(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
09 January 2025 (09.01.2025)

WIPO | PCT

(10) International Publication Number
WO 2025/010034 A1

(51) International Patent Classification:
G01S 13/86 (2006.01) *G01S 17/88* (2006.01)

(21) International Application Number:
PCT/TR2023/050635

(22) International Filing Date:
04 July 2023 (04.07.2023)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **CARETTA ROBOTICS TEKNOLOJI A.S.** [TR/TR]; Fenerbahce Mahallesi Igrip Sokak No:13/1, Kadikoy/Istanbul (TR).

(72) Inventors: **DEMIREL, Hasan**; Eastern Mediterranean University, Famagusta, 99628 (CY). **NADERI, Farhang**; No. 15, Teknopark, Eastern Mediterranean University, Famagusta, 99628 (CY). **BILGEN, Ahmet**; Oguzlar Sk. No 15, Famagusta (CY).

(74) Agent: **DERIS PATENT VE MARKA ACENTALIGI A.S.**; Inebolu Sokak No:5 Deris Patent Binasi, Kabatas Setustu, 34427 Beyoglu/Istanbul (TR).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:
— *as to the identity of the inventor (Rule 4.17(i))*

(54) Title: DEEP LEARNING BASED UNDERWATER MOVING OBJECT TRACKING AND IDENTIFICATION ON AUV-BASED EMBEDDED SYSTEM

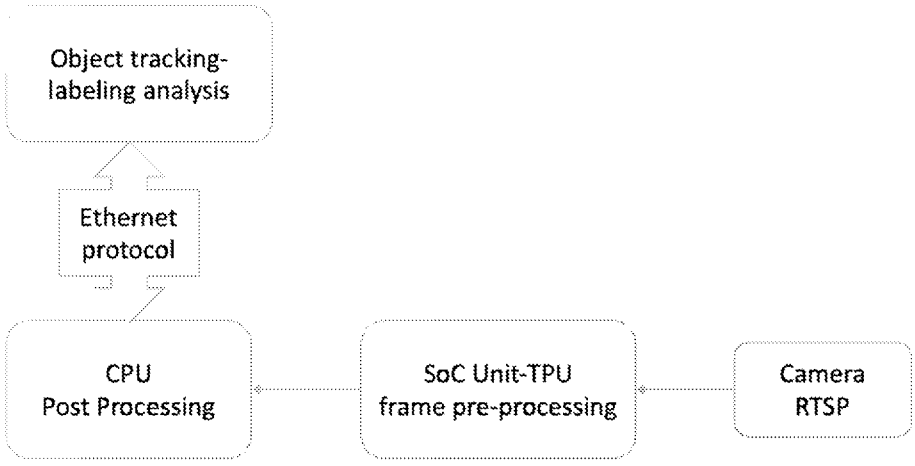


Figure 2. System connection for the integrated AI module

(57) Abstract: The present invention relates to a hardware system for AI-based underwater vehicle, comprising a lidar on the underwater vehicle, an AI camera, an embedded edge device configured to run two cascaded algorithms wherein one of the algorithms is a deep learning based moving object detector and the other is a deep learning based object classifier, an integrated AI module wherein the module includes a computer containing at least one specialized processing unit. The present invention further relates to a system for tracking and identification of underwater moving objects.

[Continued on next page]

WO 2025/010034 A1

Published:

— *with international search report (Art. 21(3))*
— *with amended claims (Art. 19(1))*
— *in black and white; the international application as filed contained color or greyscale and is available for download from PATENTSCOPE*

DEEP LEARNING BASED UNDERWATER MOVING OBJECT TRACKING AND IDENTIFICATION ON AUV-BASED EMBEDDED SYSTEM

Field of the Invention

The present invention relates to a method of tracking and identification of underwater objects and a system performing the same to be used in marine engineering-based applications requiring underwater robotics, machine vision and artificial intelligence.

Background of the Invention

Detection and classification of underwater objects, including the living organisms, is used for various purposes, for example ocean exploration and marine aquaculture. Autonomous underwater vehicles (AUVs) play a key role in monitoring the marine environment for these purposes. AUVs are robotic vehicles preprogrammed to travel underwater and collect data without an input from an operator. In other words, AUVs operate independently of humans, therefore they are also called unmanned underwater vehicles (UUVs). AUVs have no physical connection to their operator, they are untethered. On the other hand, remotely operated underwater vehicles (ROUVs) which are also UUVs, are tethered to a service vessel or on proximate land.

AUVs cannot transmit the images or videos instantaneously since it is not possible to communicate underwater without a cable. Therefore, AUVs store all the data they collect on onboard computers to be retrieved after the completion of the dive mission. AUVs have the advantage of going into shallow areas since they are small and can move freely. As the name suggests, they are operating independently, allowing the researchers to focus on other aspects of their projects. That is why their popularity increases in marine research areas.

Although, AUVs have advanced technological capabilities, without a useful software, they cannot perform their intended function. The computer software and the algorithm which provides a computer implemented method have a key role in the performance of AUV-based embedded systems.

Accordingly, Artificial Intelligence (AI) based underwater robotics have been attracting an enormous amount of attention during the last decade (Sarker, 2021). The developments in these technologies are more possible than ever due to advancements in three key technologies which are machine learning, machine vision and embedded systems.

AI, in particular machine learning, has grown rapidly in recent years in the context of data analysis and computing that typically allows the applications to function in an intelligent manner. Machine learning usually provides systems with the ability to learn and enhance from experience automatically.

Machine learning algorithms are mainly divided into three categories: Supervised learning, Unsupervised learning, and Reinforcement learning.

Supervised learning is the most common type of machine learning. The task of this machine learning is to learn a function that maps an input to an output, based on sample input-output pairs (Data Mining: Concepts and Techniques - 3rd Edition, n.d.). It uses labelled training data and a collection of training examples to infer a function. Here, the dataset provided for the training purposes is very important. The high number of labelled input data would increase the chances of the function to be more successful. However, collection of the data is a challenging task for humans, and it is not always possible to obtain sufficient amount of example data. This is one of the challenges in the development of machine learning algorithms.

Unsupervised learning analyses unlabeled datasets without the need for human interference (Data Mining: Concepts and Techniques - 3rd Edition, n.d.). This is widely used for clustering data. In unsupervised learning, as mentioned, the input data is not labelled, therefore, the algorithm cannot evaluate the suitability of the function against the output values in the dataset. Instead, the aim of the algorithm is to cluster similar examples in groups.

Reinforcement learning is a type of machine learning algorithm that enables machines to automatically evaluate the optimal behaviour in a particular environment to improve its

efficiency. Reinforcement learning is most relevant in autonomous driving tasks and robotics. The goal of the machine is to learn a mapping from its observation of the environment and its memory to what action it should take. This type of learning is based on reward or penalty. When the machine is released to the related environment, it experiments various policies and updates its policy according to the rewards and penalties it receives. If an action results in a negative reward, the mapping is weakened, if an action results in a positive reward, the mapping is reinforced.

Deep learning is a subfield of machine learning which can be applied to all three learning types, supervised, unsupervised and reinforcement. Deep learning is based on artificial neural networks and has the advantage of learning useful derived features from data automatically. It also has the ability to learn complex mappings between inputs and outputs.

Deep learning networks are mathematical models inspired by the structure of the human brain. It requires the use of multiple layers of simple information processing programs (neurons) in the network, that is why they are called “deep” learning.

The human brain is composed of massive number of neurons. Neurons are basic unit of communication having a cell body, a major branching fiber (axon) and smaller branching fibers (dendrites). They are electrically excitable cells that communicate with other nerve cells via specialized connections called synapses.

A synapse is a structure which permits a neuron to pass an electrical or chemical signal to another neuron. Dendrites are input channels of the neurons and axons are output channels. In other words, neurons receive signals via the dendrites and send out signals down the axon. Signals cross from the axon of one neuron to a dendrite of another.

Similarly, an artificial neural network consists of a network of information units, called neurons. Neurons in a neural network are organized into layers; one input layer, at least two hidden layers and one output layer. Each connection in a network connects two neurons and has a weight associated with it. Training an artificial neural network involves searching for the best set of weights.

Neurons perform a two-stage process. First, a weighted summation over the inputs to the neuron is calculated, second, the result of the weighted summation is passed through a nonlinear function. Deep learning requires high number of network runs, accordingly, millions of calculations. It is therefore important to speed up the process. Graphics processing units (GPUs) is a way to speed up the training of networks.

There are many ways to connect the neurons in a neural network. Convolutional neural network (CNN) is one of the network architectures, specifically developed for the task of image processing. AlexNet can be considered as the most popular CNN which win the ImageNet 2012 competition. Following the development of AlexNet subsequent winning architectures use more layers in a deep neural network to reduce the error rate. However, when the number of layers increases, the training and test error rate also increases because of the vanishing/exploding gradient problem. In order to solve this problem Residual Network (ResNet) architecture is used.

CNNs are commonly developed at a fixed cost, and then scaled up in order to achieve better accuracy when more resources are made available. For example, ResNet can be scaled up from ResNet-18 to ResNet-200 by increasing the number of layers, and recently, GPipe achieved 84.3% ImageNet top-1 accuracy by scaling up a baseline CNN by a factor of four. The conventional practice for model scaling is to arbitrarily increase the CNN depth or width, or to use larger input image resolution for training and evaluation. While these methods do improve accuracy, they usually require tedious manual tuning, and still often yield suboptimal performance.

EfficientNet is a CNN architecture and scaling method that uniformly scales all dimensions of depth/width/resolution. The EfficientNet scaling method uniformly scales network width, depth, and resolution with the use of a compound coefficient. EfficientNets have been shown to achieve state-of-the-art accuracy in image classification tasks while significantly reducing the model size and computational complexity.

Another important concept in deep learning-based object tracking and identification is machine vision algorithms. Machine vision is the ability of a computer to see. In object recognition 2D-CNN architectures are used by relying on architectures build upon

ImageNet classifiers such as AlexNet, ResNet and GoogLeNet. These have had great success on the popular ILSVRC1 challenge starting with the breakthrough by AlexNet which used GPU computation in the training step.

CNNs are used in vision applications because this type of data nearly always has spatial relationships between related objects in the image. The computation complexity of high-resolution images is reduced by down sampling and using sliding windows scanning the whole image and selecting a region of interest (RoI).

In object localization CNNs are being used for example in [1] where YOLO [2] model (proposed in 2015) is used to identify handles of doors and estimate its pose for a grasping application tested on a mobile robot. There is an even faster, lightweight variant called Faster YOLO achieving higher processing speed. ZFNet [3] - a faster R-CNN variant, is used in [4] to detect multiple kiwifruits from images in clustered scenes.

When taking video sequences as input, typical problems are the classification of actions or motion planning. Here DCNN networks are used for example the VGG [5] architecture [6], as well as Self Organizing Maps network which in [7] receives information about the human's location and pose in a robot workspace based on pressure activated notes in a safety mat.

As mentioned above, another key technology in the development of underwater robotics is embedded systems. An embedded system is a computer system which is completely encapsulated by the device it controls. There are some specific requirements for each system, such as functions, reliability, cost, size, and power consumption. Based on the computer technology, an embedded system is designed for specific application with hardware and software that could be tailored to adapt the system requirements. As the core device of an embedded system, the embedded microprocessor can be an 8-, 16- or 32-bit microprocessor. Because of the limitation from the lower performance such as low running speed, low addressing capability and high-power consumption, etc, the 8-or 16-bit microprocessors cannot meet the requirements of some complex embedded applications.

In the field of the 32-bit embedded system application, ARM (Advanced RISC Machine) gains tremendous success [8], [9], [10]. The ARM-kernel microprocessors present a smaller size, lower power consumption, lower cost, and relatively higher performance. For instance, with a plenty of registers and high instruction executing speed, most data operations are completed in registers. Besides, the ARM-kernel microprocessors have advantages in addressing flexibility, facility, and higher execution efficiency as well as fixed lengths of instructions, and so on. An embedded operating system can be encapsulated in ARM, to accomplish complex algorithms and to finish a variety of tasks instead of personal computers.

An embedded system is made up of the embedded operating system and the hardware supporting the operating system. The embedded operating system is the console of an embedded system, which is mainly used for controlling devices of information processing as well as user interfaces. The hardware components of embedded systems should conform to specific applications, so that there are strict limitations on them, such as cost, function, size, weight and power consumption, etc.

The development of embedded software is closely related to the hardware. The embedded software is implemented on a certain hardware platform, dealing with some parts of hardware-software co-design on device drivers. Due to the limited memory space for programs running in the embedded system, it is necessary to take the code efficiency into account during programming. For a real-time operating system, the processor should rigidly deal with all kinds of tasks produced asynchronously. Besides, an embedded software system is supposed to have some capabilities such as exception handling and quick reset, etc. Generally, to improve both executing speed and system reliability, and as well, to reduce system-reset time, the embedded software is downloaded and solidified into FLASH or ROM of the target device.

Hardware of an embedded system has small size and high integration. An embedded system is a minimum system without a lot of components needless for target functions. Hardware of the system should be capable of working in bad environments and supporting quick restart when the system is dead.

Currently, there are some solutions for machine learning and object detection or image classification, including Embedded Devices to implement image classifiers (Nvidia Jetson, Intel NuC etc.), Yolo stacks for object detection, Fast-R-CNN, SSD and R-FCN.

On the other hand, in the detection of underwater objects, there is a need in the art to a method of tracking and identification of underwater objects and a system performing the same, which is fast and having a high accuracy.

Description of the Drawings

Figure 1: The hardware module deployed on underwater vehicle.
Figure 2: System connection for the integrated AI module.
Figure 3: Buffer of decisions, decision fusion in temporal domain.
Figure 4: The datasets used for training deep learning architecture for classification.

Summary of the Invention

In one aspect, the present invention provides a hardware system for AI-based underwater vehicle, comprising a lidar on the underwater vehicle, an AI camera, an embedded edge device configured to run two cascaded algorithms wherein one of the algorithms is a deep learning based moving object detector and the other is a deep learning based object classifier, an integrated AI module wherein the module includes a computer containing at least one specialized processing unit.

In a further aspect, the computer of the hardware system may comprise at least one specialized processing unit which can be a Tensor Processing Unit (TPU), a Neural Processing Unit (NPU) and/or an ARM Cortex based CPU. These units are used for accelerating deep learning inference at the edge. Examples of such devices include the Google Coral TPU board, Edge TPU Dev Board by Google, Movidius Neural Compute Stick 2 by Intel, Jetson Nano by NVIDIA, and Rock Pi N10 by Radxa.
In another aspect, the lidar is 360-degree lidar.

In a further aspect, the present invention further provides a method for tracking and identification of underwater moving objects through deep learning, comprising the steps of:

- Training an object detection network with training images to identify and classify objects in images from a sensor system present on an unmanned underwater vehicle (UUV), particularly an autonomous underwater vehicle (AUV),
- Identifying the objects in the video frames,
- Classifying objects of each frame using a processor, and
- Applying majority voting to make classification decision.

Detailed Description of the Invention

The present invention provides a deep learning-based method to be used for tracking and identification of underwater moving objects.

Detecting underwater objects presents a multitude of challenges due to the distinct features of aquatic environments. These difficulties include:

1. Attenuation in water: Water is a highly attenuating medium, restricting light penetration to only a few meters before it becomes undetectable. Consequently, object detection algorithms must depend on the scarce light present in underwater settings.
2. Scattering and absorption: Light scatters and absorbs differently in water compared to air, complicating the process of clearly observing objects. This effect becomes particularly pronounced in turbid or murky waters, where scattering is more substantial.
3. Water currents: Objects moving underwater are subject to currents that can render their motion erratic and hard to predict, posing challenges for stationary cameras attempting to capture images amid fluctuating water currents.
4. Fluctuating lighting conditions: Rapid changes in lighting conditions can occur underwater due to factors such as time of day, weather, and water depth. These variations complicate the accurate detection and tracking of objects.
5. Restricted visibility: Visibility in underwater environments can be limited, particularly in deep or murky waters, making object detection and tracking more difficult.
6. Object movement: Underwater objects can move in three dimensions, adding complexity to accurately tracking their motion.

7. Image distortion: Water-induced image distortion can cause objects to appear larger or smaller than their actual size, complicating the estimation of their dimensions and distances.

The mentioned challenges have been addressed by the present invention. In essence, the complexities of underwater object detection stem from the unique characteristics of aquatic environments, such as water attenuation, scattering and absorption, water currents, variable lighting conditions, limited visibility, object movement, and image distortion.

The present invention encapsulates two modules, the first is the hardware module and the second is the intelligent software module.

Accordingly, the present invention provides a hardware system for AI-based underwater vehicle, comprising a lidar on the underwater vehicle, an AI camera, an embedded edge device configured to run two cascaded algorithms wherein one of the algorithm is a deep learning based moving object detector and the other is a deep learning based object classifier, an integrated AI module wherein the module includes a computer containing at least one specialized processing unit.

The unique hardware of the present invention is developed to enable an AI-based underwater vehicle to be used for detection, tracking and classification of underwater objects. Said hardware can also be used with a stationary underwater object. **Figure 1** demonstrates the whole hardware module setup.

In another embodiment the lidar is 360-degree lidar. Alternatively, more than one lidar can be used to cover the environment.

The 360-degree lidar is designed at the centre top of the underwater vehicle/object to provide 3D observation capability under the water. The 360-degree lidar provides important 3D depth information being fed to the AI-based decision pipeline. Alternatively, more than one lidar can be mounted on various regions of the vehicle/object to cover the whole environment.

In one embodiment an inspection camera, which does not play a role in AI-based decision pipeline, continuously records the surrounding activities at for example 720p quality.

In one embodiment, the computer of the hardware system may comprise at least one specialized processing unit which can be a Tensor Processing Unit (TPU), a Neural Processing Unit (NPU) and/or an ARM Cortex based CPU. These units are used for accelerating deep learning inference at the edge. Examples of such devices include the Google Coral TPU board, Edge TPU Dev Board by Google, Movidius Neural Compute Stick 2 by Intel, Jetson Nano by NVIDIA, and Rock Pi N10 by Radxa.

Said CPU is configured to apply majority voting.

To provide a real-time moving object detection with an operating vehicle or a stationary object, coupling of SSD with Fast R-CNN is important. To provide a system with the least cooling, it has been found that the best CPU is the ARM Architecture CPU, preferably the ARM Cortex Architecture CPU.

In one preferred embodiment, the computer of the hardware system comprises two specialized processing units which are TPU and ARM Cortex based CPU.

In one embodiment, the camera of the hardware system is a is a high-definition camera with at least Full-HD with 1080 p resolution; preferably **4K** AI camera.

The AI camera, preferably of 4K quality plays a crucial role in AI-based decisions. The high-quality image frames acquired through AI camera feeds the SSD reducing the region of interest to further processed by Fast R-CNN Architecture. Coupling SSD with Fast R-CNN for underwater video frames in a low-capacity embedded edge device forms a novel framework of the present invention.

In another embodiment the computer of the hardware system is SoC computer (Google Coral TPU board) containing a TPU unit and an ARM Cortex Architecture.

The integrated AI module is illustrated in **Figure 2** is designed such that under the lower ventilation circumstances in an underwater environment the performance is preserved without any interruptions. Inside this special module there is a SoC computer (Google Coral TPU board) containing a TPU unit in addition to an ARM Cortex CPU. The TPU unit takes care of producing decisions based on a combination of Fast-RCNN network and the CPU ability to apply majority voting.

Specifically, the present invention provides a hardware system for AI-based underwater vehicle, comprising a lidar on the underwater vehicle, an AI camera, an embedded edge device configured to run **SSD** coupled with **Fast R-CNN**, an integrated AI module wherein the module includes a computer containing a Tensor Processing Unit (TPU) and a Central Processing Unit (CPU)**.**

According to the present invention, TPU is capable of doing deep learning based object detection and classification followed by majority voting along with the fusion of the decisions from consecutive frames using CPU.

Regarding the intelligent software module of the present invention, the standard detection algorithms such as Fast-RCNN and SSD are used as the backbone of the present smart detection procedure which is followed by intelligent data fusion process to increase the detection performance. Per frame accuracy of Fast-RCNN or SSD is much lower than the targeted accuracy of the multi-frame video. The suggested method of majority voting applied to the time-based detections derived from a series of frames enhances the ultimate detection precision to a perfect 100%.

The exceptional final results are achievable when employing a standard detector with an average performance exceeding 50% by utilizing a series of combined temporal detection decisions, rather than relying on separate single frame-based decisions for the same video frames. The proposed framework is driven by decision fusion across the temporal domain, corresponding to numerous consecutive frames. Independent single frame decisions over N sequential frames will produce the same average performance, which is selected to be higher than 50% (e.g., 55%).

However, merging N decisions from N consecutive frames through majority voting in the temporal domain raises the combined decision accuracy to 100%, as the average performance from each frame surpasses 50% (e.g., 55%). This becomes feasible because, given an infinite number of votes (i.e., frames), 55% of voters will correctly identify the result compared to the 45% who may detect it inaccurately or fail to detect it at all. Asymptotically, the outcome is guaranteed to be 100% accurate.

This result is possible using the buffer of decisions method given in **Figure 3** which gives the details of the buffer of decisions approach for decision fusion in the temporal domain. As an example, Algorithm 1 below gives the details of the object detection using majority voting according to the present invention.

Algorithm 1 Object Detection using Majority Voting

```
Initialize frameList ← []
Initialize detectedObjectsList ← []
procedure MajorityVoting(objectsList)
    Initialize votes ← empty dictionary
    for object in objectsList do
        if object not in votes then
            votes[object] ← 1
        else
            votes[object] ← votes[object] + 1
        end if
    end for
    majorityObject ← max_{object∈objectsList} votes[object]
    return majorityObject
end procedure
while True do
    frame ← Receive a frame from the camera
    detectedObject ← Detect objects in frame
    if Length of frameList < 30 then
        Append frame to frameList
        Append detectedObject to detectedObjectsList
    else
        majorityObject ← MajorityVoting(detectedObjects)
        Output majorityObject
        Discard the first element of frameList
        Discard the first element of detectedObjects
        Append frame to frameList
        Append detectedObject to detectedObjects
    end if
end while
```

Algorithm 1. object detection using majority voting

In one embodiment, the items are classified into two primary groups, namely objects and animals, as depicted in **Figure 4**. In said embodiment, which is demonstrated in Figure 4, there are seven objects to be classified. Five of these are moving objects under the animal category whereas the remaining two categories are stationary.

The proposed machine learning framework is scalable to include additional categories to be classified. This adaptive approach allows customised datasets to suit specific needs.

The following is an example of the system preparation according to the invention:

1. Collect and preprocess the data:
    - Collect real-life underwater scenery data from videos.
    - Label the objects in the images using annotation tools like Labelling.
    - Convert the images to the required format for training the model.
2. Set up the Google Coral Dev Board:
    - Install Ubuntu 18.04 on a computer with a USB port.
    - Download the Coral Dev Board flasher tool and the Coral Dev Board USB driver.
    - Insert the Coral Dev Board into the USB port of the computer.
    - Run the flasher tool and flash the Coral Dev Board with the latest version of Mendel Linux.
    - Connect the Coral Dev Board to a monitor, keyboard, and mouse using HDMI, USB, and Bluetooth.
    - Connect a camera to the Coral Dev Board using USB.
3. Install TensorFlow Lite:
    - Install Python 3.7 on the Coral Dev Board
    - Install TensorFlow Lite using pip:
4. Choose a pre-defined model:
    - Select a pre-trained object detection model that is compatible with TensorFlow Lite and the Coral Dev Board.
    - Download the model from the TensorFlow Hub or GitHub.
5. Fine-tune the pre-defined model:
    - Install TensorFlow on your computer.

- Use TensorFlow's transfer learning API to fine-tune the pre-defined model for object detection using your preprocessed data.
- Export the fine-tuned model in TensorFlow Lite format.

6. Optimise the model:
   - Use the Edge TPU Compiler to optimise the model for running on the Coral Dev Board.
   - Download and install the Edge TPU Compiler on your computer.
   - Compile the model using the Edge TPU Compiler.
7. Deploy the model:
   - Copy the compiled model file to the Coral Dev Board.
   - Install OpenCV on the Coral Dev Board.
   - Use a Python script to test the model on real-world data.
8. Iterate and improve:
   - Iterate and improve the model by tweaking its architecture or fine-tuning it with more data until you achieve the desired performance.

In another embodiment, the mentioned embedded edge device has a low capacity. Herein, the embedded edge device has a low computational capability with power efficiency and low heat dissipation and may preferably use passive cooling solutions.

In the system, it is preferred that an additional camera is included for inspection. In another preferred configuration, a temperature system and/or depth sensor and/or a compass is included.

The system of the present invention particularly suitable for unmanned underwater vehicle (UUV), preferably an autonomous underwater vehicle (AUV).

The present invention further provides a method for tracking and identification of underwater moving objects through deep learning, comprising the steps of:

- Training an object detection network with training images to identify and classify objects in images from a sensor system present on an unmanned underwater vehicle (UUV), particularly an autonomous underwater vehicle (AUV).

- Identifying the objects in the video frames
- Classifying objects of each frame using a processor
- Applying majority voting to make classification decision.

In one preferred embodiment, the method according to the present invention comprises the steps of:

- Collecting underwater scenery data,
- Labelling the objects in the images using annotation tools such as labelling,
- Converting images to the required format for training the model,
- Training the object detection model to detect the underwater objects,
- Detecting the underwater objects,
- Receiving the real-time underwater object list,
- Applying majority voting on the detected object list,
- Identifying the detected underwater objects.

References

[1] Li, Y., Zhu, H., & Lin, J. (2017). Door handle detection and localization for mobile robot grasping using deep convolutional neural networks. Journal of Intelligent & Robotic Systems, 87(1), 85-99.

[2] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).

[3] Zeiler, M. D., & Fergus, R. (2014). Visualising and understanding convolutional networks. In European conference on computer vision (pp. 818-833). Springer, Cham.

[4] Zheng, Y., Yao, C., & Zhang, J. (2017). Kiwi fruit detection and segmentation based on Faster R-CNN. Journal of Physics: Conference Series, 892(1), 012044.

[5] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[6] Li, X., Li, Y., Li, J., & Li, Y. (2019). Human pose recognition based on pressure activated notes and convolutional neural network. Multimedia Tools and Applications, 78(15), 22169-22184.

[7] Khaldi, A., & Amira, A. (2012). Design and implementation of a programmable mobile robot based on embedded system using ARM processor. Robotics and Autonomous Systems, 60(4), 537-544.

[8] Wu, X., Wu, W., & Guo, W. (2015). Design of low power consumption embedded system based on ARM. Journal of Chemical and Pharmaceutical Research, 7(9), 776-780.

[9] Wang, H., Chen, Y., & Gao, S. (2016). A low-power embedded system based on ARM and Android for monitoring air quality. Journal of Chemical and Pharmaceutical Research, 8(8), 634-640.

[10] Shen, Y., Xie, X., & Chen, Y. (2017). Design and implementation of a smart home system based on ARM. Journal of Physics: Conference Series, 892(1), 012077.

CLAIMS

1. A hardware system for AI-based underwater vehicle, comprising a lidar on the underwater vehicle, an AI camera, an embedded edge device configured to run two cascaded algorithms wherein one of the algorithms is a deep learning based moving object detector and the other is a deep learning-based object classifier, an integrated AI module wherein the module includes a computer containing at least one specialized processing unit.

2. A hardware system according to claim 1 wherein the deep learning based moving object detector is SSD.

3. A hardware system according to claim 1 or 2 wherein the deep learning-based object classifier is Fast R CNN.

4. A hardware system according to any one of claims 1 to 3 wherein the camera is a high-definition camera with at least Full-HD with 1080 p resolution.

5. A hardware system according to claim 4 wherein the AI camera is 4K AI camera.

6. A hardware system according to any one of claims 1 to 5 wherein the computer is system-on-a-chip (SoC) computer.

7. A hardware system according to any one of claims 1 to 6 wherein the specialized processing unit is a Tensor Processing Unit (TPU), a Neural Processing Unit (NPU) and/or an ARM Cortex based CPU.

8. A hardware system according to any one of claims 1 to 7 wherein the specialized processing units are TPU and ARM Cortex based CPU.

9. A hardware system according to any one of claims 1 to 7 wherein the lidar is 360-degree lidar.

10. A hardware system according to any one of claims 1 to 8 further comprising a *Scanning Imaging Sonar which* is an underwater 360-degree sonar.

11. A hardware system according to any one of claims 1 to 9, wherein the CPU is configured to apply majority voting.

12. A hardware system according to any one of claims 1 to 11, further comprising an inspection camera.

13. A hardware system according to any one of claims 1 to 12, further comprising temperature sensor.

14. A hardware system according to any one of claims 1 to 13, further comprising depth sensor.

15. A hardware system according to any one of claims 1 to 14, further comprising a compass.

16. A system for tracking and identification of underwater moving objects, including a hardware system of claim 1 and an intelligent software.

17. A method for tracking and identification of underwater moving objects through deep learning, comprising the steps of:

- Training an object detection network with training images to identify and classify objects in the real-time images or videos obtained from a sensor system present on an underwater vehicle,
- Identifying the objects in the video frames or images by using the trained object detection network,
- Classifying objects of each frame using a processor, and
- Applying majority voting to make classification decision.

18. A method according to claim 16, comprising the steps of:

- Collecting underwater scenery data,
- Labelling the objects in the images using annotation tools such as labelling,
- Converting images to the required format for training the model,
- Training the object detection model to detect the underwater objects,
- Receiving real-time images of the underwater objects,
- Receiving the real-time underwater object list,
- Applying majority voting on the detected object list,
- Identifying the detected underwater objects.

AMENDED CLAIMS
received by the International Bureau on 04 November 2024 (04.11.2024)

1. A hardware system for AI-based underwater vehicle, comprising
- a lidar on the underwater vehicle configured to provide 3D depth information to the AI-based decision pipeline,
- an AI camera, configured to capture images for further processing,
- an embedded edge device configured to run two cascaded algorithms wherein one of the algorithms is a deep learning based moving object detector and the other is a deep learning-based object classifier,
- an integrated AI module wherein the module includes a computer containing two specialized processing unit, one processing unit is configured to perform object detection and classification, and the other processing unit is configured to manage temporal fusion using majority voting on the classification results across multiple consecutive frames.

2. A hardware system according to claim 1, wherein one of the two specialized processing units is a Tensor Processing Unit (TPU) configured to perform object detection and classification at the edge, and the other specialized processing unit is a Central Processing Unit (CPU) configured to manage intelligent decision-making processes by applying fusion of decisions in the domain using majority voting on the classification results across multiple consecutive frames.

3. A hardware system according to claims 1 or 2 wherein the deep learning based moving object detector is SSD.

4. A hardware system according to claim 1 or 3 wherein the deep learning-based object classifier is Fast R-CNN.

5. A hardware system according to any one of claims 1 to 4 wherein the camera is a high-definition camera with a minimum resolution of 1080p (Full HD).

6. A hardware system according to claim 5 wherein the AI camera is 4K AI camera.

AMENDED SHEET (ARTICLE 19)

7. A hardware system according to any one of claims 1 to 6 wherein the computer is system-on-a-chip (SoC).

8. A hardware system according to claim 1 wherein one of the specialized processing units, configured to perform object detection and classification, is a Tensor Processing Unit (TPU), a Neural Processing Unit (NPU) or an ARM Cortex based CPU.

9. A hardware system according to any one of claims 1 to 8 wherein the specialized processing units are TPU and ARM Cortex based CPU.

10. A hardware system according to any one of claims 1 to 9 wherein the lidar is 360-degree lidar.

11. A hardware system according to any one of claims 1 to 10 further comprising a Scanning Imaging Sonar configured as an underwater 360-degree sonar.

12. A hardware system according to any one of claims 1 to 11, further comprising an inspection camera.

13. A hardware system according to any one of claims 1 to 12, further comprising temperature sensor.

14. A hardware system according to any one of claims 1 to 13, further comprising depth sensor.

15. A hardware system according to any one of claims 1 to 14, further comprising a compass.

16. A system for tracking and identification of underwater moving objects, including a hardware system of claim 1 and an intelligent software.

AMENDED SHEET (ARTICLE 19)

17. A method for tracking and identification of underwater moving objects through deep learning, comprising the steps of:

- Training an object detection network with training images to identify and classify objects in the real-time images or videos obtained from a sensor system present on an underwater vehicle,
- Identifying the objects in the video frames or images by using the trained object detection network,
- Classifying objects of each frame using a processor, and
- Applying majority voting to make classification decision.

18. A method according to claim 17, further comprising the step of:

- Generating regions of interest from the input data using a deep learning-based object detection algorithm, such as SSD, to optimize the classification process by reducing the amount of data to be processed.

19. A method according to claim 17 or 18, further comprising the steps of:

- Aggregating the classification results in temporal domain obtained from consecutive frames using decision level fusion.
- Applying majority voting on the aggregated results to ensure consistency and reliability in object identification under varying underwater conditions.

20. A method according to claim 17, comprising the steps of:

- Collecting underwater scenery data,
- Labelling the objects in the images using annotation tools such as labelling,
- Converting images to the required format for training the model,
- Training the object detection model to detect the underwater objects,
- Receiving real-time images of the underwater objects,
- Receiving the real-time underwater object list,
- Applying majority voting on the detected object list,
- Identifying the detected underwater objects.

AMENDED SHEET (ARTICLE 19)

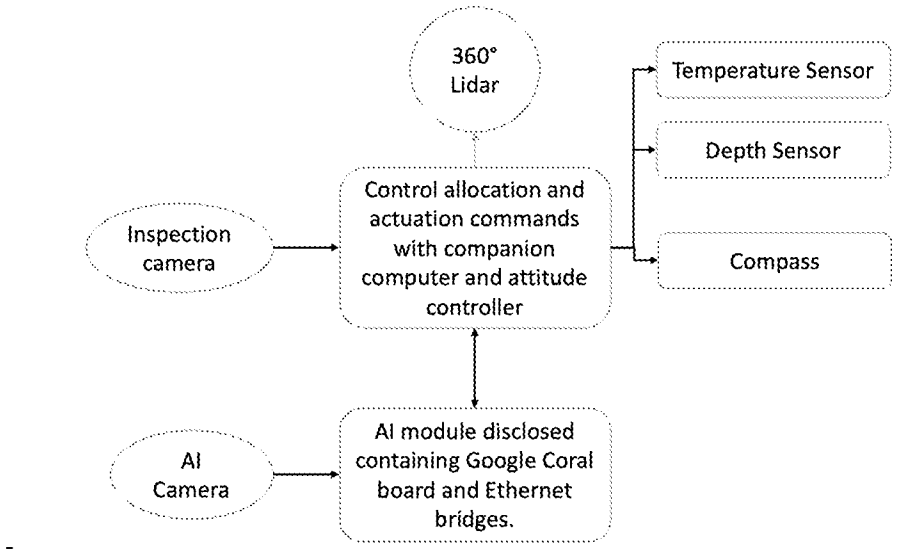


Figure 1 Proposed hardware module deployed on underwater vehicle

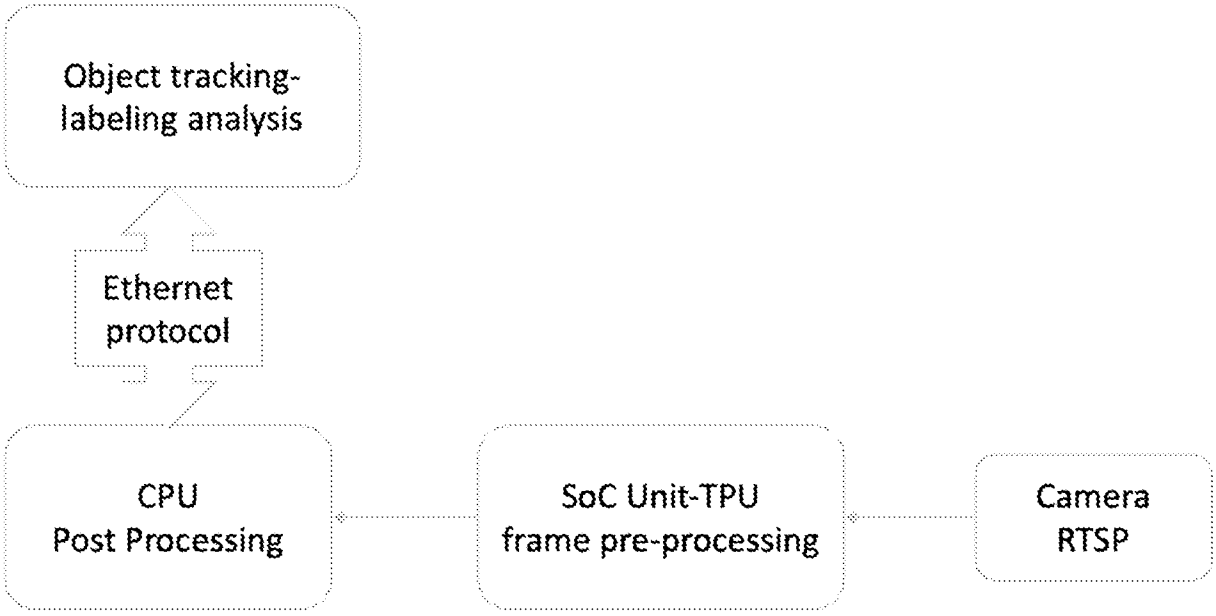


Figure 2. System connection for the integrated AI module

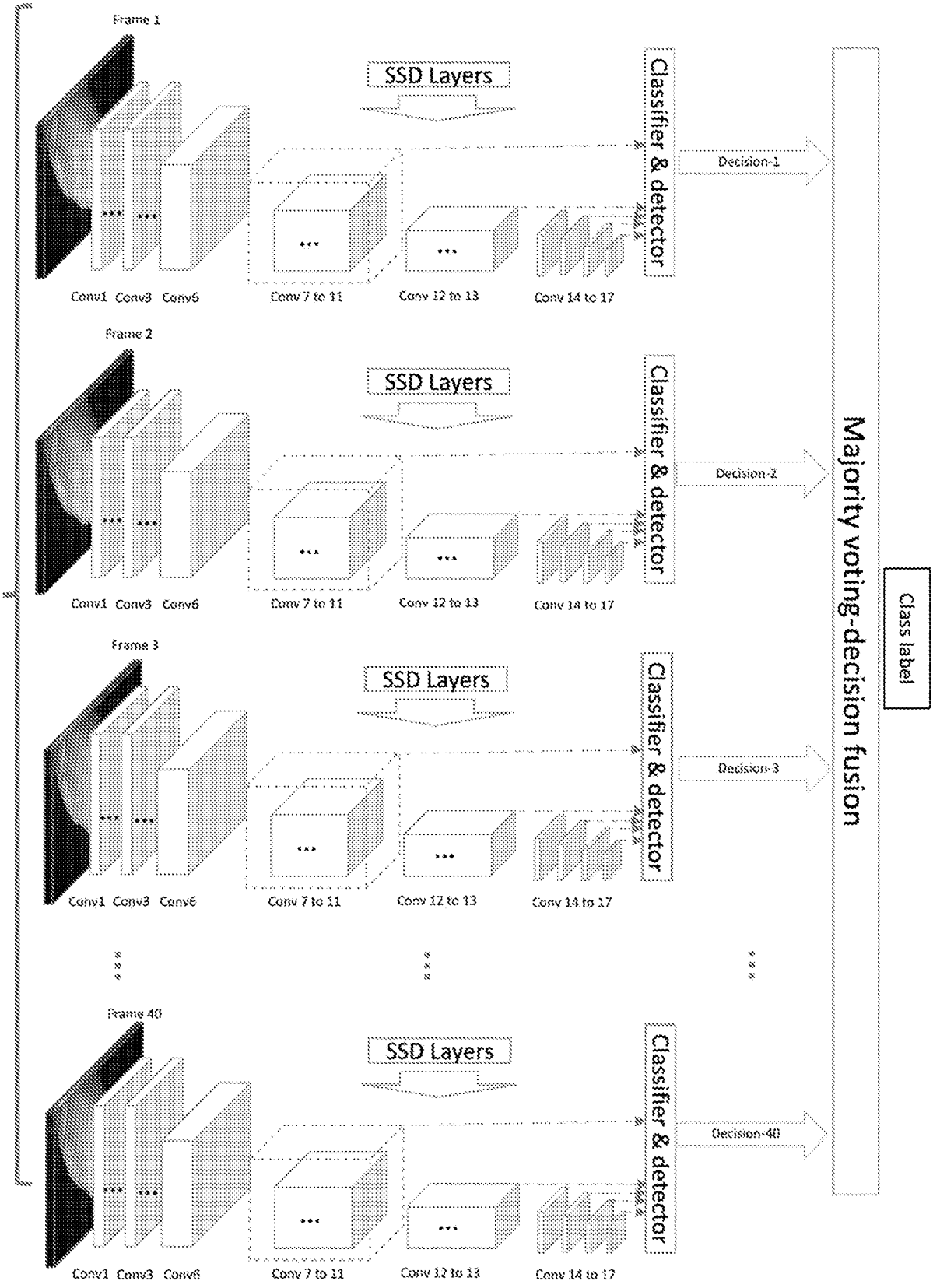


Figure 3 Buffer of decisions, decision fusion in temporal domain

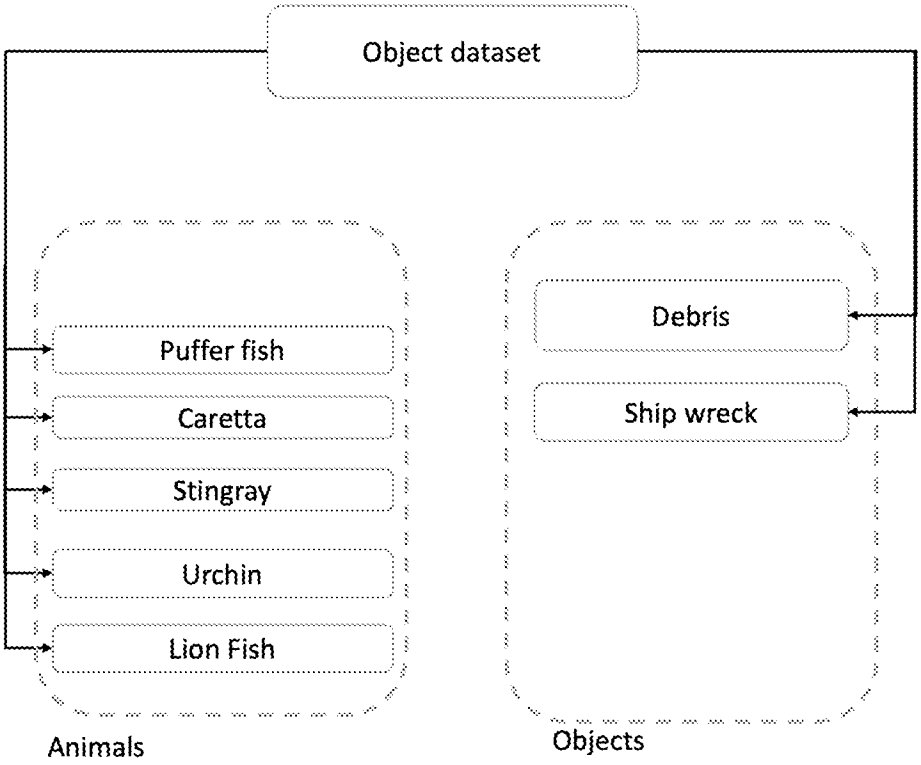


Figure 4. The datasets used for training deep learning architecture for classification

INTERNATIONAL SEARCH REPORT

International application No.
PCT/TR2023/050635

A. CLASSIFICATION OF SUBJECT MATTER

G01S 13/86 (2006.01)i; G01S 17/88 (2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G01S 13/86; G01S 17/88

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPODOC,GOOGLE PATENTS,IEEE,TURKISH NATIONAL PATENT DATABASE

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2021018611 A1 (HEADLIGHT AI LTD [GB]) 21 January 2021 (2021-01-21) paragraphs [0001], [0011]-[0017], [0087]-[0088]	1-18
A	US 2021350713 A1 (TRAVELERS INDEMNITY CO [US]) 11 November 2021 (2021-11-11) Whole document	1-18
A	US 2019138786 A1 (SIGHTLINE INNOVATION INC [CA]) 09 May 2019 (2019-05-09) Whole document	1-18

☐ Further documents are listed in the continuation of Box C.

☑ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"D" document cited by the applicant in the international application

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
19 March 2024	**19 March 2024**

Name and mailing address of the ISA/TR	Authorized officer
Turkish Patent and Trademark Office (Turkpatent) **Hipodrom Caddesi No. 13** **06560 Yenimahalle** **Ankara** **Türkiye**	**Çağrı TOSUN**
Telephone No. **+903123031000** Facsimile No. **+903123031220**	Telephone No. **+903123031661**

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/TR2023/050635

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2021018611	A1	21 January 2021	EP	3769120	A1	27 January 2021
				GB	201804539	D0	02 May 2018
				GB	201903894	D0	08 May 2019
				GB	2573635	A	13 November 2019
				WO	2019180442	A1	26 September 2019
US	2021350713	A1	11 November 2021	US	11710411	B2	25 July 2023
				US	2023316930	A1	05 October 2023
US	2019138786	A1	09 May 2019	NONE			