(12) UK Patent Application (19) GB (11) 2585197 (13) A

(43) Date of A Publication 06.01.2021

(21) Application No: 1909444.0

(22) Date of Filing: 01.07.2019

(51) INT CL:
G06T 7/593 (2017.01) *G06T 7/73* (2017.01)
H04N 13/271 (2018.01)

(56) Documents Cited:
CN 109584290 A CN 106600583 A
CN 104574391 A
Jure Zbontar, Yann LeCun, Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches, Journal of Machine Learning Research, 2016, 17(65), 1-32
Nikolaue Mayer et al, A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation, 2016, IEEE Conference on Computer Vision and Pattern Recognition, pages 4040-4048

(58) Field of Search:
INT CL G06N, G06T, H04N
Other: WPI, EPODOC, Patent Fulltext

(71) Applicant(s):
Sony Interactive Entertainment Inc.
1-7-1 Konan, Minato-ku, Tokyo 108-0075, Japan

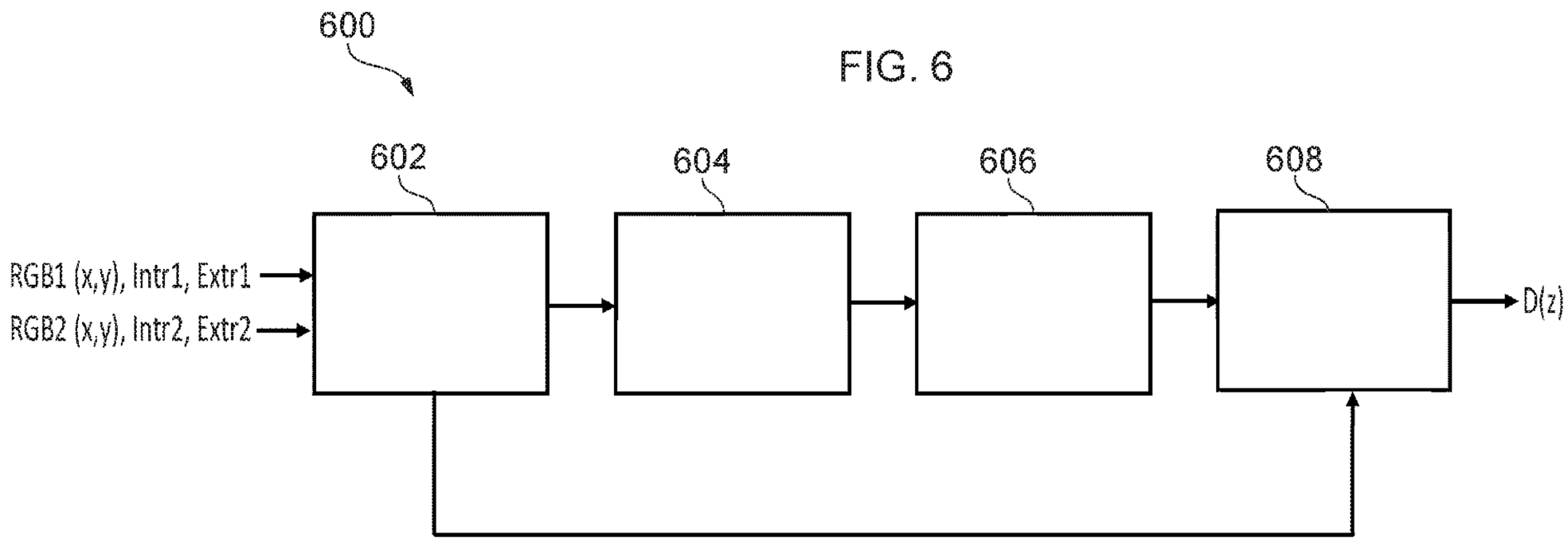
(72) Inventor(s):
Rajeev Gupta
Fabio Cappello
David Erwan Damien Uberti
Nigel John Williams

(74) Agent and/or Address for Service:
D Young & Co LLP
120 Holborn, LONDON, EC1N 2DY, United Kingdom

(54) Title of the Invention: **Method and system for obtaining depth data**
Abstract Title: **Obtaining depth information from a scene using different viewpoints and machine learning**

(57) Obtaining depth information of a scene comprises obtaining at least two colour images of a scene, each corresponding to a different but partially overlapping viewpoint. The camera extrinsics (e.g. position and orientation) and intrinsics (e.g. focal length) associated with the colour images are also obtained. Each colour image is input to a trained machine learning model to generate a representation of that colour image. Corresponding regions of the scene in the representations of the images are then identified based on a similarity between features of the representations generated by the trained machine-learning model. Depth information is obtained for at least some of the scene based on the intrinsics and extrinsics associated with the colour images and the corresponding features identified as corresponding in the feature representations generated by the machine learning model. A system 600 may comprise an input unit 602, a feature extraction unit 604 which may correspond to a convolutional neural network, a feature mapping unit 606 (e.g. forming a disparity map) and a depth analyser 608.

FIG. 6



GB 2585197 A

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

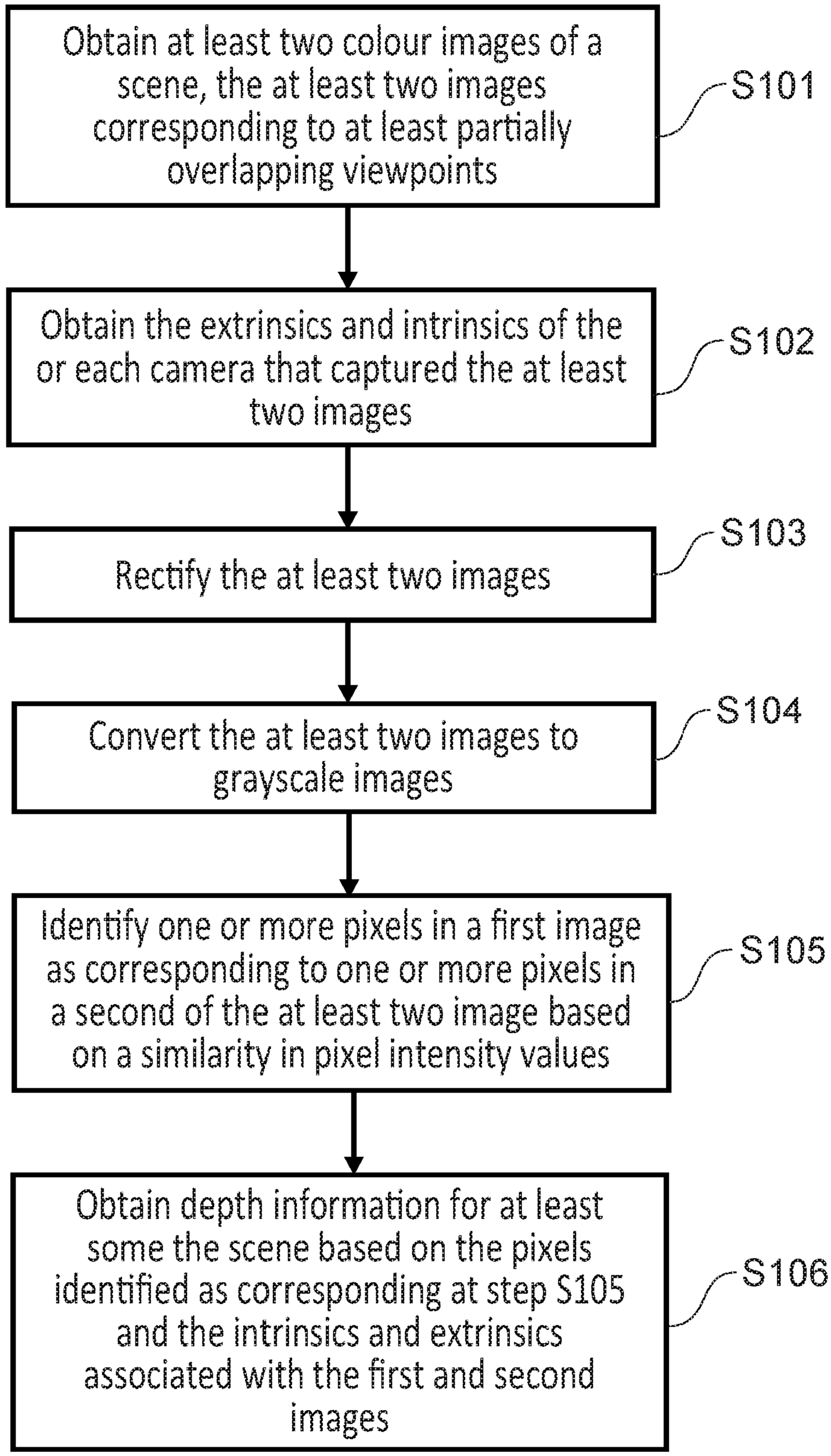


FIG. 1 (PRIOR ART)

24 07 19

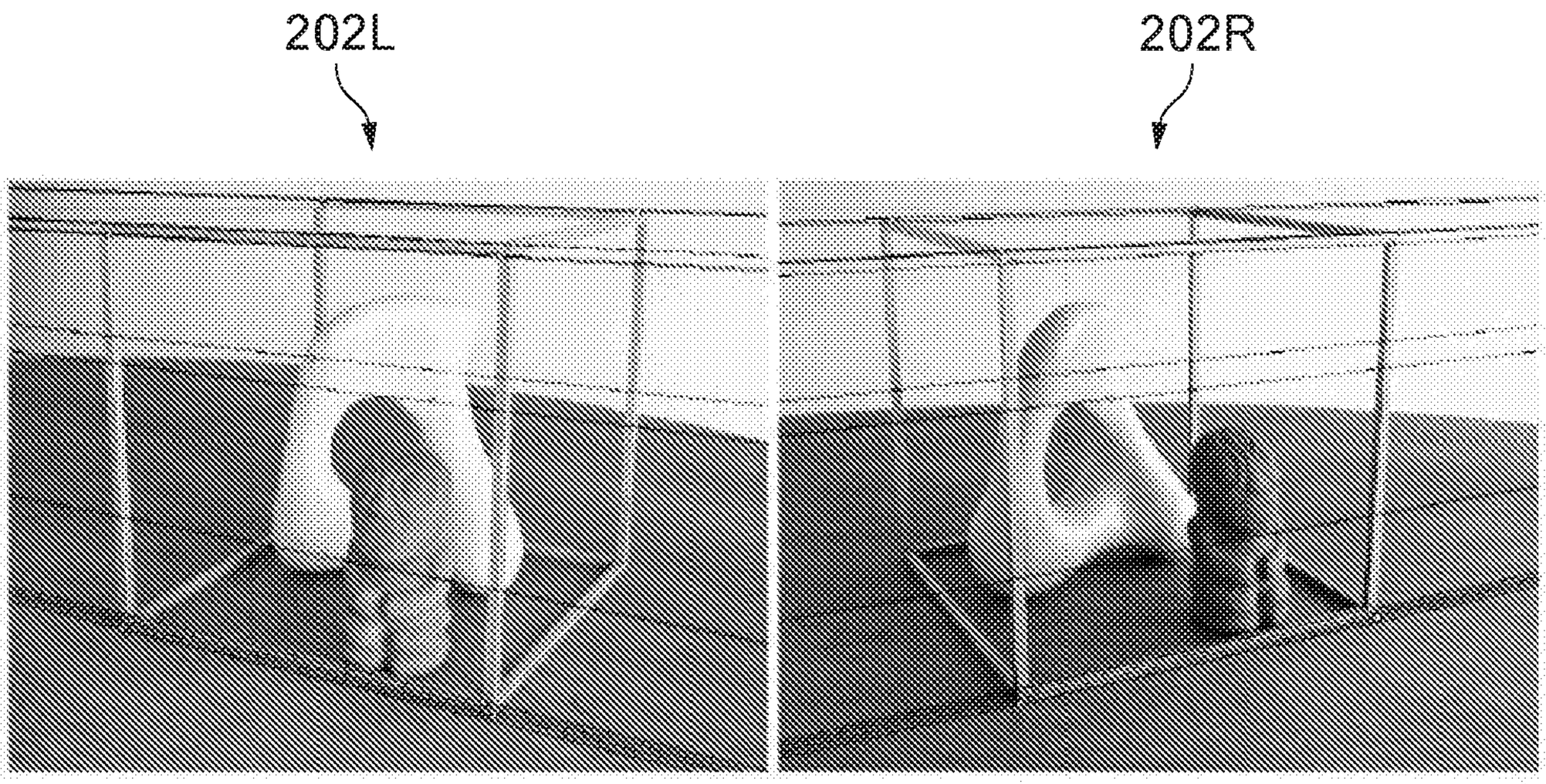


FIG. 2

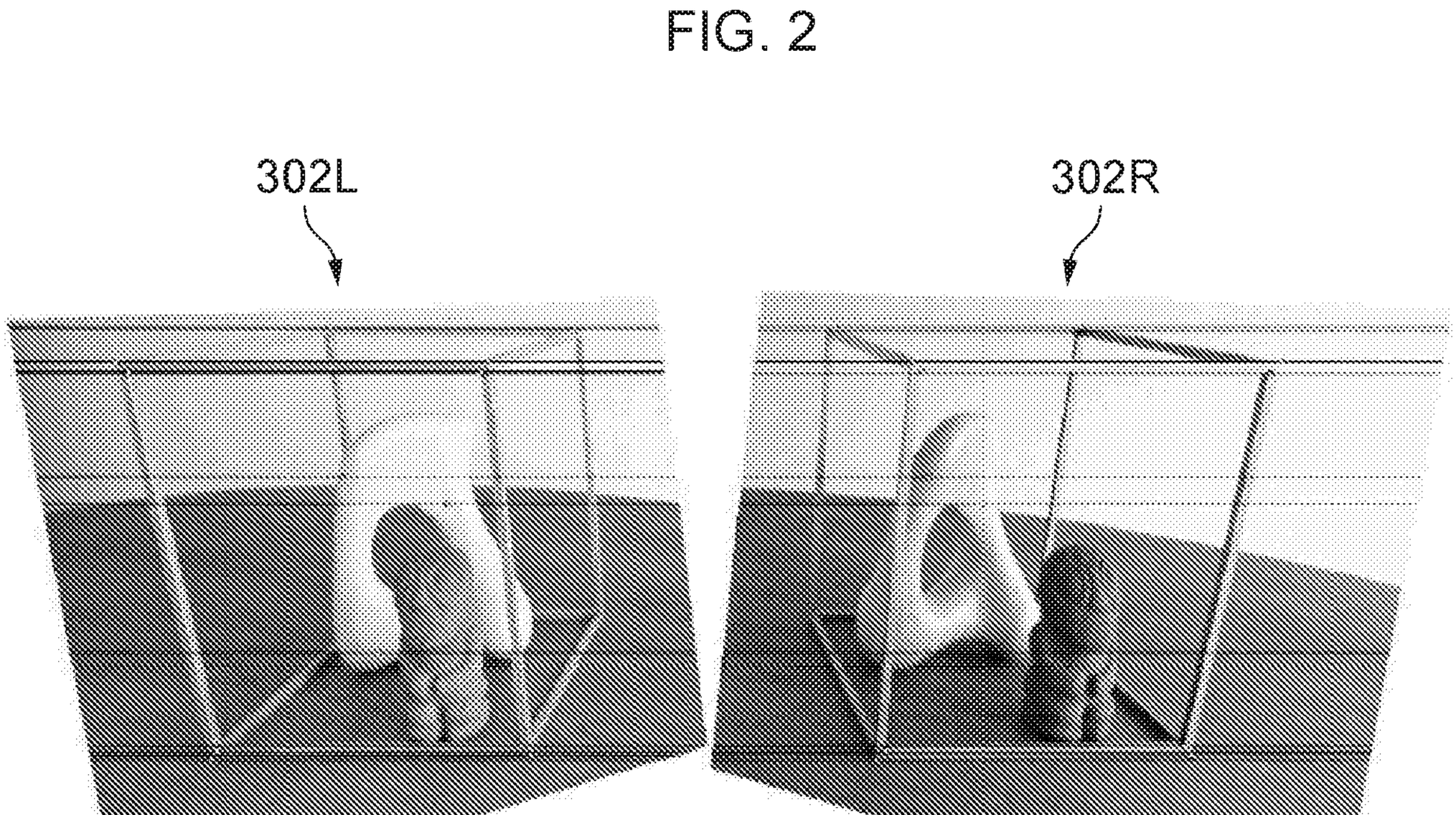


FIG. 3

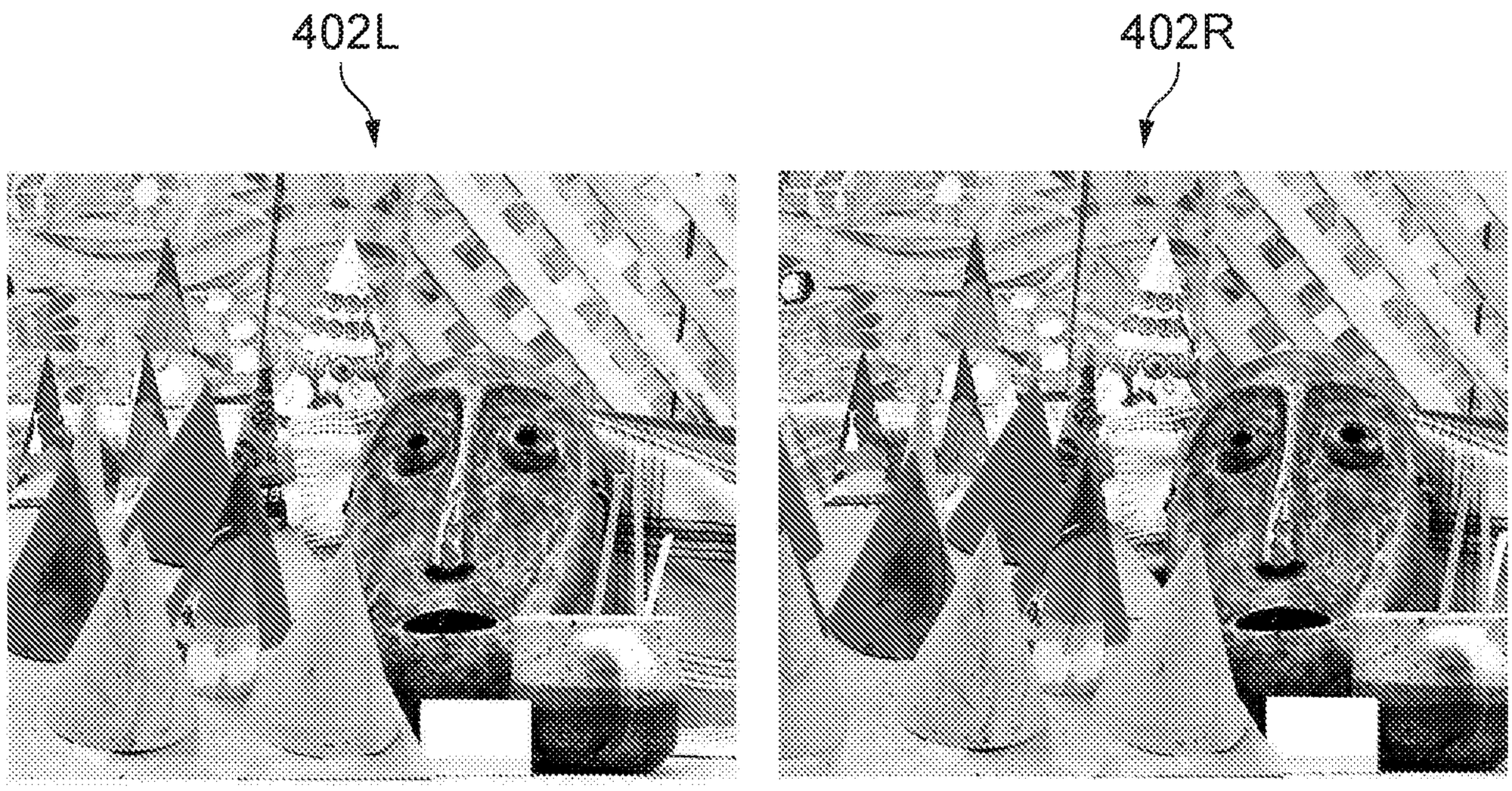

402L
402R
24 07 19
Basic block matching with sub-pixel accuracy (block size = 11)
404
40
20
0
-20
-40

FIG. 4

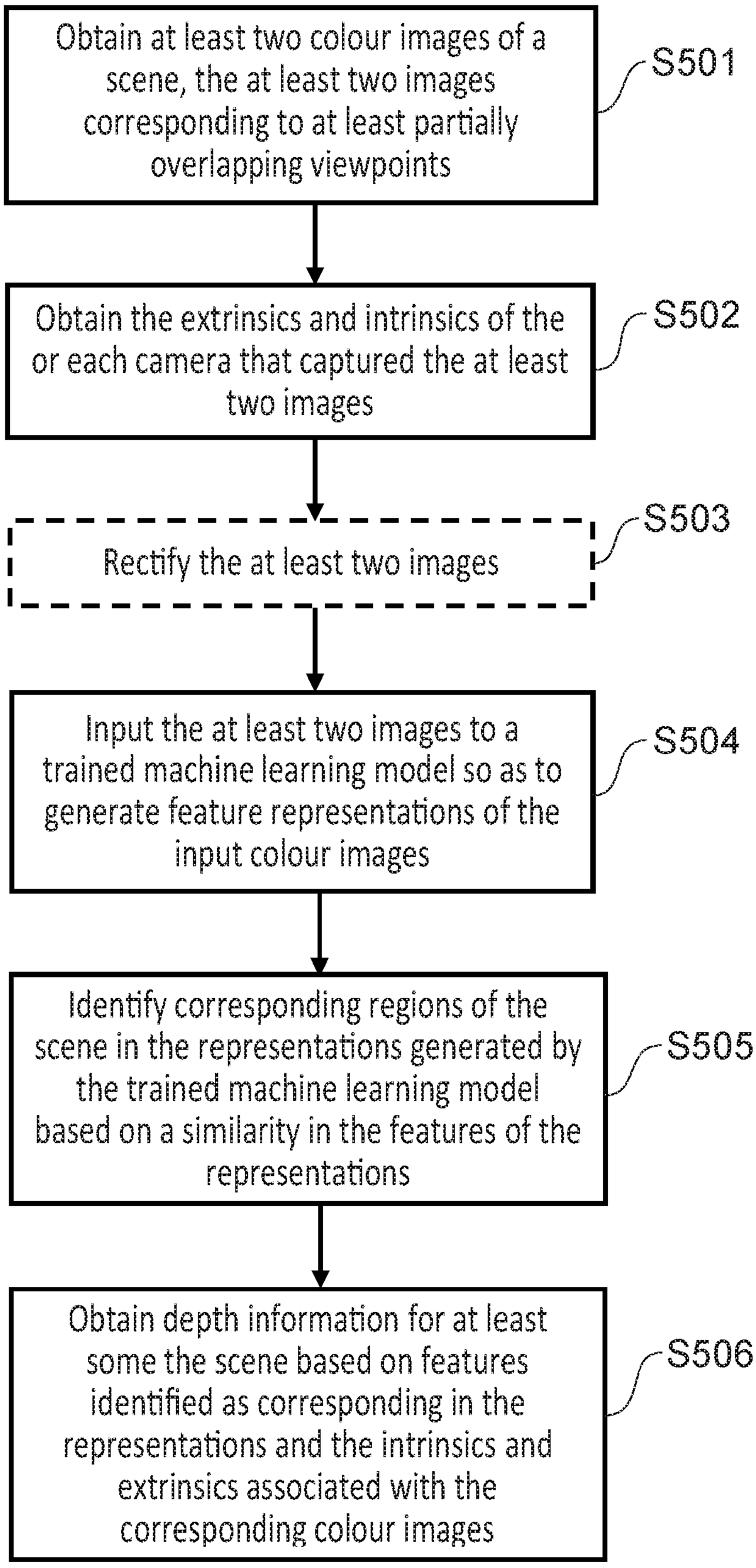

Obtain at least two colour images of a scene, the at least two images corresponding to at least partially overlapping viewpoints
S501
Obtain the extrinsics and intrinsics of the or each camera that captured the at least two images
S502
Rectify the at least two images
S503
Input the at least two images to a trained machine learning model so as to generate feature representations of the input colour images
S504
Identify corresponding regions of the scene in the representations generated by the trained machine learning model based on a similarity in the features of the representations
S505
Obtain depth information for at least some the scene based on features identified as corresponding in the representations and the intrinsics and extrinsics associated with the corresponding colour images
S506

FIG. 5

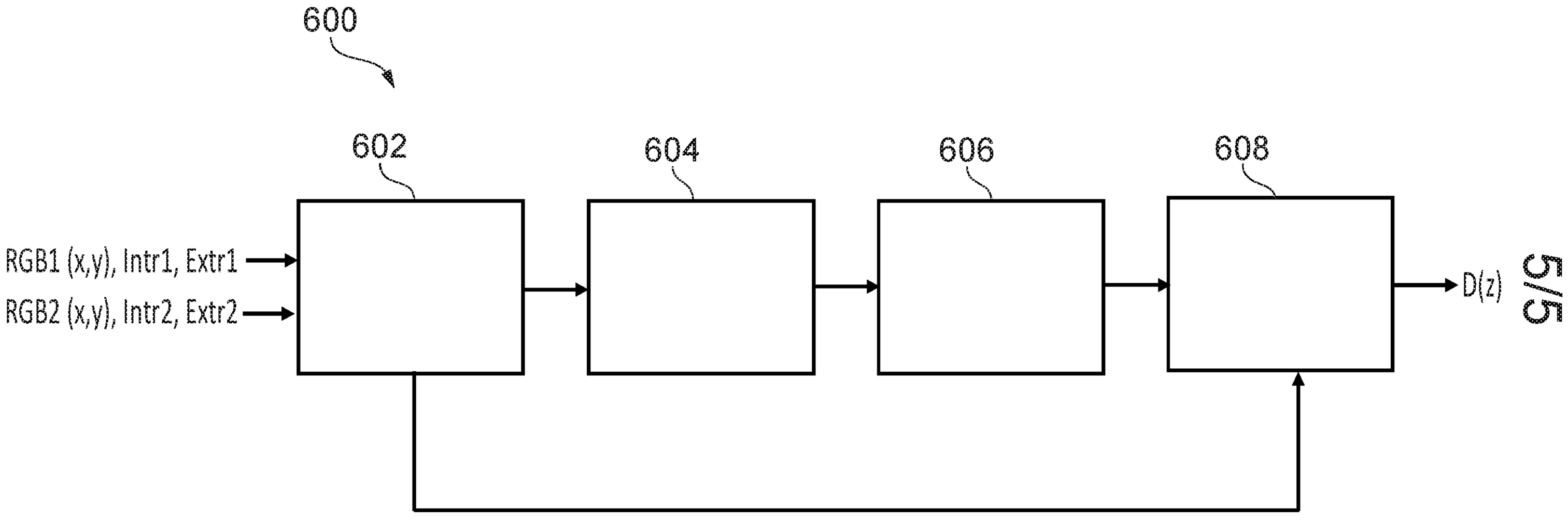


FIG. 6

Intellectual Property Office

The following terms are registered trade marks and should be read as such wherever they occur in this document:

PLAYSTATION

METHOD AND SYSTEM FOR OBTAINING DEPTH DATA

Technical Field

The present disclosure relates to a method and system for obtaining depth information of a scene.

Background

In recent years, driven at least in part by the improvements made in display technology, there has been an increase in the demand for interactive content that is able to offer an immersive experience to a user. For example, the increase in the number and quality of virtual reality (VR) and augmented reality (AR) devices lends itself to the provision of immersive experiences, while the development of televisions and other display devices that offer increased resolution, refresh rate, and colour reproduction (for example) also act as increasingly suitable devices for the provision of such content. In addition to this, advances in computing and graphics technology have contributed to the increase in suitable content that may be made available.

While video games may be provided that can offer such an experience, the approaches taken to provide viewer immersion in video games may not be applicable to captured video content such as movies or sports events. For example, when generating video game content it is common that the locations and properties of all objects in the environment are known and other features, such as lighting information, are also able to be calculated. Such information is often not available for captured video content, and therefore techniques applicable to video games to enable the provision of more immersive content are not considered to be widely applicable.

One example of captured video content that is adapted for increased immersion of a user is that of three-dimensional video. Consumer devices are available that are operable to display content that may be viewed (often aided by a corresponding set of glasses that are configured to enable the viewing of three-dimensional content) in a manner that causes the user to perceive the content as having significant depth despite the use of a two-dimensional display.

However, one drawback with such systems is that the viewpoint that is adopted by the user is often pre-defined (such as tied to the camera position in a movie) or severely limited (such as allowing a user to switch between a number of such pre-defined viewpoints).

This may serve to reduce the level of immersion that is experienced by the user when viewing the content, particularly in a VR context, as despite appearing three-dimensional there is no corresponding motion of the viewpoint as the user moves their head as would be expected were the user to move their head when viewing real-world content. The resulting disconnect between the viewpoint and the user's motion can lead to a sense of discomfort for the user, in addition to the loss of immersion.

Similarly, the restrictions placed upon the viewpoint location may be made more noticeable when a user is provided with more immersive content, as the user may be more inclined to try and explore the displayed environment. This can lead to the user attempting to relocate the viewpoint to a desired location

in the virtual environment, and becoming frustrated when such a relocation is not possible within the constraints of the provided content. Examples of such changes in viewpoints include a user moving their head in a VR system in order to look around an environment, or an input using a controller or the like in a two-dimensional display arrangement.

It is in view of the above considerations that so-called free viewpoint systems have been developed. The object of such systems is to provide content which a user is able to navigate freely, such that a viewpoint may be selected freely (or at least substantially so) within a virtual environment and a corresponding view is able to be provided to a user. This can enable a user to navigate between any number of viewpoints within the virtual environment, and/or for multiple users to occupy corresponding preferred viewpoints within the virtual environment. These viewpoints may be distributed about an environment in a discrete fashion, or the changing of viewpoints may be a result of a continuous motion within the environment, or content may incorporate elements of each of these.

Generally, free viewpoint systems require both depth and colour information of a scene to be captured, in order for the scene to be reconstructed in 3D. Usually, the colour information is used to derive the x-y coordinates of one or more objects in the scene, and depth information used to derive the corresponding z-coordinates. One known technique for obtaining depth information is stereo-matching, which involves extracting depth information from two images having at least partially overlapping fields of view. By calculating the distance between corresponding pixels in the images, a disparity map can be generated. This disparity map can then be converted to depth information using the intrinsics and extrinsics of the (or each) camera that captured the two images.

Typically, stereo-matching is performed on grayscale images, with corresponding pixels across the images being identified based on a similarity in pixel intensity values. The use of grayscale images reduces the variability in pixel values and thus enables corresponding pixels across images to be identified more easily. However, the accuracy of stereo-matching can be limited where, for example, there is a mismatch in exposure between the two images forming the stereoscopic image. This may arise, for example, as a result of changing lighting conditions between successive captures of the scene, and/or for example due to reflection/glare changing the apparent brightness of the corresponding part of an object when viewed from different viewpoints. More generally, it may be that there are unforeseen (and in most cases, unintended) differences in the capture conditions of separate images, and that, as a result, performing stereo-matching on these images lacks accuracy. In turn, this can limit the accuracy with which a real-life scene can be represented in three dimensions.

It would therefore be desirable if these differences in capture conditions could be accounted for, and a more robust method and system for obtaining depth information in a free viewpoint system provided. The present disclosure seeks to address or at least alleviate this problem.

Summary

The present disclosure is defined by the appended claims.

Brief Description of the Drawings

A more complete appreciation of the disclosure and many of the attendant advantages thereof will be readily obtained as the same becomes better understood by reference to the following detailed description when considered in connection with the accompanying drawings, wherein:

Figure 1 shows schematically an example of a conventional stereo-matching method;

Figure 2 shows an example of two images that may be used in a stereo-matching process;

Figure 3 shows an example of a pair of rectified images;

Figure 4 shows an example of a disparity map generated from the pair of rectified images shown in Figure 3;

Figure 5 shows schematically an example of a stereo-matching method in accordance with the present disclosure; and

Figure 6 shows schematically an example of a system for performing stereo-matching in accordance with the present disclosure.

Detailed Description

Figure 1 shows schematically an example of a conventional stereo-matching method. As can be seen in Figure 1, the method comprises a first step, S101 of obtaining at least two images of a scene. Each image corresponds to a different viewpoint, with each viewpoint at least partially overlapping. That is, each image comprises a portion corresponding to the same extent of the scene, albeit from a different perspective relative to the other image(s). The images may be colour images, such as e.g. RGB or YUV images.

Each image may be captured by a single camera that has been moved to a different pose (i.e. position and / or orientation) for each subsequent capture, e.g. by a camera operator. In the present disclosure, a camera refers to a device having at least an image sensor and one or more lenses for focussing light onto the image sensor. The camera may be a standalone device, e.g. a professional video camera, or be a component of another device, such as e.g. a smartphone, head-mountable display (HMD), laptop, etc.

In some embodiments, the image may be captured by a stereoscopic camera, i.e. a camera comprising at least two image sensors, with each image sensor being associated with one or more lenses for focussing light onto the image sensors, and a respective aperture located therebetween. An example of a stereoscopic camera is the PlayStation Camera™. Generally, the individual cameras forming the stereoscopic camera will be displaced relative to one another such that each image corresponds to a different viewpoint, with the displacement being such that at least some of the respective viewpoints overlap. For stereoscopic cameras it may not be necessary to move the stereoscopic camera to different poses in order to obtain depth data, but this may still be desirable, e.g. to obtain more depth data, compensate for any occlusion, etc.

An example of two images that may be used for stereo-matching is shown in Figure 2. In Figure 2, a first image 202L (shown on the left) corresponds to an image captured by a camera at first pose, and the second image 202R (shown on the right) correspond to an image of the same scene captured by a (or the) camera at a second pose.

At a second step, S102, the extrinsics (e.g. position and / or orientation) and intrinsics (e.g. focal length) associated with the at least two images are obtained. The extrinsics and intrinsics of the or each camera that captured the at least two images may be obtained using known calibration techniques. For example, by capturing multiple images of a calibration pattern, such as a checkerboard, and using the correspondences in the captured images to solve for the extrinsics and intrinsics of the or each camera. Meanwhile, for a stereoscopic camera, the extrinsics and possibly intrinsics of the built-in camera pair relevant to the relationship between the generated two images may be known in advance.

At a third step, S103, image rectification is performed on the at least two images. As is known in the art, image rectification involves reprojecting one or more of the images in a stereoscopic pair such that the epipolar lines of each image are parallel in the rectified image plane. This may be achieved, for example, by finding a set of matching key points between the two images using e.g. a Scale-Invariant Feature Transform (SIFT) algorithm or Speeded-Up Robust Features (SURF) algorithm, and then applying to transformations to one or both of the images so as to bring the key points into alignment. Figure 3 shows an example of the images shown in Figure 2, following image rectification.

By generating rectified images from the captured images, the space over which corresponding pixels, or blocks of pixels, are to be searched for in the image pairs can be reduced. For example, it may be that, following rectification, corresponding blocks need only be searched for horizontally, across corresponding rows in the rectified images.

It will be appreciated that, depending on the nature in which the camera(s) are arranged when capturing the images, it may not always be necessary to rectify the captured images prior to performing the stereo-matching. For example, if the camera(s) is (are) offset but aligned horizontally, then there may be no need for aligning the images so as to bring the images into alignment.

At a fourth step S104, the at least two colour images are converted to grayscale images. The conversion to grayscale images ensures that each image is defined in terms of pixel intensity values, with each pixel being defined in terms of how black or white that pixel is. A pixel intensity value of zero may indicate a black pixel; a pixel intensity value of 255 may indicate a white pixel, and values in between may represent various shades of grey. By converting the images to grayscale, there is less variation in the pixel values of each image, and so corresponding blocks of pixels across the images can be compared more easily.

At a fifth step S105, pixels or groups of pixels in a first image are identified as corresponding to pixels (or groups of pixels) in a second image, based on a similarity in the pixel intensity values of those pixels. This may involve, for example, performing block matching between at least some of the blocks of pixels making up each image. Examples of block-matching techniques include e.g. a sum of squared

difference measure (SSD), normalized cross correlation measure (NCC), etc. Generally, block matching involves determining, for each block in a first image, a difference in overall pixel value (e.g. sum of pixels in a block) with at least some of the blocks in the second image. It may be determined that a block in the first image corresponds with a block in the second image when the difference in overall pixel value for the two blocks is less than the difference in overall pixel value for the other blocks with which the block in the first image has been compared.

Step S105 may be equivalent to generating a disparity map from the at least two images, wherein each pixel of the disparity map encodes a difference in coordinates of corresponding image points (i.e. pixels) in an image pair. The pixel intensities of the disparity map may correspond to disparity, with corresponding image points that are separated further from each other across the two images being associated with a pixel intensity values that are closer to zero (i.e. more black), and nearer image points being associated with pixel intensity values that are closer to 255 (i.e. more white). Generally, parts of the scene located further from the camera(s) will have a smaller disparity than parts of the scene that are located closer to the camera(s). An example of a disparity map generated from the two images shown in Figure 2 is shown in Figure 4, although in this case the map, originally in colour, (red for nearby, high displarity pixels, and blue for distant, low disparity pixels), does not follow the above described greyscale intensity scheme. Nevertheless a visual inspection of the source images and the disparity map serves to illustrate the technique.

The searching for corresponding blocks across the at least two images may be performed in any appropriate manner. As mentioned above, in some examples, the at least two images may be rectified and so step S105 may be performed for the at least two rectified images. In such examples, this may involve searching for corresponding blocks in a row-wise manner, i.e. for a given block in the first image, searching a corresponding row in the second image for a matching block.

In other examples, it may be that a range of pixel values that are to be searched in the second image, relative to a given block in the first image, is defined as part of step S105. This may involve, for example, defining an offset in one or more directions relative to the block of pixels in the first image, over which matching blocks in the second image are to be searched.

As will be appreciated, there will be regions within each of the at least two images that are specific to a given camera, or camera pose, and from which no depth information can be obtained. This is because, for these regions, there is (or are) no corresponding pixels in the other image, from which depth information can be obtained. The method may thus comprise a step (not shown) of cropping the at least two images, such that the pixels in one image having no corresponding pixels in the other image are discarded from the image data.

At a sixth step S106, depth information is generated for the scene based on the pixels matched at step S105 for the at least two images and the intrinsics and extrinsics associated with the at least two images. Here, each image is associated with the intrinsics (e.g. focal length) and extrinsics (e.g. position and / or orientation) in the sense that the camera that captured the image will have had a respective pose

and one or more intrinsic parameters associated therewith, at the time of capture. For each of the at least two images, at least the pose of the camera will be different, if not the intrinsic parameters.

The depth information provides an indication of the distance of one or more objects in the scene, relative to the camera(s). The depth information may be obtained using the following equation:

$$Z = \frac{Bf}{x - x\prime} \quad \text{(Equation 1)}$$

Where Z corresponds to the z-coordinate of a point within the real world, 3D scene; f corresponds to the focal length of the camera(s) that captured the images in an image pair; B corresponds to the baseline of the cameras, i.e. the difference in position of the camera(s) for each image; and x – x' corresponds to the disparity, that is, the distance between corresponding points (i.e. pixels) in the image plane of the image pair. It may be that, for corresponding pixels in an image pair (having a disparity $x_1 - x_1'$), a corresponding z-coordinate, Z_1, is determined using equation 1. The baseline for the camera(s) associated with the images may be determined using the extrinsics (i.e. pose) of the camera(s) obtained at step S102.

It will be appreciated that equation 1 corresponds to a relatively simple example, where corresponding image points are separated horizontally, due to the configuration of the camera(s) and / or image rectification having been performed on each of the images in an image pair. In some examples, it may be that a modified version of equation 1 is required, e.g. where different focal lengths are associated with the images in the image pair. In any case, depth information can be obtained provided that the disparity of corresponding points in an image pair is known, and the extrinsics and intrinsics of the camera(s) that captured those images is also known.

Figure 1 shows schematically an example of a conventional stereo-matching method wherein corresponding image points in grayscale images are identified and used, together with the intrinsics and extrinsics of the or each camera, to generate depth information for the parts of the scene that are within the overlapping fields of view of the camera(s). A drawback with conventional stereo-matching methods such as those described above, is that the accuracy with which corresponding image regions can be identified across image pairs may be reduced when there are differences in the capture conditions of the images in a respective image pair. These differences in capture conditions may include, for example, a change in exposure of the second image compared with the first image in the image pair, and/or a change in glare or reflection for images at different viewpoints. As a result of this, pixels in the second image may appear artificially brighter or darker compared with pixels in the first image. In turn, this can result in a mismatch between corresponding image regions, with blocks that would otherwise correspond, no longer being closest in terms of the difference between the pixel values associated with those blocks.

The differences in capture conditions between successive captures may also include, for example, a change in depth of field, focal length, etc. such that the intensity of pixel values for the same points in the scene no longer correspond as closely as expected. Generally, the differences in capture conditions will be unintended, and may occur due to e.g. temperature changes (which may affect e.g. shutter speed), sudden changes in lighting of the scene, human error when operating the cameras, etc. Clearly, it would

be desirable if these differences in capture conditions could be compensated for when performing stereo-matching on an image pair.

Moreover, the conversion of colour images to grayscale images in conventional stereo-matching also results in a loss of detail. Rather than each image point being represented by e.g. three colour channels, each image point is reduced to a one-dimensional intensity value. It would thus desirable if this step of converting to grayscale images could be avoided, such that corresponding image points could be more clearly distinguished, based on the similarity of colour of those image points.

As will be appreciated, a stereo-matching method of increased accuracy will allow for more accurate extraction of depth data for a scene, thus enabling that scene to be reconstructed more accurately in three-dimensions. Such a method will now be described in relation to Figure 5.

Figure 5 shows schematically an example of a stereo-matching method in accordance with the present disclosure.

At a first step, S501, at least two colour images of a scene are obtained. Each image corresponds to a different viewpoint of the scene, with the viewpoints overlapping at least partially.
The colour images may comprise e.g. RGB or YUV images, saved in e.g. JPEG, GIF, RAW formats. The colour images may be captured in any of the manners described previously in relation to Figure 1, using any of the previously described devices.

In some examples, the at least two colour images may comprise high dynamic range images (HDR). That is, each image may comprise a composite image formed of at least two images captured at different exposures. Typically, HDR images are formed of two, three, or nine different exposure photos blended together. Generally, HDR images allow for greater detail in the scene to be captured, with the detail captured for lower exposures being used to preserve the detail that is lost for higher exposures (and vice versa). It may be advantageous to use HDR images as these will generally contain more detail, and so there may be more features within the image for which stereo-matching can be performed. The HDR images may comprise colour HDR images.

At a second step, S502, the camera intrinsics and extrinsics associated with the at least two colour images is obtained. The intrinsics and extrinsics may be obtained as described previously in relation to Figure 1, i.e. using known camera calibration techniques.

At a third, optional step, S503, the at least two colour images may be rectified before being input to a trained machine learning model (see step S504, below).

At a fourth step S504, each colour image is input to a trained machine learning model, the machine learning model being trained to generate a representation of each input colour image. The machine learning model is trained with a plurality of images of the same scene, captured from the same viewpoint, with each image being varied in at least one aspect with respect to the other training images. The machine learning model is trained to learn a representation of the scene that is independent of the variations in the plurality of training images of the scene. In this way, the machine learning model is able

to normalize the scene. The machine learning model may be trained with multiple images of different real and / or virtual scenes, so as to learn representations of the features within those scenes. The training of the machine learning model will be described in more detail later (see: 'training the machine learning model').

It should be noted that the images used to train the machine learning model need not necessarily correspond to the same scene as that for which the at least two colour images have been obtained. The machine learning model may be trained with a set of training images, where the images in the set correspond to images of the same scene, captured from the same viewpoint. The machine learning model may be trained with multiple sets of training images, with each set corresponding to a different scene and / or the same scene but captured from different viewpoints. Generally, the more types of scene that the machine learning model is trained with, the more generic the machine learning model will be, and thus the more applicable the present stereo-matching method to different types of scene. However, as will be appreciated, training the machine learning model on a variety of scenes will be computationally expensive and require more training data. In some use cases, it may be sufficient to train the machine learning model with images of one or a few scenes, where speed and / or a specific use is prioritised over generality.

The machine learning model may comprise a trained domain adversarial network.

The machine learning model may be trained to map at least some of the pixels of the input colour images to respective n-dimensional feature vectors. This mapping may be performed on a per-pixel basis, i.e. each pixel may be mapped to a respective n-dimensional vector. Alternatively, or in addition, this mapping may be performed on a per-block basis, i.e. each block may be mapped to a respective n-dimensional vector that is representative of that block. Each pixel or group of pixels may be mapped to feature vectors of the same dimensionality (this dimensionality being determined by the training of the machine learning model). The n-dimensional feature vector may correspond to a 32-dimensional vector that is output by a trained domain adversarial network, for example. The vectors may correspond to vectors in a feature space, with the dimensionality of that space corresponding to the dimensionality of the feature vectors. The distance between vectors (generated for different images) in the feature space may be indicative of how likely those vectors correspond to the same point in space.

At a fifth step, S505, corresponding regions of the scene in the representations generated by the trained machine learning model are identified based on a similarity between the features of the representations generated by the trained machine learning model. Although the term 'region' is used, it may be that the matching is between pixels or groups of pixels across the at least two colour images.

Step 505 may involve determining a distance between at least some of the n-dimensional feature vectors generated for a first input colour image and at least some of the n-dimensional feature vectors generated for a second input colour image (the two images corresponding to an image pair). An n-dimensional feature vector generated for a first colour image feature may be identified as corresponding with an n-dimensional feature vector generated for a second colour image based on the distance between those vectors being less than the distance of the other n-dimensional feature vectors generated for the

second colour image (and with which the n-dimensional feature vector generated for the first colour image has been compared). Put more simply, a pair of feature vectors may be identified as corresponding to the same point in the scene, based on those two feature vectors being identified as closest in the feature space relative (to which each pixel, or group of pixels in each image has been mapped).

Step S505 may correspond to performing a cost volume calculation, wherein a matching cost volume is computed based on the differences in distances between at least some of the vectors generated for the at least two colour images. The difference in distance between respective vector pairs corresponds to the matching cost and provides an indication of how likely two vectors in a pair match at a given disparity. Pairs of vectors may be identified as corresponding to the same points in space based on the matching cost being minimised for that pair.

As will be appreciated, in some cases, it may be that two feature vectors generated for respective colour images are identified as being closest, but the distance between those two pairs may be greater than that which would be expected for corresponding image points. Hence, in some examples, it may be that a threshold imposed such that feature vector pairs are only identified as corresponding if, despite being identified as closest, the distance between the vectors in that pair is less than a threshold distance. This ensures that a 'least worst' match is not identified as corresponding to the same point in the scene for a given image pair.

In some examples, identifying corresponding parts of the scene in the representations generated by the trained machine learning model comprises generating an n-dimensional feature vector of each pixel in the at least two colour images. For each n-dimensional feature vector generated for the first input colour image, a distance between that n-dimensional feature vector and at least some of the n-dimensional feature vectors generated for the second colour image may be determined. Corresponding n-dimensional feature vector pairs may then be identified based on the distance between the vectors in a respective pair being less than the distance between other feature vectors with which at least one of the feature vectors in the pair was compared.

The searching of the feature space for corresponding feature vector pairs may be performed in any suitable manner. This may involve, for example, for a given feature vector generated for a first colour image, searching a region of the feature space that is close to that feature vector, for other corresponding feature vectors generated for the second colour image. It may be, for example, that a threshold distance is defined relative to the feature vector generated for the feature vector generated for the first colour image, and that a corresponding feature vector generated for the second colour image is searched for within the feature space, within the threshold distance. This may involve, for example, searching in one or more directions relative to the feature vector generated for the first colour image, within the threshold distance. Generally, it is expected that searching the entire feature space for corresponding feature vectors for the at least two images will be intensive in terms of the processing power required, and so it may be that the searching is at least limited in terms of the distance and / or direction, relative to a given feature vector generated for a given colour image.

As mentioned above, the method may include a third step, S503, of rectifying the at least two colour images, prior to being input to the machine learning model. The use of rectified images may simplify the step of searching for corresponding vector pairs in the feature space by ensuring that corresponding pairs need only be searched for in a particular direction, and relative to a particular location in the feature space. This searching step may involve calculating D vector subtractions along the scan line, for each pixel, and performing a distance calculation for each of those subtractions. Here, D corresponds to the number of disparities considered, i.e. the maximum disparity handled by the system. Typically, stereo-matching systems are typically built to handle only disparity up to a certain value, which determines the minimum depth that the system can output.

At a sixth step S506, depth information for the regions of the scene that are within the at least partially overlapping viewpoints of the at least two colour images, is obtained. The depth information is obtained based on the intrinsics and extrinsics associated with the at least two colour images and the corresponding features identified in the representations generated by the trained machine learning model.

In some examples, this may involve determining, for each matched feature vector pair, the corresponding pixel locations of the feature vectors in that pair in the corresponding colour images. Once the pixel locations are known for a given pair of feature vectors, the disparity may then be determined by calculating the difference in pixel positions for that pair of feature vectors. From the disparity, the corresponding depth (e.g. z-coordinates in the real-world) can be determined using e.g. equation 1 (or a suitably adapted version thereof).

The obtained depth information may be used to generate e.g. a point cloud of the part of the scene that is within the region of each image corresponding to the overlapping fields of view of the respective camera(s) that captured the images. From this point cloud, a three-dimensional mesh, such as a triangular or polygonal mesh may be generated from the point cloud. The mesh may be used to recreate the shape of one or more objects in the scene (or indeed the scene itself). The colour information associated with the at least two images may be used to generate a texture for applying to the generated mesh. The mesh with the texture applied may enable one or more objects in the scene to be viewed in three-dimensions at a display. That it is to say, a viewer may be able to change their perspective of the reconstructed objects, such that the objects are displayed from the viewer's new perspective (i.e. free-viewpoint).

As mentioned previously, stereo-matching is usually performed on grayscale images so as to reduce the variability between images as much as possible. This allows pixel intensities to be directly compared, and corresponding pixels or blocks of pixels to be identified based on a closeness in pixel intensity values. In conventional methods, performing this stereo-matching on colour or HDR images is less robust, as there is greater variability in the pixels of each image, and so it is less clear as to which pixels (or blocks of pixels) correspond with one another in an image pair. Moreover, mismatches in e.g. the exposure of the images in an image pair can reduce the accuracy with which the conventional stereo-matching can be performed.

The above described method is advantageous compared with conventional methods because the machine learning model is trained to be robust against the variability in colour image pairs, by learning features that are representative of the individual points (or regions) within the colour images. This is because each image in a given pair can be represented in terms of features, with the features being independent of the variability in e.g. exposure, colour, focal length, depth of field, etc. across the two images. This means that the variation between colour images in a stereo-image pair is no longer the main factor reducing the accuracy with which stereo-matching can be performed.

Instead, the main factor reducing the accuracy of the stereo-matching method is obtaining images of a scene that are representative enough of the real world to enable accurate triangulation to be performed. In other words, obtaining images that contain sufficient detail, i.e. as a result of more accurate measurement of the individual pixel values. For some image formats, such as standard dynamic range (SDR) images, it may be that e.g. two similar shades of red are represented with same pixel value. Whereas in another image format, such as an HDR image, the similar shades of red are represented with different pixel values. By obtaining images that accurately represent the scene, corresponding regions within a given image pair can be more clearly distinguished and identified as corresponding.

As will be appreciated, the capture of colour and / or HDR images may require more expensive or complex camera equipment compared with conventional methods. Moreover, the processing of the colour and / or HDR images may be more intensive compared with that which is required for conventional stereo-matching methods using grayscale images. However, it is generally expected that this increase in processing and potentially complexity of camera equipment is offset by the increase in robustness with which stereo-matching can be performed for the present method.

Training the machine learning model

In order to train the machine learning model, a plurality of training images are obtained. The training images may correspond to images of a real scene, i.e. captured with a real camera, or a virtual scene captured with a virtual camera. The training images may comprise colour images and / or HDR images (which themselves may be colour images), as described above. The use of HDR images in training the machine learning model may be desirable, as these will generally include greater detail and thus the machine learning model can learn to represent more of the scene in terms of features.

As mentioned previously, the training images vary in at least one aspect with respect to the other training images. The at least one aspect in which each training image varies relative to the other training images may include one or more of:

i. an exposure of the training image;

ii. an adjustment to at least one of the colour channels of the training image;

iii. the intrinsics of the camera associated with the training image;

iv. a filtering of the training image; and

v. a transformation of the training image.

The training images may be obtained by capturing a plurality of images of the same scene, with the same camera pose, but at different exposure values. That is, the shutter speed and / or aperture of the camera may be adjusted for each captured image. Alternatively, or in addition, a plurality of training images may be generated from a single image by artificially adjusting the brightness of the source image. For example, each training image may correspond to a different variation in the brightness of the pixels of the source image. As noted above, the training images need not necessarily correspond to the same images as those for which the machine learning model is to be used, once trained.

In some examples, the aspect in which each training image is varied may correspond to one or more colour channels of the training image. For example, at least some of the training images may correspond to images of the same scene, captured from the same pose, but with the pixel values in at least one of the colour channels being adjusted relative the pixel values in the same colour channel(s) in the other training images. This may involve, for example, generating a plurality of training images from a source training image wherein the source image corresponds to an initial capture of the scene. The pixels in the source image may take their default (i.e. original) values. A plurality of training images may then be generated from this source image by incrementing (and / or decrementing) the pixel values in one or more colour channels by different amounts. Each training image may therefore correspond to an image in which at least one of the colours has been perturbed by a different amount, relative to the original source image.

As will be appreciated, this perturbation in colour across the training images may also be achieved manually, e.g. by using an appropriate filter arranged in front of the lens of the camera, during the capture of the training images. The filter may attenuate specific wavelengths (or wavelength ranges) of light, such that the contribution of that light is lessened in the captured images. The attenuation strength of the filter may be adjustable, and / or a plurality of different filters may be used during the capture process, so as to control the extent to which different contributions of colour are attenuated in the captured images. In this way, a plurality of training images of the scene can be generated, with each training image including a different contribution of one or more colours.

In some examples, the colour of a training image may be perturbed by switching the pixel values for a given colour channel with the pixel values in a different colour channel. For example, if a given pixel in a training image has a pixel value of n_1 in the red channel, and a pixel value of n_2 in the green channel, then a second training image may be generated by switching the pixel values such that in the second training image, the same pixel takes the value of n_2 in the red channel and n_1 in the green colour channel. A plurality of training images may be generated in this way by switching the values of some or all of the pixels in different colour channels (e.g. red switched with green, red switched with blue, green switched with blue, etc.).

In some examples, at least some of the training images may be associated with different intrinsic parameters. For example, at least some of the training images may have been captured at different focal

lengths relative to one another (e.g. zoomed in or out by different amounts relative to a default focal length).

In further examples, at least some of the training images may have been captured with a different depth of field (i.e. aperture, but not necessarily at a different exposure).

In additional or alternative examples, the training images may correspond to images of the same scene, but to which different filtering operations have been applied. The filtering operation may include a blurring operation, such as e.g. Gaussian blurring. Each training image may correspond to a source image to which a blur of different size and / or shape and / or location has been applied. The application of the Gaussian blur to an image may simulate a change in depth of field of the image, with the blurred regions of the image corresponding to parts of the image that are out of focus and the sharper regions of the image corresponding to the part of the image that is in focus. It will be appreciated that blurring is just one example of a filtering operation and that respective training images may be generated by applying other filtering operations, by different amounts, to a source image.

By introducing variations in the training images for a given capture, captured from a respective pose, the machine learning model can be trained to generalise the scene more in terms of features. The more variation that is introduced into the training images, the more machine learning model can learn to generalise the corresponding scene (from the corresponding viewpoint). Generally, it is desirable for each of the images in a training set to correspond to the same viewpoint to ensure that at least a majority of the image points are corresponding across the images in the set. This ensures that the machine learning model can learn to represent each pixel (or groups of pixels), appearing in each image, in terms of features that are independent of the variations across the images in the training set.

Having obtained a plurality of training images, the training images are input to the machine learning model. A first set of training images may comprise images of the same scene, captured from the same pose, but varied in one or more the above-described manners. The machine learning model may comprise e.g. a convolutional neural network (CNN) that operates on each image in the set, and transforms each pixel of each image into a feature vector. The convolutional neural network may be equivalent to a feature extractor that extracts features from each of the pixels in the training images.

Each feature representation of each image in the set corresponds to a different domain. Domain adversarial training may then be used to enforce the constraint that all of the features, for all of the domains, have the same distribution.

The machine learning model may thus further comprise a domain adversarial network. The domain adversarial network may comprise a discriminator that looks at the features generated for each image and tries to discriminate between the domains (e.g. determine which variation the image corresponds to – a higher aperture domain, a lower exposure domain, a higher red domain, etc.) for any number of possible domains corresponding to alterations. The discriminator is included in the loss; that is, the discriminator is trained to be able to discriminate between the different domains, while the CNN feature extractor is penalised when the discriminator is able to accomplish its task successfully. As a

result of this, the CNN is trained learn representations of the scene that converge towards being domain-independent. In this way, the machine learning model is trained to learn a representation of the scene that is independent of the variations in the training images across the set.

As mentioned above, the machine learning model may be trained with multiple sets of training images, each set corresponding to images of the same scene or different ones and / or captured from different respective poses. That is, the training images may not necessarily be of the same scene for which the trained machine learning model is to be used in practice. Generally, the more scenes that the machine learning model has been trained with, the more types of scene the model will be applicable to. In some cases, it may be that a specific use of the machine learning model is envisaged, and so the machine learning model may be trained with images of corresponding scenes.

It will be appreciated that, although machine learning model is referred to in the singular, there may be at least two machine learning models involved; a first corresponding to the CNN and the second corresponding to the discriminator.

In some examples, a computer readable medium having computer executable instructions may be adapted to cause a computer system to perform any of the method steps described above in relation to Figure 5.

An example of a system for implementing the method described above in relation to Figure 5 is shown in Figure 6. In Figure 6, the system 600 is shown as comprising an input unit 602, a feature extraction unit 604, a feature mapping unit 606 and a depth analyser 608. In some examples, there may be an additional component, in the form of an image rectifier (not shown), that sits between the input unit and feature extraction unit, and which receives the at least two colour images and performs image rectification thereon. In such examples, the feature extraction unit is configured to receive at least two rectified colour images.

The input unit is operable to receive at least two colour images and the extrinsic and intrinsic camera parameters associated with the at least two colour images. As discussed above, the at least two colour images may comprise HDR images. The at least two images may be obtained in any of the previously described manners. The at least two images correspond to at least partially overlapping viewpoints, such that at least some of the pixels in each image correspond to the same point in space (i.e. the same part of the scene). The intrinsics may correspond to the focal length of the or each camera that captured the at least two colour images. The extrinsics may correspond to the position and / or orientation of the or each camera that captured the at least two colour images.

The feature extraction unit is configured to receive the at least two colour images and generate respective feature representations of the at least two colour images. The feature extraction unit comprises a machine learning model trained to generate feature representations of input colour images, the machine learning model being trained with multiple images of the same scene, captured from the same viewpoint, with each image being varied in at least one aspect with respect to the other training images. The machine

learning model is trained to learn a representation of the scene that is independent of the variations in the training images. The machine learning model may be trained with multiple sets of training images, each set corresponding to a different scene and / or the same scene but captured from a different respective pose.

In some examples, the feature extraction unit may correspond to a convolutional neural network (CNN) that is trained in conjunction with a domain adversarial network, so as to learn a representation of the scene that is independent of the variations in the training images. The feature extraction unit may be configured to transform at least some of the pixels in each image to corresponding n-dimensional feature vectors.

The machine learning model may be trained and operate in any of manners described previously in relation to Figure 5. As previously discussed, this may involve generating, for each pixel in each of the at least two colour images, a corresponding feature representation of that pixel. The feature representation may correspond to a n-dimensional feature vector, which can be represented in feature space.

Although the feature extraction unit is shown as being separate from the input unit, in some examples, these units may form part of the same overall input module.

The feature matching unit is configured to receive the representations generated by the feature extraction unit and to identify corresponding features in the feature representations of the at least two colour images. The feature matching unit may be configured to determine distances between respective pairs of n-dimensional feature vectors, wherein each vector in a pair corresponds to a different one of the at least two images. A pair of feature vectors may be identified as corresponding to the same point in space when the distance between those feature vectors (in feature space) is less than the distance between other vectors with which one of the vectors in that pair has been compared.

In some examples, the feature matching unit is operable to determine a distance between at least some of the n-dimensional feature vectors generated for a first input colour image with at least some of the n-dimensional feature vectors generated for a second colour image. The feature matching unit may be configured to determine for at least some (if not each) of the feature vectors generated for the first colour image, respective n-dimensional feature vectors generated for the second colour image that are located closest (in feature space) to the feature vectors generated for the first colour image. The feature vectors in a respective pair that are determined as being closest may be identified as corresponding to the same point in the scene.

As described previously, a threshold distance may be imposed to ensure that a feature vector generated for one colour image is not matched to a feature vector in another image, due to those feature vectors being deemed the ‘least worst’ match.

Generally, the feature matching unit is configured to perform the matching step described in relation to Figure 5 (see step S505). The matching may be performed on per-vector basis, or groups of vector basis.

The depth analyser is configured to obtain depth information for parts of the scene that are within the partially overlapping fields of view of the at least two colour images, based on the corresponding features identified by the feature matching unit and the extrinsics and intrinsics associated with the at least two colour images from which the feature representations were generated. The extrinsics and intrinsics associated with the at least two images may be provided to the depth analyser as an input from the input unit. The vector pairs identified as corresponding may be provided as input from the feature-matching unit (e.g. in the form of a disparity map).

The depth analyser may be configured to determine depth information from the matched vector pairs based on a known relationship between the intrinsics and extrinsics associated with the at least two colour images and the identified corresponding pixels (or groups of pixels). As described previously, this may involve determining a disparity for the colour pixels corresponding to the matched vectors in a respective vector pair, and using the baseline and focal length associated with the or each camera to determine a z-coordinate of the point in the scene that corresponds with the matched pixels. This may be repeated for each matched vector pair, such that the colour pixels corresponding the vectors in that pair are mapped to a corresponding z-coordinate.

The depth information obtained by the depth analyser may be output by the depth analyser. This depth information may be used, for example, in combination with the colour information in the at least two colour images to reconstruct the part of the scene that is within the partially overlapping viewpoints of the at least two images. The system may further comprise, e.g. a point cloud generator that is configured to receive the at least two colour images and the obtained depth information and to generate a point cloud from this depth information. The depth information may be used in any suitable process for reconstructing part of the scene, such that the scene can be viewed in a free viewpoint manner.

It will be appreciated that example embodiments can be implemented by computer software operating on a general purpose computing system such as a games machine. In these examples, computer software, which when executed by a computer, causes the computer to carry out any of the methods discussed above is considered as an embodiment of the present disclosure. Similarly, embodiments of the disclosure are provided by a non-transitory, machine-readable storage medium which stores such computer software.

It will also be apparent that numerous modifications and variations of the present disclosure are possible in light of the above teachings. It is therefore to be understood that within the scope of the appended claims, the disclosure may be practised otherwise than as specifically described herein.

It will be further appreciated that while a CNN and domain adversarial network have been described in relation to the training and operation of the machine learning model, any suitably trained machine learning model may be used in accordance with the present disclosure.

CLAIMS

1. A method of obtaining depth information of a scene, the method comprising:

obtaining at least two colour images of a scene, each image corresponding to a different viewpoint of the scene, the viewpoints overlapping at least partially;

obtaining the camera intrinsics and extrinsics associated with the at least two colour images;

inputting each colour image to a trained machine learning model, the machine learning model being trained to generate a representation of each input colour image;

wherein the machine learning model is trained with a plurality of images of a scene captured from the same respective viewpoint, each image being varied in at least one aspect with respect to the other training images, the machine learning model being trained to learn a representation of the scene that is independent of the variations in the plurality of training images;

identifying, based on a similarity between the features of the representations generated by the trained machine-learning model, corresponding regions of the scene in the representations generated by the trained machine learning model; and

obtaining depth information for the regions of the scene that are within the at least partially overlapping viewpoints of each image, the depth information being obtained based on the intrinsics and extrinsics associated with the at least two colour images and the corresponding features identified in the representations generated by the trained machine learning model.

2. A method according to claim 1, wherein the machine learning model is trained to map at least some of the pixels of the input colour images to respective n-dimensional feature vectors;

wherein identifying the corresponding regions of the scene in the representations generated by the trained machine learning model comprises:

determining a distance between at least some of the n-dimensional feature vectors generated for a first input colour image with at least some of the n-dimensional feature vectors generated for a second input colour image;

determining, for at least some of the n-dimensional feature vectors generated for the first input colour image, respective n-dimensional feature vectors generated for the second colour image that are closest to the n-dimensional feature vectors generated for the first input colour image, the feature vectors determined as being closest corresponding to the same region of the scene in the at least two colour images.

3. A method according to any preceding claim, wherein the colour images comprise high dynamic range images, each dynamic range image comprising at least two images of the scene captured at different exposures.

4. A method according to any preceding claim, wherein the at least one aspect in which each training image of the scene is varied relative to the other training images comprises at least one of:

i. an exposure of the training image;

ii. an adjustment to at least one of the colour channels of the training image;

iii. the intrinsics of the camera that captured the training image relative to the intrinsics of the camera that captured the other training images; and

iv. a filtering of the training image.

5. A method according to any preceding claim, wherein the machine learning model comprises a domain adversarial neural network.

6. A method according to any preceding claim, comprising rectifying the at least two colour images prior to inputting the colour images to the trained machine learning model.

7. A computer readable medium having computer executable instructions adapted to cause a computer system to perform the method of any of claims 1 to 6.

8. A system for generating depth information of a scene, the system comprising:

an input unit operable to receive at least two colour images and the extrinsic and intrinsic camera parameters associated with the at least two colour images;

a feature extraction unit configured to receive the at least two colour images and generate respective feature representations of the at least two colour images;

wherein the feature extraction unit comprises a machine learning model trained to generate feature representations of input colour images, the machine learning model being trained with multiple images of a scene captured from the same respective viewpoint, each image being varied in at least one aspect with respect to the other training images;

the machine learning model being trained to learn a representation of the scene that is independent of the variations in the training images;

a feature matching unit operable to receive the representations generated by the feature extraction unit and to identify corresponding features in the feature representations of the at least two colour images;

a depth analyser configured to obtain depth information for parts of the scene that are within the field of view of the at least two colour images, based on the corresponding features identified by the feature matching unit.

9. A system according to claim 8, wherein the input unit is operable to obtain at least two colour high dynamic range images, the feature extraction unit being configured to generate respective feature representations of the at least two colour high dynamic range images.

10. A system according to claim 8 or claim 9, wherein the feature extraction unit comprises a neural network trained to generate feature representations of input colour images, the neural network being trained via domain adversarial training.

11. A system according to any of claims 8 to 10, wherein the machine learning model is trained with training images of a scene, the at least one aspect in which each training image is varied with respect to the other training images comprising at least one of:

i. an exposure of the training image;

ii. an adjustment to at least one of the colour channels of the training image;

iii. the intrinsics of the camera that captured the training image relative to the intrinsics of the camera that captured the other training images; and

iv. a filtering of the training image.

12. A system according to any of claims 8 to 10, wherein the feature extraction unit is configured to generate n-dimensional feature vectors of at least some or each of the pixels in the at least two colour images, each feature vector being representable in feature space.

13. A system according to claim 12, wherein the feature matching unit is configured to determine distances between respective pairs of vectors in the feature space, the vectors in each pair being generated for a different one of the at least two colour images; and

wherein the feature matching unit is configured to identify corresponding features in the feature representations of the at least two colour images by identifying, for at least some of the vectors generated for a first colour image, respective vectors generated for the second colour image that are closest to the vectors generated for the first colour image.

14. A system according to any of claims 8 to 13, wherein the depth analyser is configured to determine a disparity associated with each of the pixels corresponding to the matched feature representations; and

wherein the depth analyser is configured to obtain the depth information based on the disparity determined for the matched feature representations and the intrinsics and extrinsics associated with the at least two corresponding colour images.

15. A system according to any of claims 8 to 13, comprising an image rectifier operable to receive the at least two colour images and rectify the at least two colour images; and

wherein the feature extraction unit is operable to generate feature representations of the at least two rectified images.

Intellectual Property Office

Application No: GB1909444.0 **Examiner:** Ralph Cannon

Claims searched: 1-15 **Date of search:** 24 December 2019

Patents Act 1977: Search Report under Section 17

Documents considered to be relevant:

Category	Relevant to claims	Identity of document and passage or figure of particular relevance
X	1-15	CN 109584290 A (BEIHANG UNIVERSITY) in particular see WPI abstract, accession number 2019-33270X
X	1-12, 14 and 15	CN 106600583 A (XIDIAN UNIVERSITY) in particular see WPI abstract, accession number 2017-28096P and figs. 2 and 4
X	1-7, 10-13 and 15	Jure Zbontar, Yann LeCun, Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches, Journal of Machine Learning Research, 2016, 17(65), 1-32 In particular see abstract
X	1-8, 10-12, 14 and 15	CN 104574391 A (XI AN JIAOTONG UNIVERSITY) in particular see EPODOC abstract
X	1, 3, 4, 7-11 and 14	Nikolaue Mayer et al, A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation, 2016, IEEE Conference on Computer Vision and Pattern Recognition, pages 4040-4048 see whole document

Categories:

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

Field of Search:

Search of GB, EP, WO & US patent documents classified in the following areas of the UKC^{X} :

Worldwide search of patent documents classified in the following areas of the IPC

G06N; G06T; H04N

The following online and other databases have been used in the preparation of this search report

WPI, EPODOC, Patent Fulltext

Intellectual Property Office

International Classification:

Subclass	Subgroup	Valid From
G06T	0007/593	01/01/2017
G06T	0007/73	01/01/2017
H04N	0013/271	01/01/2018