include, for example, but are not limited to, a position, a shape, opacity, and spherical harmonic coefficients of the Gaussian ellipsoid.

[00108] In some embodiments, a spatial point of each pixel in each environment image in the three-dimensional coordinate system may be determined based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images; the initial Gaussian ellipsoids corresponding to the respective spatial points are determined based on the respective spatial points. For example, for each spatial point, initial values of parameters such as a position, a shape, opacity, and appearance parameters of a three-dimensional Gaussian ellipsoid are initialized to obtain the initial Gaussian ellipsoid corresponding to the spatial point. For each environment image, an initial Gaussian ellipsoid subset corresponding to the environment image may be obtained. An initial Gaussian ellipsoid set is determined based on the initial Gaussian ellipsoid subsets of the respective environment images, and the initial Gaussian ellipsoid set is determined as the initial Gaussian point cloud. Optionally, the initial Gaussian ellipsoid subsets of the respective environment images may be merged to obtain the initial Gaussian ellipsoid set.

[00109] In some alternative embodiments, valid pixels may be filtered based on the environment images of the multiple viewing angles and the depth maps respectively

corresponding to the environment images, then spatial points corresponding to the valid pixels are determined, and the initial Gaussian point cloud is determined based on the spatial points corresponding to the valid pixels. Filtering the valid pixels includes, for example, removing pixels with invalid depth. The invalid depth is, for example, a depth not within a preset depth range, and the preset depth range may be set according to actual requirements. Exemplarily, pixels with a depth value = 0 or with a depth value greater than a preset depth threshold may be removed, and pixels with a valid depth are used as the valid pixels.

[00110] At step 2420, the initial Gaussian point cloud is iteratively adjusted based on the environment images of the multiple viewing angles, to determine the Gaussian point cloud data.

[00111] The Gaussian point cloud to be adjusted may be iteratively adjusted based on an image reconstruction error, to continuously optimize the Gaussian point cloud and obtain the Gaussian point cloud data. The Gaussian point cloud to be adjusted may be the initial Gaussian point cloud or a Gaussian point cloud adjusted at any iteration. The image reconstruction error is an error between a reconstructed image of each viewing angle reconstructed based on the Gaussian point cloud to be adjusted, and the environment image of said each viewing angle.

[00112] In some embodiments, during a first iterative adjustment, the initial Gaussian point cloud may be used as the Gaussian point cloud to be adjusted; a reconstructed image of each viewing angle is generated by rendering based on the Gaussian point cloud to be adjusted; the image reconstruction error is determined based on the reconstructed image of each viewing angle and the environment image of each viewing angle; based on the image reconstruction error and with an objective of minimizing the image reconstruction error, the Gaussian point cloud to be adjusted is adjusted by using a preset optimization algorithm, to obtain an adjusted Gaussian point cloud; the adjusted Gaussian point cloud is then determined as the Gaussian point cloud to be adjusted, and a next iteration is entered until an iteration termination condition is satisfied, to obtain a converged Gaussian point cloud which is determined as the Gaussian point cloud data. The preset optimization algorithm includes, for example, but is not limited to, one or more of single Gaussian parameter gradient optimization, adaptive density control optimization, opacity periodic reset optimization, loss and regularization optimization, and the like.

[00113] During any iteration, adjustment manners for the Gaussian point cloud to be adjusted include, for example, but are not limited to, a position (pos) correction of the Gaussian ellipsoid, three-axis scaling (scale) of a shape, adjustment of an orientation (rot), adjustment of an opacity, optimization of spherical harmonic coefficients, Gaussian splitting (Split), Gaussian cloning (Clone), Gaussian pruning (Prune), and the like.

[00114] The position correction refers to correcting the position of a Gaussian ellipsoid in

three-dimensional space, for example, correcting a Gaussian ellipsoid floating in the air or having an inaccurate position to a surface of an object.

[00115] The three-axis scaling of the shape refers to controlling the size of the Gaussian ellipsoid; if the Gaussian ellipsoid is rather large, the Gaussian ellipsoid is reduced, and if the Gaussian ellipsoid is rather small, the Gaussian ellipsoid is enlarged, so that the Gaussian ellipsoid can fit an outline of the object.

[00116] The adjustment of the orientation refers to changing the placement angle of the Gaussian ellipsoid, so that the Gaussian ellipsoid fits a curved surface trend of the object.

[00117] The adjustment of the opacity refers to adjusting a degree of light transmission of the Gaussian ellipsoid, for example, making a redundant Gaussian ellipsoid gradually transparent.

[00118] The optimization of the spherical harmonic coefficients refers to optimizing colors of the object at different viewing angles, to address reflections, color differences, and the like.

[00119] By finely adjusting positions, shapes, orientations, opacities, colors, etc., of the respective Gaussian ellipsoids, the object fitted by the respective Gaussian ellipsoids is made to better conform to an actual object.

[00120] The Gaussian splitting (Split) refers to refining larger Gaussian ellipsoids and increasing the number of Gaussian ellipsoids. For example, if the pixel size of a Gaussian ellipsoid after the Gaussian ellipsoid is projected onto an environment image is too large or an image gradient is too large, the Gaussian ellipsoid is split into two or more smaller Gaussian ellipsoids to better restore edge and corner details of the object. The splitting manner includes, for example, but is not limited to, splitting along a longest axis of the Gaussian ellipsoid, and reducing an overall size by scaling.

[00121] Gaussian cloning (Clone) refers to copying a new Gaussian ellipsoid in place with a slightly positional offset and inheriting parameters of a parent Gaussian ellipsoid. An objective of Gaussian cloning is to add new Gaussian ellipsoids to detail regions and increase the number of Gaussian ellipsoids. For example, if the image gradient is large while the size of the Gaussian ellipsoid itself is very small, it indicates that details in this region are insufficient and the image is blurry; through Gaussian cloning, Gaussian ellipsoids are supplemented for fine textures (such as lane markings, branches, etc.), to improve image fineness.

[00122] Gaussian pruning (Prune) refers to deleting useless Gaussian ellipsoids to reduce the number of Gaussian ellipsoids. Optionally, whether a Gaussian ellipsoid needs to be deleted may be determined according to the opacity of the Gaussian ellipsoid, the size of the Gaussian ellipsoid projected onto the environment image, etc. In one example, if the opacity α is less than a minimal threshold, it indicates that the region of the Gaussian ellipsoid is nearly transparent

and makes no contribution to imaging, and the Gaussian ellipsoid may be deleted. The minimal threshold is, for example, but not limited to, 0.005. In another example, if the size of the Gaussian ellipsoid projected onto the environment image is abnormally large, it indicates that the Gaussian ellipsoid is a floating giant pseudo-Gaussian ellipsoid, and the Gaussian ellipsoid may be deleted. In yet another example, if the Gaussian ellipsoid is an outlier Gaussian ellipsoid far away from the scene and floating outside space, the Gaussian ellipsoid may be deleted.

[00123] By Gaussian splitting, Gaussian cloning, and the like, the detail regions are split, or the new Gaussian ellipsoids are added, to improve scene restoration accuracy. By Gaussian pruning, the redundant Gaussian ellipsoids are deleted to reduce a data amount of the Gaussian point cloud and improve the accuracy and effectiveness of the Gaussian point cloud.

[00124] In some alternative embodiments, determining the Gaussian point cloud data by using a pre-configured Gaussian splatting algorithm according to the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images includes the above steps 2410 to 2420.

[00125] In the embodiments of the present disclosure, the initial Gaussian point cloud is obtained by initializing the Gaussian point cloud based on the environment images of the multiple viewing angles and the depth maps of the environment images, and then the Gaussian point cloud is iteratively optimized based on the environment images of the multiple viewing angles, to obtain the Gaussian point cloud data conforming to the actual three-dimensional scene, so that the Gaussian ellipsoids in the Gaussian point cloud data can more finely simulate the three-dimensional scene, helping to improve scene restoration accuracy.

[00126] In some alternative embodiments, based on any one of the above embodiments, the step 250 of generating an around view image based on the Gaussian point cloud data may include: generating the around view image by rendering through a rendering engine based on the Gaussian point cloud data.

[00127] The rendering engine is a tool capable of rendering and generating an image based on the Gaussian point cloud data. The Gaussian point cloud data and camera parameters for rendering are input into the rendering engine, and image rendering is performed through the rendering engine to generate the around view image. The camera parameters for rendering include, but are not limited to, preset parameters of a virtual camera or parameters of the cameras with the multiple viewing angles on the vehicle, which may be specifically determined according to actual requirements.

[00128] For example, if the around view image is an image simulating a user inside the vehicle observing the surroundings, the camera parameters for rendering may be parameters of a virtual

camera simulating an observation viewing angle of the user inside the vehicle.

[00129] Optionally, the parameters of the virtual camera may be predetermined and stored, and the parameters of the virtual camera may include intrinsic parameters of the virtual camera and extrinsic parameters of the virtual camera relative to the vehicle. For example, a position of the virtual camera is a preset position in a vehicle cabin, a default pose (i.e., orientation) of the virtual camera is a traveling direction of the vehicle, and the virtual camera may be rotated to any direction.

[00130] During vehicle traveling, the Gaussian point cloud data may be obtained in real time or periodically according to the environment images of the multiple viewing angles at a current moment through steps 210 to 240; and then, based on the Gaussian point cloud data, the rendering engine is invoked to render the around view image of any viewing angle range. The rendered viewing angle range may be a preset viewing angle range, or may be determined by adjusting the camera parameters for rendering according to an operation of a user inside the vehicle, for example, adjusting a position and/or a pose of the virtual camera, where different poses correspond to different viewing angle ranges.

[00131] For example, during rendering, an around view image of a front viewing angle range observed from the preset position in the vehicle is rendered by default, and the around view image is displayed on a display screen in the vehicle; and a change of the displayed viewing angle range may be triggered by the user through an operation on the display screen or voice.

[00132] In one example, an interaction control may be displayed on the display screen, and the user may select a viewing angle range to be viewed through the interaction control; and, after the user determines the viewing angle range, the around view image of the viewing angle range is rendered and generated according to the viewing angle range selected by the user and the Gaussian point cloud data. In another example, the user may adjust the viewing angle range of the around view image displayed on the display screen by performing a swipe gesture on the display screen, and the around view image displayed on the display screen is dynamically updated according to the swipe gesture operation of the user. In yet another example, the user may adjust the displayed viewing angle range by voice, for example, "please rotate the picture 20 degrees to the left". In yet another example, the user may adjust an observation position of the displayed image by performing a zoom gesture on the display screen, then adjustment of a position of the camera for rendering is triggered; for example, when there is a toll station in front of the vehicle, the user performs a zoom-in gesture on the display screen to simulate an action of the user or the vehicle approaching the toll station, and the rendering engine adjusts the position of the camera for rendering and renders the Gaussian point cloud data into the around view

image displayed on the display screen according to the adjusted camera parameters, so that the scene image seen by the user from the display screen is an image of the toll station seen from a closer distance.

[00133] Optionally, an around view image of an entire around view angle range may be rendered based on the Gaussian point cloud data and the camera parameters for rendering, a partial image within the displayed viewing angle range is quickly extracted from the around view image according to a default displayed viewing angle range or a displayed viewing angle range selected by the user, and the partial image is displayed on the display screen; and the image displayed on the display screen may be dynamically updated according to a user operation.

[00134] Based on the obtained Gaussian point cloud data, the present disclosure does not limit a specific manner of rendering and generating the around view image and a viewing angle range for rendering, so that the user may view a scene picture of any viewing angle range from any position in the three-dimensional scene.

[00135] In the embodiments of the present disclosure, the around view image is generated by invoking the rendering engine for rendering based on the Gaussian point cloud data. Since the Gaussian point cloud data is capable of finely and accurately expressing a real three-dimensional scene, the around view image of any viewing angle rendered thereby can achieve a display effect of viewing in the real scene, thereby further improving the visualization effect.

[00136] The above embodiments of the present disclosure may be implemented individually or in any combination without conflict, which may be specifically set according to actual requirements, and the present disclosure imposes no limitation thereon.

[00137] Any of the around view image generation methods provided by the embodiments of the present disclosure may be executed by any appropriate device having a data processing capability, including but not limited to: a terminal device, a server, software and hardware in the terminal device, software and hardware in the server, etc. Alternatively, any of the around view image generation methods provided by the embodiments of the present disclosure may be executed by a processor; for example, the processor executes any of the around view image generation methods mentioned in the embodiments of the present disclosure by invoking corresponding instructions stored in a memory. Alternatively, any of the around view image generation methods provided by the embodiments of the present disclosure may also be executed by corresponding modules or units included in the exemplary apparatus described below and run by a processor.

Exemplary Apparatus

[00138] FIG. 7 is a schematic structural diagram of an around view image generation apparatus

provided by some exemplary embodiments of the present disclosure. The around view image generation apparatus of the embodiments of the present disclosure may be used to implement the around view image generation method provided by any one of the above embodiments of the present disclosure. As shown in FIG. 7, the around view image generation apparatus may include: a determination module 51, a first processing module 52, a second processing module 53, a third processing module 54, and a fourth processing module 55.

[00139] The determination module 51 is configured to determine environment images of multiple viewing angles.

[00140] The first processing module 52 is configured to reuse voxel depth data of surroundings of the vehicle, which is output by an occupancy network model pre-configured in the vehicle according to the environment images.

[00141] The second processing module 53 is configured to determine, based on the voxel depth data, depth maps respectively corresponding to the environment images of the multiple viewing angles.

[00142] The third processing module 54 is configured to determine Gaussian point cloud data based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images.

[00143] The fourth processing module 55 is configured to generate an around view image based on the Gaussian point cloud data.

[00144] FIG. 8 is a schematic structural diagram of an around view image generation apparatus provided by other exemplary embodiments of the present disclosure.

[00145] In some alternative embodiments, based on the embodiment shown in FIG. 7, the voxel depth data includes depth values of voxel grids occupied by an object around the vehicle; as shown in FIG. 8, the second processing module 53 may include: a first processing unit 531, a second processing unit 532, a third processing unit 533, and a fourth processing unit 534.

[00146] The first processing unit 531 is configured to, for any one of the environment images of the multiple viewing angles, determine, based on camera parameters corresponding to the environment image, a ray from an optical center of a camera to a target pixel in the environment image.

[00147] The second processing unit 532 is configured to determine, based on the voxel depth data, an occupied target voxel grid in the voxel depth data that intersects with the ray corresponding to the target pixel.

[00148] The third processing unit 533 is configured to use a depth value of the target voxel grid as a depth value of the target pixel.

[00149] The fourth processing unit 534 is configured to determine a depth map corresponding to the environment image based on depth values of target pixels.

[00150] FIG. 9 is a schematic structural diagram of an around view image generation apparatus provided by still other exemplary embodiments of the present disclosure.

[00151] In some alternative embodiments, as shown in FIG. 9, the apparatus of the embodiments of the present disclosure further includes: a fifth processing module 61, a sixth processing module 62, and a seventh processing module 63.

[00152] The fifth processing module 61 is coupled to the determination module 51, and is configured to determine an object segmentation result corresponding to the environment image, before determining the ray from the optical center of the camera to the target pixel in the environment image based on the camera parameters corresponding to the environment image. The object segmentation result includes one or more segmentation regions of objects.

[00153] The sixth processing module 62 is coupled to the fifth processing module 61, and is configured to determine, based on the object segmentation result, a sampling density corresponding to each of the segmentation regions.

[00154] The seventh processing module 63 is coupled to the sixth processing module 62 and the first processing unit 531, respectively, and is configured to determine target pixels from the environment image based on the sampling density corresponding to each of the segmentation regions.

[00155] In some alternative embodiments, based on any one of the above embodiments, the fourth processing unit 534 is specifically configured to: for any segmentation region, determine depth values of other pixels in the segmentation region based on the depth values of the target pixels in the segmentation region, where the other pixels are pixels in the segmentation region other than the target pixels; and determine the depth map corresponding to the environment image based on the depth values of the target pixels and the depth values of the other pixels in the segmentation region.

[00156] In some alternative embodiments, based on any one of the above embodiments, the third processing module 54 is specifically configured to: determine the Gaussian point cloud data by using a pre-configured Gaussian splatting algorithm based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images.

[00157] FIG. 10 is a schematic structural diagram of an around view image generation apparatus provided by yet other exemplary embodiments of the present disclosure.

[00158] In some alternative embodiments, based on any one of the above embodiments, as

shown in FIG. 10, the third processing module 54 includes: a first determining unit 541 and a second determining unit 542.

[00159] The first determining unit 541 is configured to determine an initial Gaussian point cloud based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images. The initial Gaussian point cloud includes parameters of a plurality of initialized Gaussian ellipsoids.

[00160] The second determining unit 542 is configured to iteratively adjust the initial Gaussian point cloud based on the environment images of the multiple viewing angles, to determine the Gaussian point cloud data.

[00161] In some alternative embodiments, based on any one of the above embodiments, the fourth processing module 55 is specifically configured to generate the around view image by rendering through a rendering engine based on the Gaussian point cloud data.

[00162] In some embodiments, the apparatus of the embodiments of the present disclosure includes, for example, but is not limited to, one or more of the following: a central processing unit, a cockpit processing unit, a cockpit-driving integrated processing unit, and other processors on the vehicle. The above modules may be modules in the corresponding processors.

[00163] The embodiments of the exemplary apparatus correspond to the embodiments of the exemplary method above in terms of implementation manner, and corresponding contents therebetween may be mutually referred to, combined with, or cited, which is not repeated herein. Beneficial technical effects corresponding to the embodiments of the exemplary apparatus may be referred to corresponding beneficial technical effects of the embodiments of the exemplary method above, which is not repeated herein.

Exemplary Electronic Device

[00164] FIG. 11 is a structural diagram of an electronic device provided by an embodiment of the present disclosure, including at least one processor 91 and a memory 92.

[00165] The processor 91 may be a central processing unit (CPU) or another form of processing unit having a data processing capability and/or an instruction execution capability, and may control other components in the electronic device 90 to perform a desired function.

[00166] Optionally, the processor 91 may include, for example, but is not limited to, one or more of the following: a central processing unit, a cockpit processing unit, a cockpit-domain integrated processing unit, and other processors on the vehicle.

[00167] The memory 92 may include one or more computer program products, and the computer program product may include various forms of computer-readable storage media, such as a volatile memory and/or a non-volatile memory. The volatile memory may include, for

example, a random access memory (RAM) and/or a cache. The non-volatile memory may include, for example, a read-only memory (ROM), a hard disk, a flash memory, etc. One or more computer program instructions may be stored on the computer-readable storage medium, and the processor 91 may run the one or more computer program instructions to implement the around view image generation method of the various embodiments of the present disclosure described above and/or other desired functions.

[00168] In one example, the electronic device 90 may further include: an input device 93 and an output device 94, which are interconnected through a bus system and/or another form of connection mechanism (not shown).

[00169] The input device 93 may further include, for example, a touch screen, a microphone, various sensors, and the like. The sensors may include, for example, an image sensor (such as a still camera, a camera, etc.), a lidar, a millimeter-wave radar, an ultrasonic radar, a positioning sensor, a pressure sensor, an air quality sensor, a temperature sensor, etc. The image sensor, the lidar, the millimeter-wave radar, the ultrasonic radar, etc., may be used for perception of the surroundings, that is, detecting dynamic and static objects in the surroundings. The dynamic and static objects may include, for example, static objects such as lane lines, road edges, arrows, sign boards, trees, and buildings, and dynamic objects such as surrounding vehicles, pedestrians, and riders. The positioning sensor is used for implementing positioning of a movable device (such as the ego-vehicle, a robot, etc.) where the electronic device is located. The positioning sensor may include, for example, an inertial measurement unit (IMU), a global positioning system (GPS), etc. The pressure sensor may be used for detecting a seat pressure. The temperature sensor may be used for detecting a temperature inside a cabin of the vehicle. The air quality sensor may be used for detecting air quality inside the cabin of the vehicle.

[00170] The output device 94 may output various information to the outside, and may include, for example, a display, a speaker, a communication network, and a remote output device connected thereto, etc.

[00171] Certainly, for simplicity, only some components in the electronic device 90 which are related to the present disclosure are shown in FIG. 11, and components such as a bus and an input/output interface are omitted. In addition, the electronic device 90 may further include any other appropriate components according to specific application situations.

Exemplary Computer Program Product and Computer-Readable Storage Medium

[00172] In addition to the above method and device, the embodiments of the present disclosure may further provide a computer program product, including computer program instructions. When the computer program instructions are run by a processor, the processor is enabled to

perform the steps of the around view image generation method of the various embodiments of the present disclosure described in the above "Exemplary method".

[00173] The computer program product may write program codes configured to perform operations of the embodiments of the present disclosure, by using any combination of one or more programming languages. The programming languages include an object-oriented programming language, such as Java, C++, etc., and further include a conventional procedural programming language, such as the 'C' language or a similar programming language. The program codes may be entirely executed on a user computing device, partially executed on a user device, executed as an independent software package, partially executed on a user computing device and partially executed on a remote computing device, or entirely executed on a remote computing device or a server.

[00174] In addition, the embodiments of the present disclosure may further include a computer-readable storage medium storing computer program instructions thereon; and when the computer program instructions are run by a processor, the processor is enabled to perform the steps of the around view image generation method of the various embodiments of the present disclosure described in the above "Exemplary method".

[00175] The computer-readable storage medium may be one or any combination of a plurality of readable media. The readable storage medium may be a readable signal medium or a readable storage medium. The readable storage medium includes, for example, but is not limited to, a system, an apparatus, or a device of electricity, magnetism, light, electromagnetism, infrared rays, or semiconductor, or any combination of the above. More specific examples (a non-exhaustive list) of the readable storage medium include: an electrical connection with one or more conducting wires, a portable disk, a hard disk, a random access memory (RAM), a read-only memory (ROM), an erasable programmable read-only memory (EPROM or a flash memory), an optical fiber, a portable compact disk read-only memory (CD-ROM), an optical storage device, a magnetic storage device, or any suitable combination of the above.

[00176] The basic principles of the present disclosure are described above in combination with specific embodiments. However, the advantages, superiorities, and effects mentioned in the present disclosure are merely examples but are not for limitation, and it cannot be considered that these advantages, superiorities, and effects are necessary for each embodiment of the present disclosure. In addition, the specific details described above are merely for illustrative and ease-of-understanding purposes, rather than limiting. The above details do not restrict the present disclosure to be implemented using the above specific details.

[00177] Those skilled in the art may make various modifications and variations to the present

disclosure without departing from the spirit and scope of the present disclosure. Accordingly, provided that these modifications and variations of the present disclosure fall within the scope of the claims of the present disclosure and their equivalents, the present disclosure is also intended to encompass such modifications and variations.

CLAIMS

1. An around view image generation method, comprising:

determining environment images of multiple viewing angles;

reusing voxel depth data of surroundings of a vehicle, which is output by an occupancy network model pre-configured in the vehicle according to the environment images;

determining, based on the voxel depth data, depth maps respectively corresponding to the environment images of the multiple viewing angles;

determining Gaussian point cloud data based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images; and

generating an around view image based on the Gaussian point cloud data.

2. The method according to claim 1, wherein the voxel depth data comprises depth values of voxel grids occupied by an object around the vehicle;

determining, based on the voxel depth data, the depth maps respectively corresponding to the environment images of the multiple viewing angles comprises:

for any one of the environment images of the multiple viewing angles, determining, based on camera parameters corresponding to the environment image, a ray from an optical center of a camera to a target pixel in the environment image;

determining, based on the voxel depth data, an occupied target voxel grid in the voxel depth data that intersects with the ray corresponding to the target pixel;

using a depth value of the target voxel grid as a depth value of the target pixel; and

determining a depth map corresponding to the environment image based on depth values of target pixels.

3. The method according to claim 2, wherein, before determining, based on the camera parameters corresponding to the environment image, the ray from the optical center of the camera to the target pixel in the environment image, the method further comprises:

determining an object segmentation result corresponding to the environment image, wherein the object segmentation result comprises one or more segmentation regions of objects;

determining, based on the object segmentation result, a sampling density corresponding to each of the segmentation regions; and

determining target pixels from the environment image based on the sampling density

corresponding to each of the segmentation regions.

4. The method according to claim 3, wherein determining the depth map corresponding to the environment image based on the depth values of the target pixels comprises:

for any segmentation region, determining depth values of other pixels in the segmentation region based on the depth values of the target pixels in the segmentation region, wherein the other pixels are pixels in the segmentation region other than the target pixels; and

determining the depth map corresponding to the environment image based on the depth values of the target pixels and the depth values of the other pixels in the segmentation region.

5. The method according to claim 1, wherein the determining Gaussian point cloud data based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images comprises:

determining the Gaussian point cloud data by using a pre-configured Gaussian splatting algorithm based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images.

6. The method according to claim 5, wherein determining the Gaussian point cloud data by using the pre-configured Gaussian splatting algorithm based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images comprises:

determining an initial Gaussian point cloud based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images, wherein the initial Gaussian point cloud comprises parameters of a plurality of initialized Gaussian ellipsoids; and

iteratively adjusting the initial Gaussian point cloud based on the environment images of the multiple viewing angles, to determine the Gaussian point cloud data.

7. The method according to any one of claims 1 to 6, wherein generating the around view image based on the Gaussian point cloud data comprises:

generating the around view image by rendering through a rendering engine based on the Gaussian point cloud data.

8. An around view image generation apparatus, comprising:

a determination module, configured to determine environment images of multiple viewing angles;

a first processing module, configured to reuse voxel depth data of surroundings of a vehicle, which is output by an occupancy network model pre-configured in the vehicle according to the environment images;

a second processing module, configured to determine, based on the voxel depth data, depth maps respectively corresponding to the environment images of the multiple viewing angles;

a third processing module, configured to determine Gaussian point cloud data based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images; and

a fourth processing module, configured to generate an around view image based on the Gaussian point cloud data.

9. The apparatus according to claim 8, wherein the voxel depth data comprises depth values of voxel grids occupied by an object around the vehicle; and

wherein the second processing module comprises: a first processing unit, a second processing unit, a third processing unit, and a fourth processing unit;

the first processing unit is configured to, for any one of the environment images of the multiple viewing angles, determine, based on camera parameters corresponding to the environment image, a ray from an optical center of a camera to a target pixel in the environment image;

the second processing unit is configured to determine, based on the voxel depth data, an occupied target voxel grid in the voxel depth data that intersects with the ray corresponding to the target pixel;

the third processing unit is configured to use a depth value of the target voxel grid as a depth value of the target pixel; and

the fourth processing unit is configured to determine a depth map corresponding to the environment image based on depth values of target pixels.

10. The apparatus according to claim 9, further comprising: a fifth processing module, a sixth processing module, and a seventh processing module;

the fifth processing module is configured to determine an object segmentation result corresponding to the environment image, before determining the ray from the optical center of the camera to the target pixel in the environment image based on the camera parameters corresponding to the environment image, wherein the object segmentation result comprises one

or more segmentation regions of objects;

the sixth processing module is configured to determine, based on the object segmentation result, a sampling density corresponding to each of the segmentation regions; and

the seventh processing module is configured to determine target pixels from the environment image based on the sampling density corresponding to each of the segmentation regions.

11. The apparatus according to claim 10, wherein the fourth processing unit is configured to: for any segmentation region, determine depth values of other pixels in the segmentation region based on the depth values of the target pixels in the segmentation region, wherein the other pixels are pixels in the segmentation region other than the target pixels; and determine the depth map corresponding to the environment image based on the depth values of the target pixels and the depth values of the other pixels in the segmentation region.

12. The apparatus according to claim 8, wherein the third processing module is configured to: determine the Gaussian point cloud data by using a pre-configured Gaussian splatting algorithm based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images.

13. The apparatus according to claim 12, wherein the third processing module comprises: a first determining unit and a second determining unit;

the first determining unit is configured to determine an initial Gaussian point cloud based on the environment images of the multiple viewing angles and the depth maps respectively corresponding to the environment images, wherein the initial Gaussian point cloud includes parameters of a plurality of initialized Gaussian ellipsoids; and

the second determining unit is configured to iteratively adjust the initial Gaussian point cloud based on the environment images of the multiple viewing angles, to determine the Gaussian point cloud data.

14. A computer-readable storage medium, comprising a computer program stored thereon, wherein the computer program, when executed, is configured to implement the around view image generation method according to any one of claims 1 to 7.

15. An electronic device, comprising: a processor; and a memory configured to store executable instructions executed by the processor;

wherein the processor is configured to read the executable instructions from the memory

and execute the executable instructions to implement the around view image generation method according to any one of claims 1 to 7.