

## Method and System for Determining a Trajectory of a Target Object

### Background and Field

5

The invention relates to a method and system for determining a trajectory of a target object, in particular, but not exclusively, for use in an autonomous vehicle (AV).

Autonomous vehicles have attracted considerable attention due to the desire to  
10 reduce fatal accidents. Significant progress in hardware has made it possible to develop self-drive vehicles which can respond to their environment in real-time. One basic function for AV is vehicle following, i.e. following the vehicles ahead safely, in particular for preventing rear-end collisions. Object detection is employed in order to help a vehicle understand the behaviour and intentions of the objects ahead of it.  
15 Various sensors, such as cameras, LIDAR, radar, or the fusion of these modalities, can be used to complete the detection task.

Although recent progress has been made in the detection of objects using deep learning methods, relatively little attention has been given to object detection under  
20 bad weather conditions.

In general, autonomous vehicles perform object detection using an in-car camera. However, image quality may be rather poor under heavy rain conditions. Information regarding a particular object of interest, for example a vehicle's shape or colour, may  
25 be lost due to rain effects as well as rain streaks or raindrops that remain on the windscreen. Approaches for tackling this issue using conventional image processing have tended to focus on reducing the rain effects from the images, i.e. employing de-raining methods. However, de-raining methods are limited, particularly under heavy rain conditions, because the underlying image models representing rain streaks or  
30 raindrops may be very different from the images captured by the in-car camera or the

degradation caused by rain may be too significant to make the objects clear enough to be detected. Furthermore, existing de-rain processes are computationally intensive, and infeasible for online implementation in real time, as needed for use in AV applications.

5

As radar is relatively insensitive to rain, another way to address objection detection in rainy conditions has been to rely on the use of radar as the main detection sensor.

However, under heavy rain conditions, problems due to false positives caused by reflection from rain droplets may be a significant issue. In addition, the object

10 information obtained from a radar sensor is limited as the point cloud returned from a radar is very noisy. Further, only a single point is provided for each object using a commercial radar.

It is desirable to provide a method and system for determining a trajectory of a target

15 object which addresses at least one of the drawbacks of the prior art and/or to provide the public with a useful choice.

### Summary

20 In a first aspect, there is provided a method of determining a trajectory of a target object, comprising: receiving a plurality of source images of the target object captured by an image capturing device; generating corresponding augmented images from the plurality of source images using a data augmentation model trained using a plurality of training images, each augmented image including an augmented target object label  
25 associated with the respective target object in the plurality of source images; and tracking movement of the augmented target object label in the corresponding augmented images using an object detector instead of tracking movement of the target object, in order to determine the trajectory of the target object.

By generating augmented target images from a model trained using source images with the augmented target images having augmented target object labels representing the target object in the source images and tracking the movement of the augmented target object label rather than tracking the movement of the object itself, the object  
5 detection accuracy may be improved, particularly when the images are captured in conditions of reduced visibility, such as heavy rain.

The method may further comprise generating a plurality of binary saliency region images from the plurality of source images by discarding pixels of the plurality of  
10 source images having a saliency level below a threshold, and the data augmentation model may generate the corresponding augmented images from both the plurality of source images and the plurality of binary saliency region images. By employing saliency data in the generation of the augmented images, positioning of the augmented target object label as well as preservation of the background features of the source image  
15 may be improved. Generating the plurality of binary saliency region images may or may not comprise generating a saliency map corresponding to the source image.

The augmented target object label may comprise a bounding box around the target object. The bounding box may comprise a colour indicating a category to which the  
20 target object belongs. The bounding box may comprise at least one contour and tracking movement of the augmented target object label may comprise detecting the at least one contour. Bounding boxes may provide labels which are easy to detect without compromising the visibility of much of the background of the image. The bounding box may comprise a plurality of contours including an external contour, and  
25 tracking movement of the augmented target object label may comprise detecting the external contour of the bounding box; detecting all of the plurality of contours in the augmented image to obtain an all-contours detection result; and subtracting the external contour from the all-contours detection result. This process may enable merged bounding boxes to be distinguished, thereby ensuring that as much individual  
30 object data is captured as possible and that objects are accurately differentiated.

The method of determining the trajectory of the target object may further include training the data augmentation model to generate the corresponding augmented images directly from the plurality of source images, i.e. the source images may  
5 comprise a direct input into the data augmentation model and the augmented images may comprise a direct output from the data augmentation model. Training the data augmentation model may comprise receiving the plurality of training images; receiving a plurality of annotated images of the target object, each annotated image including an annotated target object label associated with the respective target object suitable  
10 for movement tracking by the object detector instead of the target object; generating corresponding augmented training images from the plurality of training images using the data augmentation model; and updating the data augmentation model to minimize a discriminability between the augmented training images and the plurality of annotated images.

15

The discriminability between the augmented training images and the annotated images may be determined by inputting the augmented training images and the plurality of annotated images into a discriminator model, and the discriminator model may be trained concurrently with the data augmentation model to determine if an  
20 input image to the discriminator model is one of the augmented training images or one of the plurality of annotated images. Employing a discriminator model may ensure high accuracy of the augmented images and ensure that the image background is well preserved. The method of training may employ a Generative Adversarial Networks (GAN) approach, for example a cycle-GAN or a conditional-GAN approach. Training the  
25 data augmentation model may further comprise generating the plurality of annotated images from the plurality of training images by annotating the training images (e.g. conditional-GAN). Alternatively, it may comprise annotating images not belonging to the plurality of training images (e.g. cycle-GAN).

Training the data augmentation model may further comprise generating a plurality of binary saliency region training images from the plurality of training images by discarding pixels of the plurality of training images having a saliency level below a threshold; generating a plurality of augmented binary saliency region images from the augmented training images by discarding pixels of the augmented training images having the saliency level below the threshold, and updating the data augmentation model to minimize a background preserving loss quantifying a dissimilarity between the plurality of augmented binary saliency region images and the plurality of binary saliency region training images. Generating the binary saliency region training images and the augmented binary saliency region images may or may not comprise generating saliency maps corresponding to both images. Training the data augmentation model to minimize a background preserving loss may improve the accuracy of the data augmentation model in labelling the target object and may ensure that the background of the target object is better preserved in the augmented images generated by the model.

The training images may be selected based on a visibility of the target object in each of the plurality of training images. The images may be selected with a preference for poor visibility of the target object in the image, for example where the target object is obscured by heavy rain, snow or fog. This may ensure improved performance and accuracy of the data augmentation model when operating on source images captured under conditions of poor visibility.

In a second aspect, there is provided a method for training a data augmentation model for use in determining a trajectory of a target object, the method comprising: receiving a plurality of training images of the target object captured by an image capturing device; receiving a plurality of annotated images, each annotated image including an annotated target object label associated with the respective target object suitable for movement tracking by an object detector instead of the target object; generating corresponding augmented images from the plurality of training images using the data

augmentation model; and training the data augmentation model to minimize a discriminability between the augmented images and the annotated images and thereby generate augmented images including an augmented target object label associated with the respective target object suitable for movement tracking by the object detector instead of the target object in order to determine a trajectory of the target object.

In a third aspect, there is provided a system for determining a trajectory of a target object. The system may comprise: an image capturing device operable to capture a plurality of source images of the target object; a processor configured to perform a method of determining a trajectory of a target object and an output configured to output information regarding the trajectory of the target object. The method of determining a trajectory of the target object performed by the processor may comprise receiving a plurality of source images of the target object captured by the image capturing device; generating corresponding augmented images from the plurality of source images using a data augmentation model trained using a plurality of training images, each augmented image including an augmented target object label associated with the respective target object in the plurality of source images; and tracking movement of the augmented target object label in the corresponding augmented images using an object detector instead of tracking movement of the target object, in order to determine the trajectory of the target object.

In a fourth aspect, there is provided a vehicle comprising a system for determining a trajectory of a target object comprising an image capturing device operable to capture a plurality of source images of the target object; a processor configured to perform a method of determining a trajectory of a target object comprising: receiving a plurality of source images of the target object captured by the image capturing device, generating corresponding augmented images from the plurality of source images using a data augmentation model trained using a plurality of training images, each augmented image including an augmented target object label associated with the

respective target object in the plurality of source images, and tracking movement of the augmented target object label in the corresponding augmented images using an object detector instead of tracking movement of the target object, in order to determine the trajectory of the target object, and an output configured to output  
5 information regarding the trajectory of the target object; and a controller configured to control an operation of the vehicle based on the information regarding the trajectory of the target object. The vehicle may be an autonomous vehicle (AV).

In a fifth aspect, there is provided a system for training a data augmentation model for  
10 use in a method of determining a trajectory of a target object, comprising: an image capturing device configured to capture a plurality of training images of the target object; a processor configured to perform a method of training a data augmentation model comprising: receiving a plurality of training images of the target object captured by the image capturing device; receiving a plurality of annotated images, each  
15 annotated image including an annotated target object label associated with the respective target object suitable for movement tracking by an object detector instead of the target object; generating corresponding augmented images from the plurality of training images using the data augmentation model; and training the data augmentation model to minimize a discriminability between the augmented images  
20 and the annotated images and thereby generate augmented images including an augmented target object label associated with the respective target object suitable for movement tracking by the object detector instead of the target object in order to determine a trajectory of the target object; and an output configured to output the trained data augmentation model.

25

In a sixth aspect, there is provided a tangible or intangible computer readable medium configured to cause a processor to perform a method of determining a trajectory of a target object, the method comprising: receiving a plurality of source images of the target object captured by an image capturing device; generating corresponding  
30 augmented images from the plurality of source images using a data augmentation

model trained using a plurality of training images, each augmented image including an augmented target object label associated with the respective target object in the plurality of source images; and tracking movement of the augmented target object label in the corresponding augmented images using an object detector instead of tracking movement of the target object, in order to determine the trajectory of the target object.

In a seventh aspect, there is provided a tangible or intangible computer readable medium configured to cause a processor to perform a method of training a data augmentation model, the method comprising: receiving a plurality of training images of the target object captured by an image capturing device; receiving a plurality of annotated images, each annotated image including an annotated target object label associated with the respective target object suitable for movement tracking by an object detector instead of the target object; generating corresponding augmented images from the plurality of training images using the data augmentation model; and training the data augmentation model to minimize a discriminability between the augmented images and the annotated images and thereby generate augmented images including an augmented target object label associated with the respective target object suitable for movement tracking by the object detector instead of the target object in order to determine a trajectory of the target object.

In an eighth aspect, there is provided a method of object detection in an image or a sequence of images, wherein the object in the image is captured under poor visibility, and the object trajectory is tracked, the method comprising receiving a set of images from a live source of objects captured under poor or degraded visibility; translating the image from the set of images to a synthetic image of objects with bounding boxes based on a trained data augmentation model; detecting the inner and outer boundaries of the bounding boxes in the synthetic image by colour of the bounding boxes; and grouping the bounding boxes to represent the objects to be detected; performing a temporal trajectory analysis on the group of bounding boxes over the

sequence of images to categorise the bounding boxes as stable trajectory objects, or temporary trajectory objects; wherein the trained data augmentation model is generated by a generative adversarial network (cycle GAN) with an annotated data set of images of objects with bounding boxes of one or more predefined colour that  
5 represent objects for detection; and synthesise or generate a dataset of training images of objects captured in poor visibility, and discriminating the synthesized images to label the bounding boxes representing the objects.

It is envisaged that features relating to one aspect may be applicable to the other  
10 aspects.

#### Brief Description of the Drawings

An exemplary embodiment will now be described with reference to the accompanying  
15 drawings, in which:

Fig. 1 is a simplified block diagram of an autonomous vehicle (AV) according to a preferred embodiment;

Fig. 2 illustrates an object detection system of the AV of Fig. 1;

20 Fig. 3 illustrates a method performed by a processor of the object detection system of Fig. 2;

Figs. 4a, 4b and 4c show an example of a raw image, a corresponding saliency map and a corresponding binary saliency region image, respectively determined in step 203 of the method of Fig. 3;

25 Figs. 5a, 5b and 5c show an example of a raw image, a corresponding saliency map and a corresponding binary saliency region image, respectively determined in step 203 of the method of Fig. 3;

Figs. 6a, 6b and 6c show an example of a raw image, a corresponding saliency map and a corresponding binary saliency region image, respectively determined in step 203 of  
30 the method of Fig. 3;

Fig. 7 illustrates a neural network architecture employed in step 205 of the method of Fig. 3;

Fig. 8a and 8b illustrate an example of a raw and translated image determined in step 207 of the method of Fig. 3;

- 5 Fig. 9a, 9b, and 9c illustrate examples bounding box detection results based on all boundaries of three bounding boxes, external boundaries of the same three bounding boxes, and the external boundaries subtracted from all of the boundaries, respectively determined in accordance with step 209 of the method of Fig. 3;

Fig. 10 illustrates a method of determining vehicle trajectory performed in step 213 of  
10 the method of Fig. 3;

Fig. 11 illustrates a method of training the data augmentation model employed in step 205 of the method of Fig. 3;

Fig. 12 illustrates a method of optimizing the data augmentation model performed in step 1007 of the method of Fig. 11;

- 15 Fig. 13a, 13b, 13c, and 13d illustrate examples of training images for use in steps 1001, 1003 and 1005 of the method of Fig. 11;

Fig. 14a, 14b, 14c, and 14d illustrate examples of training images for use in steps 1001, 1003 and 1005 of the method of Fig. 11;

- Fig. 15a, 15b and 15c illustrate image augmentation results obtained following step  
20 207 of the method of Fig. 3;

Fig. 16a and 16b illustrate image augmentation results following step 207 of the method of Fig. 3;

Fig. 17a and 17b illustrate image augmentation results following step 207 of the method of Fig. 3;

- 25 Fig. 18a and 18b illustrate image augmentation results following step 207 of the method of Fig. 3; and

Fig. 19a and 19b illustrate image augmentation results following step 207 of the method of Fig. 3.

30 Detailed Description of Preferred Embodiment

The described embodiment relates to an object detection system in an autonomous vehicle (AV). As will be described in detail below, the system is configured to perform a method of determining trajectory of other moving vehicles in the surrounding environment of the AV. Initially, the method comprises receiving a plurality of source images of the vehicles captured by a camera. Subsequently, corresponding augmented images from the plurality of source images are generated using a data augmentation model trained using a plurality of training images, each augmented image including an augmented target object label associated with the respective vehicle in the plurality of source images. Finally, the movement of the augmented target object label in the corresponding augmented images is tracked using an object detector instead of tracking movement of the vehicles themselves, in order to determine the trajectory of the vehicles.

Fig. 1 is a simplified block diagram of the AV 100, for example a car, according to the described embodiment. The AV 100 includes an image capturing device in the form of a camera 101, which is a GigE camera (Gigabit Ethernet camera) according to the preferred embodiment. Examples include a Point Grey BFLY-PGE-20E4 camera, 2-megapixel sensor that supports frame rates up to 47 FPS. The AV 100 further includes a system for detecting a trajectory of a target object, for example another vehicle, in the form of object detection system 380, and a controller 105. The camera 101 is configured to capture images of the surrounding environment of the AV 100 for processing by the object detection system 380. The controller 105 is configured to control an operation of the AV 100 based at least in part on information received from the object detection system 380, for example in such a way as to avoid collision with other objects and/or vehicles in its environment.

As such, the controller 105 may itself comprise further computing devices. The controller 105 may comprise several sub-systems for controlling specific aspects of the movement of the AV 100 including but not limited to a deceleration system, an

acceleration system and a steering system. Certain of these sub-systems may comprise one or more actuators, for example the deceleration system may comprise brakes, the acceleration system may comprise an accelerator pedal, and the steering system may comprise a steering wheel or other actuator to control the angle of turn of wheels of  
5 the AV 100, etc.

Although the object detection system 380 is shown as a separate module in Fig. 1, it is envisaged that it may form part of the controller 105.

10 Fig. 2 illustrates the object detection system 380. The object detection system 380 includes a processor 382 (which may be referred to as a central processor unit or CPU) that is in communication with memory devices including secondary storage 384, read only memory (ROM) 386, random access memory (RAM) 388, input/output (I/O) devices 390, network connectivity devices 392 and a graphics processing unit (GPU)  
15 394, for example a mini GPU. The processor 382 and/or GPU 394 may be implemented as one or more CPU chips. The GPU 394 may be embedded alongside the processor 382 or it may be a discrete unit, as shown in Fig. 2.

It is understood that by programming and/or loading executable instructions onto the  
20 object detection system 380, at least one of the CPU 382, the RAM 388, the ROM 386 and the GPU 394 are changed, transforming the object detection system 380 in part into a particular machine or apparatus having the novel functionality taught by the present disclosure. It is fundamental to the electrical engineering and software engineering arts that functionality that can be implemented by loading executable  
25 software into a computer can be converted to a hardware implementation by well-known design rules. Decisions between implementing a concept in software versus hardware typically hinge on considerations of stability of the design and numbers of units to be produced rather than any issues involved in translating from the software domain to the hardware domain. Generally, a design that is still subject to frequent  
30 change may be preferred to be implemented in software, because re-spinning a

hardware implementation is more expensive than re-spinning a software design. Generally, a design that is stable that will be produced in large volume may be preferred to be implemented in hardware, for example in an application specific integrated circuit (ASIC), because for large production runs the hardware

5 implementation may be less expensive than the software implementation. Often a design may be developed and tested in a software form and later transformed, by well-known design rules, to an equivalent hardware implementation in an application specific integrated circuit that hardwires the instructions of the software. In the same manner as a machine controlled by a new ASIC is a particular machine or apparatus,

10 likewise a computer that has been programmed and/or loaded with executable instructions may be viewed as a particular machine or apparatus.

Additionally, after the system 380 is turned on or booted, the CPU 382 and/or GPU 394 may execute a computer program or application. For example, the CPU 382 and/or

15 GPU 394 may execute software or firmware stored in the ROM 386 or stored in the RAM 388. In some cases, on boot and/or when the application is initiated, the CPU 382 and/or GPU 394 may copy the application or portions of the application from the secondary storage 384 to the RAM 388 or to memory space within the CPU 382 and/or GPU 394 itself, and the CPU 382 and/or GPU 394 may then execute instructions that

20 the application is comprised of. In some cases, the CPU 382 and/or GPU 394 may copy the application or portions of the application from memory accessed via the network connectivity devices 392 or via the I/O devices 390 to the RAM 388 or to memory space within the CPU 382 and/or GPU 394, and the CPU 382 and/or GPU 394 may then execute instructions that the application is comprised of. During execution, an

25 application may load instructions into the CPU 382 and/or GPU 394, for example load some of the instructions of the application into a cache of the CPU 382 and/or GPU 394. In some contexts, an application that is executed may be said to configure the CPU 382 and/or GPU 394 to do something, e.g., to configure the CPU 382 and/or GPU 394 to perform the object detection according to the described embodiment. When

the CPU 382 and/or GPU 394 is configured in this way by the application, the CPU 382 and/or GPU 394 becomes a specific purpose computer or a specific purpose machine.

The secondary storage 384 is typically comprised of one or more disk drives or tape  
5 drives and is used for non-volatile storage of data and as an over-flow data storage device if RAM 388 is not large enough to hold all working data. Secondary storage 384 may be used to store programs which are loaded into RAM 388 when such programs are selected for execution. The ROM 386 is used to store instructions and perhaps data which are read during program execution. ROM 386 is a non-volatile memory  
10 device which typically has a small memory capacity relative to the larger memory capacity of secondary storage 384. The RAM 388 is used to store volatile data and perhaps to store instructions. Access to both ROM 386 and RAM 388 is typically faster than to secondary storage 384. The secondary storage 384, the RAM 388, and/or the ROM 386 may be referred to in some contexts as computer readable storage media  
15 and/or non-transitory computer readable media.

I/O devices 390 may include a wireless or wired connection to the camera 101 for receiving image data from the camera 101 and/or a wireless or wired connection to the controller 105 for transmitting information regarding the trajectory of a target  
20 object so that the controller 105 can control the operation of the AV 100 accordingly. The I/O devices 390 may alternatively or additionally include electronic displays such as video monitors, liquid crystal displays (LCDs), plasma displays, touch screen displays, or other well-known output devices.

25 The network connectivity devices 392 may enable a wireless connection to facilitate communication with other computing devices such as components of the AV 100, for example the camera 101 and/or controller 105 or with other computing devices not part of the AV 100. The network connectivity devices 392 may take the form of modems, modem banks, Ethernet cards, universal serial bus (USB) interface cards, serial interfaces, token ring cards, fibre distributed data interface (FDDI) cards, wireless  
30

local area network (WLAN) cards, radio transceiver cards that promote radio communications using protocols such as code division multiple access (CDMA), global system for mobile communications (GSM), long-term evolution (LTE), worldwide interoperability for microwave access (WiMAX), near field communications (NFC), radio frequency identity (RFID), and/or other air interface protocol radio transceiver cards, and other well-known network devices. These network connectivity devices 392 may enable the processor 382 and/or GPU 394 to communicate with the Internet or one or more intranets. With such a network connection, it is contemplated that the processor 382 and/or GPU 394 might receive information from the network, or might output information to the network in the course of performing an object detection method according to the described embodiment. Such information, which is often represented as a sequence of instructions to be executed using processor 382 and/or GPU 394, may be received from and outputted to the network, for example, in the form of a computer data signal embodied in a carrier wave.

15

Such information, which may include data or instructions to be executed using processor 382 and/or GPU 394 for example, may be received from and outputted to the network, for example, in the form of a computer data baseband signal or signal embodied in a carrier wave. The baseband signal or signal embedded in the carrier wave, or other types of signals currently used or hereafter developed, may be generated according to several methods well-known to one skilled in the art. The baseband signal and/or signal embedded in the carrier wave may be referred to in some contexts as a transitory signal.

25 The processor 382 and/or GPU 394 executes instructions, codes, computer programs, scripts which it accesses from hard disk, floppy disk, optical disk (these various disk-based systems may all be considered secondary storage 384), flash drive, ROM 386, RAM 388, or the network connectivity devices 392. While only one processor 382 and GPU 394 are shown, multiple processors may be present. Thus, while instructions may be discussed as executed by a processor, the instructions may be executed

30

simultaneously, serially, or otherwise executed by one or multiple processors. Instructions, codes, computer programs, scripts, and/or data that may be accessed from the secondary storage 384, for example, hard drives, floppy disks, optical disks, and/or other device, the ROM 386, and/or the RAM 388 may be referred to in some  
5 contexts as non-transitory instructions and/or non-transitory information.

In an embodiment, the object detection system 380 may comprise two or more computers in communication with each other that collaborate to perform a task. For example, but not by way of limitation, an application may be partitioned in such a way  
10 as to permit concurrent and/or parallel processing of the instructions of the application. Alternatively, the data processed by the application may be partitioned in such a way as to permit concurrent and/or parallel processing of different portions of a data set by the two or more computers. In an embodiment, virtualization software may be employed by the object detection system 380 to provide the functionality of a  
15 number of servers that is not directly bound to the number of computers in the object detection system 380. For example, virtualization software may provide twenty virtual servers on four physical computers. In an embodiment, the functionality according to the described embodiment may be provided by executing the application and/or applications in a cloud computing environment. Cloud computing may comprise  
20 providing computing services via a network connection using dynamically scalable computing resources. Cloud computing may be supported, at least in part, by virtualization software. A cloud computing environment may be established by an enterprise and/or may be hired on an as-needed basis from a third-party provider. Some cloud computing environments may comprise cloud computing resources owned  
25 and operated by the enterprise as well as cloud computing resources hired and/or leased from a third-party provider.

In an embodiment, some or all of the functionality of the described embodiment may be provided as a computer program product. The computer program product may  
30 comprise one or more computer readable storage medium having computer usable

program code embodied therein to implement the functionality according to the described embodiment. The computer program product may comprise data structures, executable instructions, and other computer usable program code. The computer program product may be embodied in removable computer storage media and/or non-removable computer storage media. The removable computer readable storage medium may comprise, without limitation, a paper tape, a magnetic tape, magnetic disk, an optical disk, a solid-state memory chip, for example analogue magnetic tape, compact disk read only memory (CD-ROM) disks, floppy disks, jump drives, digital cards, multimedia cards, and others. The computer program product may be suitable for loading, by the object detection system 380, at least portions of the contents of the computer program product to the secondary storage 384, to the ROM 386, to the RAM 388, and/or to other non-volatile memory and volatile memory of the object detection system 380. The processor 382 and/or GPU 394 may process the executable instructions and/or data structures in part by directly accessing the computer program product, for example by reading from a CD-ROM disk inserted into a disk drive peripheral of the object detection system 380. Alternatively, the processor 382 and/or GPU 394 may process the executable instructions and/or data structures by remotely accessing the computer program product, for example by downloading the executable instructions and/or data structures from a remote server through the network connectivity devices 392. The computer program product may comprise instructions that promote the loading and/or copying of data, data structures, files, and/or executable instructions to the secondary storage 384, to the ROM 386, to the RAM 388, and/or to other non-volatile memory and volatile memory of the object detection system 380.

25

In some contexts, the secondary storage 384, the ROM 386, and the RAM 388 may be referred to as a non-transitory computer readable medium or a computer readable storage media. A dynamic RAM embodiment of the RAM 388, likewise, may be referred to as a non-transitory computer readable medium in that while the dynamic RAM receives electrical power and is operated in accordance with its design, for

30

example during a period of time during which the object detection system 380 is turned on and operational, the dynamic RAM stores information that is written to it. Similarly, the processor 382 and/or GPU 394 may comprise an internal RAM, an internal ROM, a cache memory, and/or other internal non-transitory storage blocks, sections, or components that may be referred to in some contexts as non-transitory computer readable media or computer readable storage media.

Fig. 3 illustrates a method of determining a trajectory of a target object, in the form of a vehicle, with steps performed by the processor 382 and/or GPU 394. The method comprises steps 201 to 213. The steps of the method will now be described in detail.

In step 201, a plurality of source images in the form of a sequence of raw (i.e. unmodified) images captured by the camera 101 are received by the processor 382 and/or GPU 394, for example via one of the I/O devices 390 or the network connectivity devices 392.

In step 203, a binary saliency region image corresponding to each of the raw images is generated. The binary saliency region image is generated by first determining a saliency map of the raw image, for example using the approach described in Itti, L, Koch, C. and Niebur, E. *A Model of Saliency-Based Visual Attention for Rapid Scene Analysis* IEEE Transactions on Pattern Analysis and Machine Intelligence 20(11):1254-1259 (1998). A threshold is then applied to the saliency level of each pixel of each saliency map image so as to discard any pixels below the threshold saliency level.

In a saliency map, the rear lights of a vehicle (which are red) will naturally be attributed a high saliency level relative to a majority of other pixels in the image, thereby highlighting them. Thus, a threshold can be identified to distinguish pixels corresponding to rear lights of a vehicle from the majority of other pixels in the image.

In the described embodiment, an image segmentation method, such as OTSU's thresholding technology is adopted to generate the binary saliency region image from the saliency map. In this approach, the algorithm returns a single intensity threshold that separate pixels into two classes, foreground and background. The threshold is  
5 determined by minimizing intra-class intensity variance, or equivalently, by maximizing inter-class variance.

An example of this process is shown in Figs. 4 – 6. Figs. 4a, 5a and 6a show raw images. Figs. 4b, 5b and 6b show a saliency map corresponding to each respective raw image, and Figs. 4c, 5c and 6c show a binary saliency region image corresponding to each  
10 respective raw image, wherein all pixels of the original raw images 4a, 5a and 6a having a saliency level below a threshold (as visible in Figs. 4b, 5b and 6b) have been discarded. Figs. 4c, 5c and 6c therefore are identical to Figs. 4a, 5a and 6a but with certain pixels missing.

15

As can be seen from Figs. 4c, 5c and 6c the binary saliency region image retains the rear (red) lights of vehicles appearing in the raw images Figs. 4a, 5a and 6a. The significance of this will become apparent below.

20 In step 205, a raw image and its corresponding binary saliency region are input into a data augmentation model.

According to the described embodiment, the data augmentation model is a generative neural network which is trained using target images to learn a mapping from a source  
25 image  $x$  and a random noise vector  $z$  to an output image  $y$ , i.e. to perform the mapping  $G(x, z) \rightarrow (y)$ , where  $G$  is the data augmentation model. Neural networks (neural models) are adaptive models trained by machine learning methods comprising sets of algorithms configured to map inputs to outputs. A schematic of the simplest type of neural network is shown in Fig. 7. The neural network 19 comprises an input layer  
30 1901 where the input data is input into the network, one or more hidden layers 1903

where inputs are combined and an output layer 1905 at which the output is received. The hidden layer 1903 comprises a series of biased nodes 1909. Each input to each hidden layer is weighted and combined at a node with a non-linear activation function.

- 5 Neural networks are defined by a series of parameters including those characterizing an architecture of the neural network (i.e. number of nodes and number of hidden layers), activation functions, weights and biases. The weights and biases are determined during training of the neural network. The training of the data augmentation model according to the described embodiment will be described in  
10 detail below.

Note that although only one hidden layer is shown in Fig. 7, the neural network may comprise a plurality of hidden layers, according to the architecture employed.

- 15 The data augmentation model employed in step 205 of the method of Fig. 2 is trained to translate (map) a raw image and its corresponding saliency region image to a counterpart or corresponding augmented image in the form of a synthetic, or predicted image having labels associated with vehicles captured in the image. In the described embodiment, the labels are in the form of bounding boxes, having a  
20 boundary thickness of, for example, 5 pixels, enclosing the captured vehicles.

- The translation performed by the data augmentation model is illustrated in Fig. 8a and 8b, with Fig. 8a showing a raw image captured by the camera 101 and Fig. 8b showing a corresponding augmented image, output by the data augmentation model. Five  
25 target objects in the form of vehicles 701 are captured in the raw image of Fig. 8a. The translated image of Fig. 8b contains five corresponding target object labels in the form of bounding boxes 703 around each of the vehicles 701.

A class of the vehicle 701 may be denoted by a colour of the bounding box 703, for example, cars may be denoted by a white bounding box and lorries by a green bounding box.

- 5 In step 209, an object detector in the form of a hue, saturation, value (HSV) colour-based object detection model is used to detect the bounding boxes 703. The colour-based object detection model is configured to detect pixels having a colour corresponding to the bounding boxes 703 above a threshold value, thereby helping to enable the detection model to extract the locations of contours of the bounding boxes
- 10 703 (i.e. inner and outer edges of lines defining the bounding boxes 703) from the translated image.

- In step 211 merged objects in the object detection are separated in order to determine the object detection results. In order to achieve this, a call of an OpenCV function is
- 15 employed to extract pixels grouped as rectangles using all contours of bounding boxes 703 present in the image, and then subtracting this all-contours result from extracted pixels that are grouped using external contours, thereby separating any rectangles that are merged.

- 20 This process is illustrated in Figs. 9a, 9b and 9c which show a synthetic, or translated image with three bounding boxes 703. Fig. 9a shows object detection results based on all contours of the bounding boxes 703 present in the translated image. Fig. 9b shows object detection results based on external contours of the bounding boxes 703, and Fig. 9c shows the object detection results obtained by subtracting the results of Fig. 9b
- 25 from Fig. 9a.

- In step 213, tracking of any movement of bounding boxes 703 from image to image is performed in order to determine a trajectory of vehicles 701 captured in the images. Temporal spatial analysis, a process to examine if a target detected in a current frame
- 30 has occurred in a similar area in previous frames, can be used to track movement of

bounding boxes 703 with information stored for a predetermined period of time. As a result, locations of the bounding boxes 703 are spatially continuous on the image sequence whether the vehicles 701 (and correspondingly the bounding boxes 703) are moving or not.

5

In the described embodiment, a tracking history of a bounding box 703 corresponding to a vehicle instance is represented as a vehicle trajectory containing the following components: type; location; age; and discontinuity. The type and location of a vehicle instance are defined by the bounding box colour and location, respectively,

10 determined as described above. A number of the frames since a bounding box 703 was first detected gives the age of the vehicle trajectory. The discontinuity of the trajectory is defined as a number of frames since the last detection of the bounding box instance.

The vehicle trajectory is categorized as a Boolean variable, indicating a trajectory  
15 stability. Two classes of trajectories are then defined: (1) a stable vehicle; (2) a temporary vehicle. A stable vehicle corresponds to a vehicle trajectory which is confirmed as a continuous tracked bounding box 703.

A vehicle trajectory pool, a collection of all of the current trajectories is updated  
20 continuously. Initially, once a new bounding box 703 is detected, a temporary trajectory is initialized in the pool. To update a temporary trajectory to a stable trajectory, a minimal lifetime - one second in the described embodiment - and a minimal number of instances of the bounding box 703 in the images (i.e. the age) – five instances in the described embodiment - are required.

25

A trajectory is deleted from the pool if its lifetime is longer than a threshold, which is seventy seconds according to the described embodiment. This prevents the pool of trajectories from overflowing and ensures that trajectories corresponding to tracked objects which have left the field of view of the camera 101 and therefore no longer  
30 need to be tracked are deleted.

Fig. 10 illustrates a method performed when bounding box detection results corresponding to a new raw image frame are received by the CPU 382 and/or GPU 394. The method comprises steps 901 to 909. In step 901, information indicating a location of a detected bounding box 703 is received and added to the vehicle trajectory pool.

In step 903, the received bounding box location information is compared with existing vehicle trajectories.

In step 905 it is determined if any existing vehicle trajectories have a bounding box 703 which is less than a threshold distance from the location of the instant bounding box 703. In the described embodiment, the pre-defined threshold distance is sixty pixels.

If yes, then in step 907, the new bounding box location is added to the vehicle trajectory in which a distance of the new location of the bounding box 703 to the existing bounding box location of the trajectory is the shortest relative to the other vehicle trajectories in the pool.

If no, then in step 909, a new temporary vehicle trajectory is created corresponding to this new bounding box 703. The new bounding box 703 is classified as a stable vehicle if a stable trajectory is found. Otherwise, the new bounding box 703 (which could be a false positive) is recorded as a temporary vehicle.

In summary, if a bounding box 703 is recognized from a newly captured frame, those trajectories in the pool corresponding the bounding box 703 are checked and the new location added to the appropriate vehicle trajectory. Thus, a trajectory of a vehicle 701 is determined by tracking the movement of the bounding boxes 703 from frame to frame using the object detector, instead of tracking the movement of the corresponding vehicle 701.

In use, images are captured by the camera 101 and processed by the object detection system 380 in real time as the AV 100 is motion and trajectories of neighboring or nearby vehicles 701 are determined. The trajectory information is passed to the  
5 controller 105 which employs the trajectory information to inform the control of the movement of the AV 100. For example, the controller 105 may actuate the brakes to slow a forward movement of the AV 100, based on one or more nearby vehicle trajectories identified by the object detection system 380, for example to avoid collisions with detected stationary or moving vehicles 701.

10 In the described embodiment, the data augmentation model is pre-trained to label the images captured by the camera 101 in real time in step 207 of the method of Fig. 3. A method of training of the data augmentation model is outlined in Fig. 11. The method comprises steps 1001 to 1009.

In step 1001, training images in the form of raw training images are obtained.  
15 According to the described embodiment, these images are captured by an image capturing device, for example a camera, under conditions of poor or limited visibility, for example, heavy rain, fog, or snow.

In step 1003, saliency region images corresponding to these raw source images are determined as described above in relation to step 203 of Fig. 3.

20 In step 1005, the raw training images are annotated by manually labelling any vehicles 701 captured in the raw training images with coloured bounding boxes 703 as described above. Saliency region images corresponding to the annotated training images are also determined in the same was as described above.

The raw source images, saliency region images and annotated training images are then  
25 employed to train the data augmentation model in step 1007. As discussed above, the data augmentation model is a generative image-to-image translation model. In the described embodiment, the data augmentation model is trained using a conditional

Generative Adversarial Networks (GAN) approach in which the generative model is trained in tandem with a discriminator model configured to distinguish, or equivalently discriminate between augmented images and “real” images.

A conceptual representation of the training step 1007 is shown in Fig. 12. The training  
5 step 1007 itself comprises steps 1201 to 1209.

In step 1201, the raw training image and corresponding saliency region image obtained in steps 1001 and 1003, respectively, of the method of Fig. 11 are input into the data augmentation model. The model outputs a translated, or augmented, image in step 1203.

10 In step 1205, a saliency region image of the translated image is obtained as described above and this saliency region image of the translated image is compared with the saliency region image of the raw training image. The data augmentation model is updated based on the comparison between the two images. As will be explained below, in the described embodiment this comprises minimising a background  
15 preserving loss quantifying a dissimilarity between the two saliency region images.

In step 1207, the translated image and saliency region image of the translated image are input into a discriminator 1207 along with the annotated training image obtained as described above in step 1005 of the method of Fig. 11 and the corresponding saliency region image of the annotated training image. The discriminator is a neural  
20 network 19 as described with regard to Fig. 7 which maps an input image to a classification result in step 1209, the classification result indicating if the input corresponds to a “real” image, i.e. a manually annotated image, or a “fake” image, i.e. an augmented image translated by the data augmentation model.

Based on the results obtained in step 1209, the data augmentation model is updated  
25 so as to maximize the classification error (i.e. to “fool” the discriminator into believing the translated images are real images), while, in tandem with the updating of the data augmentation model, the discriminator is updated with the goal of minimizing the classification error (i.e. correctly detecting that translated images are “fake” images).

Thus, the discriminator,  $D$ , is adversarially trained to do as well as possible at detecting the data augmentation model's "fakes" and the data augmentation model  $G$  is trained to produce outputs that cannot be distinguished, or discriminated, from "real" images by the discriminator  $D$ .

5 Mathematically, the training proceeds as follows.

Assume  $\{X, S_X, Y, S_Y\}$  represents a training sample, where  $X$  and  $Y$  represent images,  $X$  being raw training images and  $Y$  being annotated training images, i.e. the same as  $X$  but with vehicle bounding boxes 703 in a specific colour manually annotated, labelled or drawn onto the image.  $S_X$  and  $S_Y$  are saliency region images of  $X$  and  $Y$ , respectively.

10 In the described embodiment, the raw training images are captured under conditions of poor visibility, such as rain. Fig. 13a to Fig. 13d and Fig. 14a to Fig. 14d show examples of two sets of training samples, respectively. Figs. 13a and 14a showing raw images  $X$  captured from an onboard camera through the windshield of an AV 100 under wet weather conditions; Figs. 13b and 14b show corresponding saliency map images of  $X$ , i.e.  $S_X$ ; Figs. 13c and 14c show annotated images  $Y$  with white bounding boxes 703 labelled over each vehicle 701; and Figs. 13d and 14d show corresponding saliency region images of  $Y$ , i.e.  $S_Y$ .

Generally, in conditional GAN a mapping is learnt from an observed image  $x$  and random noise vector  $z$  to  $y$ , where the data augmentation model is defined as

20  $G(x, z) \rightarrow (y)$ . In the described embodiment, this corresponds to the mapping  $G(X, S_X, Z) \rightarrow (Y, S_Y)$ , where  $Z$  is the corresponding noise vector.

A generative adversarial loss is generally defined as follows:

$$L(G, D) = L_{GAN}(G, D) + \lambda E_{x,y,z}[\|y - G(x, z)\|_1] \quad (1)$$

25 where

$$L_{GAN}(G, D) = E_{x,y}[\log(D(x, y))] + E_{x,z}[\log(1 - D(x, G(x, z)))] \quad (2)$$

and  $E_m$  is the expected value over the inputs  $m$  to the discriminator,  $D(x, G(x, z))$  is the discriminator's estimate that the probability that a translated image is fake,  $D(x, y)$  is the discriminator's estimate that the probability that a manually annotated image is real.  $\lambda$  controls the relative importance of the accuracy of the discriminator relative to ability of the data augmentation model to "fool" the discriminator.

In addition to the above described adversarial loss, the approach according to the described embodiment employs a background preserving loss which uses a comparison between saliency region images to represent loss (as represented conceptually by step 1205). The background preserving loss is defined as a pixel-wise weighted  $l_1$ -loss where the background is given weight 1 and a vehicle weight 0. Only pixels in the background in both original and translated images are considered. For the original  $(X, S_x)$  and translated  $(Y, S_y)$ ,

$$L_o = \|w(S_x, S_y) \odot (X - Y)\|_1 \quad (3)$$

where  $L_o$  is the background preserving loss.  $\odot$  is the element-wise product. The inclusion of  $L_o$  in the optimization equation of the model helps to ensure that the bounding box 703 is enforced to a captured vehicle 701 while maintaining the background.

The weight  $w(S_x, S_y)$  in (3) is defined as follows.

$$w(S_x, S_y) = \begin{cases} 1 & \text{if } S_x \neq S_y \\ 0 & \text{else} \end{cases} \quad (4)$$

Thus, the background preserving loss quantifies a dissimilarity between the saliency region images corresponding to the raw training images  $X$  and the saliency region images corresponding to the augmented images  $Y$  translated from the raw training images.

Overall, therefore, the GAN learning procedure aims at:

$$G^* = \arg \min_G \max_D L_{GAN}(G, D) + \lambda E_{x,y,z} [\|y - G(x, z)\|_1] + \lambda_o L_o \quad (5)$$

where  $\lambda_o$  controls the relative importance of the background preserving loss  $L_o$ .

The trained data augmentation model is then output in step 1009 for use in step 205 of the object detection method of Fig. 3.

The object detection method of Fig. 3 is formulated as a data augmentation problem.

- 5 Data-aware image augmentation is employed on raw images, using the domain knowledge about the object to be detected, and a new image with the target objects enclosed in bounding boxes 703 is generated. Object detection is then performed by detecting the coloured bounding boxes 703 on the new image, as opposed to the objects (i.e. the vehicles 701 themselves). This may improve object detection accuracy,
- 10 in particular traffic or vehicle detection under non-optimal conditions where the image of the vehicle 701 is captured and subjected by poor or degraded visibility caused by rain, fog, or snow.

- The system and method of determining the trajectory of a target object according to the described embodiment therefore focuses on enhancing object detection by
- 15 labelling images with bounding boxes 703. This is fundamentally different from conventional approaches which tend to be based on visual quality as a criterion. The system and method of the described embodiment may enable generated augmented images to be processed efficiently via the detection only of bounding boxes 703 that represent objects to be detected and tracked. In doing so, the burden of object
- 20 detection is directed to detecting colour bounding boxes 703 from the new images. Detection of colour bounding boxes 703 may be robust to the artefacts caused by the image translation because only the presence of bounding boxes 703 or the colour of the bounding boxes 703 are used to detect objects, thereby enhancing accuracy.

- 25 Thus, this method of detection may enhance the efficiency of trajectory determination and may be performed without use of a sophisticated algorithm nor computationally extensive training and learning of the synthetic data set to identify a bounding box 703.

The method of the described embodiment may be particularly effective in conditions of poor visibility, such as heavy rain. Compared with the existing de-rain approaches, the approach according to the described embodiment may not require radar or LIDAR. Further, the data augmentation model trained in accordance with the described may  
5 not require a large database of training images, thereby reducing the cost and time of training.

The method according to the described embodiment may further enhance vehicle awareness in conditions of poor visibility by exploiting the fact that the vehicle rear  
10 lights are usually turned on during rainy conditions in computing a saliency map of the image, and using it to formulate a background preserving constraint on the learning vehicle loss function. The use of rear lights may enable the key region of interest on images to be identified based on colour information and may ensure that the bounding  
box 703 is enforced to a captured vehicle 701, by ensuring that the lights of the vehicle  
15 701 are positioned within the bounding box 703. The described model may make the output indistinguishable from reality while enhancing the object detection ability. Thus, the use of the saliency region images may result in improved object awareness while preserving the background.

20 The use of bounding boxes 703 as well as training the model with a background preserving loss may enable the image background of images to be preserved which may ensure that other objects relevant to the driving of the AV 100 remain visible in the image and can themselves be tracked/avoided, as appropriate.

25 Fig. 15a-c illustrate the performance of the data augmentation model according to the described embodiment. Fig. 15a shows an original image captured under conditions of heavy rain. Fig. 15b shows the predicted image, i.e. the output of the data augmentation model. Fig. 15c shows the ground truth. It can be observed that the predicted bounding boxes 703 of Fig. 15b are close to their ground truth locations 705  
30 in Fig. 15c. Thus, tracking of the predicted bounding boxes 703 of Fig. 15b may enable

accurate determination of the trajectory of the vehicles 701 captured in the image of Fig. 15a.

Figs. 16 -19 further illustrate the performance of the data augmentation model according to the described embodiment. Figs. 16a, 17a, 18a and 19a show original images and Figs. 16b, 17b, 18b and 19b show corresponding predicted images. Once again, the vehicles 701 (with bounding boxes 703) were observed to be synthesized or predicted correctly. The bounding boxes 703 of the vehicles 701 were also found to be clear enough for detection using a colour-based object detection.

The trajectory determination described above in relation to Fig. 10 may enable accurate temporal spatial tracking of vehicles 701 via the tracking of the bounding box movement in two aspects: (1) smoothness may be improved as missing or low confident vehicle information may be identified; (2) isolated false positives may be removed.

Employing a GigE camera as the image capturing device (101) may ensure stability, for example no missing frames or delay.

The described embodiment should not be construed as limitative.

Although the labels are described as being bounding boxes 703 according to the described embodiment, they could take any form suitable for detection, for example a simple highlighting of the vehicle 701, etc. The label may not be visible to the human eye.

Although the described embodiment is directed to determining the trajectories of vehicles for controlling the operation of the AV 100, it will be appreciated that methods according to the described embodiment could be employed for the detection of any target object. The method of Fig. 3 described above may be implemented in

fields including (but not limited to) Advanced Driver Assistance System (ADAS), autonomous driving systems, and robotic vision and guidance systems.

Further, although the above discussion has focused on cars, the AV could be any type of vehicle, such as car, lorry, motorbike, bus, or a bicycle. The object detection system  
5 could also be configured to determine the trajectory of any type of vehicle or an object which is not a vehicle.

Although the data augmentation model of the described embodiment is trained using saliency region data, the data augmentation could alternatively be trained without  
10 employing saliency region data, i.e. to perform the mapping  $G(X, Z) \rightarrow (Y)$ . In this variation,  $\lambda_o$  is set to zero in equation (5), i.e. the final term of equation (5) vanishes. When the data augmentation model is trained without the use of saliency region data, saliency region data is no longer required as an input to the data augmentation model  
15 and step 203 of the method of Fig. 3 is omitted.

Similarly, it is envisaged that the background preserving loss could be employed  $L_o$  with optimization functions other than the generative adversarial loss function defined in equation (1) for training the data augmentation model and/or that the data  
20 augmentation model could be trained without the use of a discriminator.

Although the data augmentation model is described above as being trained using a conditional GAN approach, alternatively, a modified cycle GAN approach to train the data augmentation model is envisaged. As with conditional GAN, cycle GAN trains a  
25 generator to transform an image set to another image set. However, it requires that the trained network can transfer A to B as well as from B to A.

For cycle GAN, the data augmentation model is defined as  $G(X, a) \rightarrow (Y, b)$ , where X and Y represent images and a and b represent object bounding boxes 703. The discriminator is D as before. An additional function F is defined as  $F(Y, b) \rightarrow (X, a)$ . Then a generative

adversarial loss is defined as follows.

$$L(G, F, D_X, D_Y) = L_{GAN}(G, D_Y, X, Y) + L_{GAN}(F, D_X, Y, X) + \lambda L_{cyc}(G, F)$$

where  $\lambda$  controls the relative importance of the accuracy of the discriminator relative to  
5 ability of the data augmentation model to “fool” the discriminator, as before and

$$L_{GAN}(G, D_Y, X, Y) = E_{y \sim p-augment(y)} [\log(D_Y(y))] + E_{I \sim p-rainy} [\log(1 - D_Y(G(x)))]$$

$$L_{GAN}(F, D_X, Y, X) = E_{x \sim p-rainy(x)} [\log(D_X(x))] + E_{I \sim p-rainy} [\log(1 - D_X(F(y)))]$$

$$L_{cyc} = E_{x \sim p-rainy(x)} [\|F(G(x)) - x\|_1] + E_{y \sim p-augment(y)} [\|G(F(y)) - y\|_1]$$

where  $p - rainy(x)$  and  $p - augment(y)$  are distributions for rainy (i.e. source  
10 training images) and augmentation data, respectively.

In this variation, the learning procedure aims at:

$$G^*, F^* = \arg \min_{G, F} \max_{D_X, D_Y} L(G, F, D_X, D_Y) .$$

15 Cycle GAN does not require paired training samples, i.e. source and corresponding manually augmented images and therefore may enable improved training of the data augmentation model where the availability of paired samples is limited.

Although in the described embodiment, a generative neural network model is employed to generate the synthetic images, it is envisaged that other models capable  
20 of applying labels to images or generating synthetic images with labels could be employed.

Although trajectories are described above as being deleted from the pool if their lifetime is longer than a threshold, it is envisaged that the history could alternatively be divided into two if the lifetime of the trajectory exceeds the threshold, i.e. a new

trajectory may be generated for an object although its old trajectory is deleted from the pool because the lifetime of the old trajectory exceeds the threshold. This may help to ensure the continuous tracking of objects which remain in the field of view of the camera 101, i.e. at a detectable distance longer than the threshold lifetime (e.g. a vehicle moving in front of the AV 100) while preventing the trajectory pool from overflowing.

Although example values are given above for the minimal lifetime for a temporary trajectory to be updated to a stable trajectory, and a minimal number of instances, it will be appreciated that these values can be optimized using empirical trials.

Although the training images are described as being captured under conditions of poor visibility, such as rain, it is envisaged that the training images may instead or additionally include images captured under clear conditions, i.e. conditions of good visibility.

It is envisaged that the images captured by the camera 101 could be static images or video. In the case of the latter, data augmentation according to the described embodiment may be applied to individual frames of the video. The use of image capturing devices other than a camera 101 are also envisaged.

Although a GigE camera is employed at the camera 101 in the described embodiment, it will be appreciated that other cameras could be employed, for example a USB camera.

25

Although the data augmentation model is described as being pre-trained, it is envisaged that training of the model could be performed or updated based on images received by the camera 101, while the AV 100 is stationary or in use.

Although an example approach to generate a saliency map is given above, it is envisaged that any approach for generating saliency maps could be employed. Indeed, accurate trajectory results have been demonstrated using a range of approaches for generating saliency maps.

5

Generating binary saliency region images may comprise explicitly generating a saliency map from the source image in order to determine the saliency level of each pixel of the source image. Alternatively, it is envisaged that the saliency levels of the pixels may be determined directly from the source image itself, for example by employing a pixel-to-  
10 region saliency computation approach and using corner features extracted from the source image to generate the binary saliency region image.

It is envisaged that the object detector 380 may comprise greater or fewer components than shown in Fig. 2. For example, some of the components of Fig. 2 may  
15 be omitted, for example the GPU 394.

Having now described the invention, it should be apparent to one of ordinary skill in the art that many modifications can be made hereto without departing from the scope as claimed.