

DRIVER ATTENTION MODELLING SYSTEM

Technical Field

- 5 The present invention relates, in general terms, to a system for estimating a driver attention map, and a method implemented by such a system.

Background

- 10 The human attention mechanism is an essential and indispensable capability allowing humans to allocate limited cognitive processing resources to simultaneously deal with voluminous visual information. The human attention mechanism allows a human driver to quickly identify important visual cues or potential risks and make corresponding decisions in highly dynamic and complex
15 driving events.

Recently proposed driver attention models aim to replicate human driver attention behaviour during a driving situation. This is crucial in interpreting and designing a human-like driving system. This can further benefit development of
20 human-centric advanced driver assistance systems for supporting human drivers and enhancing driving safety.

Some modern datasets are built around driving events – e.g. lane changes and collisions. Such datasets enable development of deep learning-based
25 approaches for end-to-end driver attention estimation. These methods typically use a pixel-level representation of the visual stimuli, which can replace the classic handcrafted features and descriptors. However, the human visual cognitive system is complex, relating to both low-level visual texture and high-level semantic information. Some methods have been proposed to reflect
30 human visual cognition, but the feature integration is suboptimal since the semantic information is leveraged in an implicit way, where the image is segmented into several areas.

Research focussed on task-driven approaches (the other broad research field
5 being task-free attention determination), particularly for driven attention, aims
to analyse the drivers themselves. These studies estimate where the driver is
looking by detecting the certain driver behaviours including facial feature, head
pose, and body pose. Usually, the driver's visual field is divided into several sub-
fields or targets, hence, the estimation is converted to a classification task.

10

It would be desirable to provide an attention detection or mapping system and
method that overcomes or ameliorates in attention detection performance of
the technologies described above, or at least to provide a useful alternative.

15 **Summary**

Unlike the research methodologies set out above, the present system and
method build a driver attention model that can indicate where the driver would
look at. This avoids the need to obtain the facial features and gestures of a
20 human driver, to augment these features and gestures with the driver's visual
field.

The present methods leverage multi-level visual content, including low-level
texture features, middle-level optical flows, and high-level semantic
25 information, as the model input. Subsequently, a heterogeneous model is
proposed to handle the multi-level input, which integrates graph and
convolutional neural networks. These methods leverage objection detection
information in an interpretable manner.

30 Disclosed is a system for estimating a driver attention map. The system includes
one or more processor (processor(s)) and a memory. The memory stores
instructions that the processor(s) execute to perform the methods described

herein. In this instance, the processor(s) receive a stream of images captured from a camera directed towards a field of view of a driver of a vehicle, and process at least one image of the stream of images using a first neural network to determine an image feature map. The processor(s) also process those
5 image(s) using an object detection model to detect objects in the image(s). A graph representation of the detected objects is then defined, with each detected object being represented as a node of the graph and with connections between the nodes representing relationships between the detected objects. The image feature map and the graph representation of the detected objects can then be
10 integrated to obtain a driver attention map corresponding to the at least one image. In this case, the driver attention map estimates regions in the at least one image corresponding to regions of attention of a driver's visual perception.

Also disclosed is a computer implemented method for estimating a driver
15 attention map. The method involves receiving a stream of images captured from a camera directed towards a field of view of a driver of a vehicle and processing at least one image of the stream of images using a first neural network to determine an image feature map. The image(s) are also processed using an object detection model to detect objects in the at least one image. A graph
20 representation of the detected objects is then defined, where each detected object is represented as a node of the graph and connections between the nodes represent relationships between the detected objects the image feature map and the graph representation of the detected objects are then integrated to obtain a driver attention map corresponding to the at least one image. In this
25 method, the driver attention map estimates region in the at least one image corresponding to regions of attention of a drivers visual perception.

Also disclosed is one or more non-transitory computer-readable storage media storing instructions that when executed by one or more processors cause the
30 one or more processors to perform the methods described herein.

Driver attention map, also referred to as a driver saliency map, is a

representation of estimated regions in an image of the driver's field of view that are focused on by the driver's visual perception or visual attention.

5 The first and second neural network may comprise a convolutional neural network or a convolutional encoder network for processing the raw image data and the optical flow map respectively.

Semantic attention map is a representation of results of object detection superimposed on the source image that served as the basis for object detection.
10 The semantic attention map reflects the relative importance of each detected object for the estimation of the driver attention/saliency map.

Brief description of the drawings

15 Embodiments of the present invention will now be described, by way of non-limiting example, with reference to the drawings in which:

Figure 1 illustrates a method for estimating a driver attention map, in accordance with present teachings;

20

Figure 2 illustrates multi-level visual content;

Figure 3 illustrates the architecture of a system for performing the method of Figure 1;

25

Figure 4 is an example of a graph representation construction of detected objects;

Figure 5 illustrates an example of calculating y_{node_i} of the node i , being
30 equal to the ratio of the sum of the values in the Region of Interest area to the sum of the entire labelled attention map;

Figure 6 is a system for implementing the method of Figure 1;

Figure 7 shows the comparison of different protocols with respect to different backbones on BDD-A dataset;

5

Figure 8 shows the comparison of different protocols with respect to different backbones on DADA-2000 dataset;

Figure 9 shows the comparison of different protocols with respect to
10 different backbones on DR(eye)VE dataset; and

Figure 10 shows the percentage correlation of different values in different datasets, where the correlation indicates the relationship between semantic information and ground truth saliency.

15

Detailed description

The methods described herein leverage multi-level visual content (MVC), including low-level texture features, middle-level optical flows, and high-level
20 semantic information, as the model input. A heterogeneous model is proposed to handle the multi-level input, which integrates graph and convolutional neural networks. These methods leverage objection detection information in an interpretable manner.

25 With reference to Figure 1, a computer implemented method 100 is illustrated, for estimating a driver attention map. The method 100 includes the steps of: receiving a stream of images (102), processing one or more images from the stream of images to determine an image feature map (104), processing the same image(s) to detect objects therein (106), and defining a graph
30 representation of the detected objections (108). Thereafter, the image feature map and graph representation are integrated to produce a driver attention map (110). The driver attention map estimates a region or regions in the processed

image(s) corresponding to a region or regions of attention of a driver's visual perception.

The method of Figure 1 addresses existing issues and improves driver attention
5 map estimation performance, by leveraging multi-level visual content (MVC),
including low-level texture features, middle-level optical flows (in some
embodiments), and high-level semantic information. With reference to Figure
2, steps 104, 106 and 108 recognise that images in the image stream can low-
level texture features (200) and high-level semantic information (202), as the
10 model input. In some embodiments, middle-level optical flows (204) are also
extracted. From these features, a driver attention map 208 is assembled.

Step 102 involves receiving a stream of images. The images will typically be
received from one or more image capture devices (e.g. cameras) directed
15 towards a field of view of the driver of a vehicle. In some embodiments, the
driver's eyes may be tracked and the field of view aligned with the driver's line
of sight. In other embodiments, images may be captured in the general direction
of a driver viewing a field of view through a windscreen.

20 Each image may be a raw RGB image, infrared image or other type of image
from which features can be extracted.

The method, schematically illustrated in Figure 3, estimates a driver attention
map based on a driving scenario image captured in accordance with step 102.
25 MVC is used to formulate the inputs. While embodiments will be described with
reference to optical flow, it will be appreciated that optical flow is not
incorporated into some embodiments.

Step 104 involves processing at least one image of the stream of images using
30 a first neural network to determine an image feature map. In general, multiple
images will be processed. However, in embodiments where object flow detection
(discussed below) is not performed, a single image may be processed.

The image feature map (300 in Figure 3) is derived from low-level textures – these are detectable directly in the raw image. Each texture feature can be used to identify a distinct pixel from its neighbouring pixels in each image. Thus, the
5 image feature map encodes pixel-level features.

Step 106 operates on high-level semantics – namely, object detection. Step 106 involves processing the image or images from which image feature maps were extracted, using an object detection model to detect objects in the at least one
10 image. Object detection is performed on the raw image since the raw image has the most information, hence, it can be used to extract the multi-scale features.

To obtain a high-level semantic relationship, an object detector is utilized. Semantic segmentation does not distinguish between foreground and
15 background. Instead, objects of a particular classification or type (e.g. human, car, light) are segmented from the raw image and classified with equal weight or importance, regardless of the distance of those objects. In contrast, the present method uses object detection. Object detection can be interpreted as introducing prior knowledge and can concentrate on the foreground traffic-
20 related objects and participants, including pedestrians, vehicles, bikes, traffic lights, and so on, while ignoring the background. This type of multi-level information processing agrees more closely with the human visual attention mechanism, which largely disregards distant obstacles that might, if considerably closer, be of primary importance. Object detection is also more
25 robust for detection of the unknown objects and false objects and is faster than segmentation.

Then, the detection results are embedded into the low-dimension feature vectors for reducing the learning and computing costs. This enables a graph
30 representation (304) of the detected objects to be defined per step 108, wherein each detected object is represented as a node of the graph and connections between the nodes represent relationships between the detected objects.

The image feature map 300 and the graph representation 304 of the detected objects are integrated to obtain a driver attention map 306 (also referred to as a driver saliency map) corresponding to the image or images processed under
5 steps 104 and 106.

These multi-level inputs, namely the low-level features, high-level features and, where used, mid-level optical flow features, can increase the interpretability of the model. This can also ensure that the model pays attention to specific objects
10 and areas, such as the pedestrian crossing the road in the raw image (300) in Figure 3.

In some embodiments, the graph representation 304 is transformed prior to integration with the image feature map 300. In such embodiments, the graph
15 representation 304 can be processed using a graph neural network (GNN) 308. The GNN handles high-level semantic extraction by constructing the features of the region of interest (ROI). In particular, the GNN 308 takes Region of Interest (ROI)-wise features (i.e. features corresponding to each ROI) and transforms the graph representation 304 into a two dimensional (2D) Gaussian map for
20 each detected object. In each case, the Gaussian map is produced from an ROI-wise probability for each detected object, the probability being the likelihood that the detected object is actually an object – i.e. something to be avoided during driving. There may therefore be a number of Gaussian maps 310 where the same number of objects are detected.

25 The 2D Gaussian maps 310 are then combined to obtain a semantic attention map 312. Thus, step 110 involves integrating the semantic attention map with the image feature map to obtain the driver attention feature map. By formulating the encoded semantic information as a semantic attention map
30 explicitly during the decoding process (discussed below), the result is more interpretable compared to existing approaches.

With regard to the graph representation, a ROI-wise Graph Attention Network (GAT) is generated. The GAT deals with the detected objects by explicitly constructing the semantic information, rather than handle the features extracted by convolutional modules for building the latent space features.

5 Further, some embodiments employ a semantic attention module to integrate the non-Euclidean output of the graph attention network with the Euclidean feature maps of the convolutional neural networks, the integrated features ultimately being decoded to generate a driver attention map.

10 To leverage the high-level semantic information, the YOLOv3 object detection model is adopted as the object detector to detect various objects and produce the ROI areas. In other embodiments, R-CNN, Region-Based Convolutional Neural Networks, other YOLO versions and other networks may be used in place of YOLOv3.

15

With reference to image 4, the detection results 400 from object detection cannot be represented in a grid-like structure as the image. Consequently, the GNN model is introduced. GNNs are able to dynamically handle data in an irregular domain – e.g. a domain that cannot be represented with a

20 predetermined structure. The different objects in one image lend themselves to representation in a graph structure 402 where an object represents a node – h_i being the node for the i^{th} detected object. Each node of the graph comprises features relating to the corresponding detected object, the features comprise at least location coordinates of the detected object, a class index of the detected
25 object, and a prediction confidence probability. In some embodiments, for each node or object, there exist six features that are defined as follows:

$$h_i = (b_x^i, b_y^i, b_w^i, b_h^i, cls_i, conf_i) \quad (1)$$

30 where (b_x, b_y) represent the central position of the object, b_w, b_h denote the width and height of the detected bounding box, cls indicates the class index of prediction, and $conf$ is the corresponding prediction confidence. In some

embodiments, the features representing each node are normalized separately.

With defined nodes a fully connected graph can be formulated as depicted in Figure 4. In Figure 4, each object in the detection results 400 is bounded by a bounding box, the bounding box being converted to the respective node in the graph structure 402. Notably, each node h is connected to others and itself. Such a structure allows the network to learn the relationship rather than memorizing a predefined one.

- Specifically, with reference to Figures 3 and 4, a graph attention network (GAT) 308, 402 is utilized to deal with an arbitrarily structured graph. This follows the self-attention mechanism for computing the representations of each node. In particular, the graph representation of the detected objects is a fully connected graph structure (e.g. bidirectional edges with other nodes as well as a self-connecting edge), and the representations of each node are computed as follows:

$$\begin{aligned}
 e_{ij} &= f([Wh_i \parallel Wh_j]) \\
 \alpha_{ij} &= \frac{\exp(e_{ij})}{\sum_{k \in N_i} \exp(e_{ik})} \\
 h'_i &= \sigma(\sum_{j \in N_i} \alpha_{ij} Wh_j) \\
 h'_i(K) &= \parallel_{k=1}^K \sigma(\sum_{j \in N_i} \alpha_{ij}^k W^k h_j)
 \end{aligned} \tag{2}$$

- where e_{ij} represents an attention coefficient regarding the neighbour nodes, W denotes a shared and learnable weight matrix, h_i corresponds to node i , h_j corresponds to node j , $f()$ is a mapping function, N_i denotes a neighbourhood of node i , K is a number of multi-head attention, and $\parallel$ indicates the concatenation operation.

GAT 308, 402 adopts a node-wise construction. In this embodiment, there are two GATs, with the output of the first GAT 308 feeding into the second GAT. It can also parallelize the operation of node-neighbour pairs. Furthermore, it is

appropriate for inductive learning tasks. This makes GAT robust with respect to changes in a graph structure.

Leveraging the GAT model, the ROI-wise response probability graph can be obtained. In particular, with reference to Figure 3, the corresponding probability 312 of each node (object) is output from the GAT. The next challenge is therefore determining how to integrate the irregular graph with the grid-like feature map of the raw image and, where used, the optical flow map.

- 10 To achieve this goal, a semantic attention module is designed. The semantic attention module can transform the graph nodes to ROI-wise response maps which can be fused into an attention map – i.e. transform the probability graph to a semantic attention map. Combining the 2D Gaussian maps to obtain a semantic attention map comprises computing:

15

$$G(x, y) = \sum_{i \in N} \vec{h}_i * g_i(x, y) \quad (3)$$

$$g_i(x, y) = \frac{1}{2\pi\epsilon^2 b_w^i b_h^i} \exp\left(-\left(\frac{(x-b_x^i)^2}{2\epsilon^2 b_w^{i2}} + \frac{(y-b_y^i)^2}{2\epsilon^2 b_h^{i2}}\right)\right)$$

where $\vec{h}_i$ represents a final response probability of node i , $(b_x^i, b_y^i, b_w^i, b_h^i)$ are feature of a bounding box of a detected object, and ϵ is a hyper-parameter to regulate the synthetic Gaussian map g_i .

20

The Gaussian maps 310 of all nodes are combined to generate the final semantic attention map G (314). The semantic attention map reflects the importance of each object while aligning with its position and area. Thus, the semantic attention map 314 can introduce high-level semantic information to the model and induce the model to pay attention to specific objects. The size of map G is the same as the feature map that it needs to induce. Through the semantic attention module 316, in which the Gaussian maps 310 and semantic attention map 314 are generated, the irregular graph 308 is transformed into a fixed attention map 314. The fixed attention map 314 can be used to element-wise

25

30

multiply the extracted feature maps similar to the convolutional spatial attention mechanism. With particular regard to Figure 3, integrating the semantic attention map with the image feature map comprises performing an element wise multiplication operation (318) to obtain a multiplication result, decoding
5 the multiplication result using a deconvolution model 320 to obtain the driver attention feature map 322. The semantic attention module 316 is thus critical for constructing the heterogeneous model that integrates GNN and CNN.

With further reference to Figure 3, along with object detection 302, optical flow
10 324 is derived from the raw image, though the two are encoded separately. The encoded features are integrated during the decoding processing discussed below, where the semantic attention module bridges the non-grid output of the GAT module, namely the semantic attention map 314, and feature maps extracted by the CNN model comprising the convolutional encoders for the
15 optical flow 324 and raw image 300.

Owing to the sensitivity of human eye to dynamic (moving) objects, the optical flow is a dense optical flow calculated to obtain the motion information of a given scene. Various mechanisms can be used for optical flow determination,
20 including the Farneback algorithm, gradient based estimation (assuming pixel intensities move from one frame to the next), and Iterative Optical Flow Estimation algorithms. Presently, the Farneback algorithm will be used. The optical flow is divided into a predetermined number of bins, for example 9 bins, according to the optical vector of each pixel to generate a coarse optical flow
25 map and reduce data noise.

To calculate optical flow, pixels and/or features must move from one frame to the next. To that end, the method 100 can involve processing, using a second neural network 326, a subset of the stream of images comprising image or
30 images processed by steps 104 and 106. The subset of images enables pixels and/or features to be tracked between images. The second neural network 326 determines an optical flow map (i.e. 324) that indicates a movement of image

features over the subset of the stream of images. In embodiments where the image feature map 300 and graph representation 308 are used, the image feature map is integrated with the optical flow map and the graph representation of the detected objects, to obtain the driver attention map. Integration may occur in any order – e.g. feature map with optical flow map followed by graph representation, or optical flow map with graph representation followed by feature map, or any other sequential or simultaneous integration.

To handle multi-level inputs, three different models are utilized to encode the input levels independently. For the raw image, a pre-trained CNN-based backbone (i.e. first neural network 328) is adopted to extract its multi-scale features. These multiscale features are also transferred to the decoder 320 as residuals. In this regard, the decoder 320 is a deconvolution model that mirrors the structure of the first neural network, and an output of at least one layer of the first neural network 328 is provided as a residual input to a corresponding mirrored layer of the deconvolution model 320. The decoder 320 has multiple deconvolutional blocks, where each block upscales the feature maps and ensures that they have the same shape as the corresponding feature maps of the encoder. The semantic attention map is positioned after the first deconvolutional block by default. It may alternatively be placed after a different deconvolutional block. The earlier the position is, the more involved the decoding process becomes, thus, the stronger the non-linearity and performance is.

Another CNN model 326 is utilized to obtain the features of the optical flow map 324. These raw features are fused with the corresponding raw image features.

For the semantic information, because the number of detected objects is not fixed, a GNN model 304 is introduced. Furthermore, the relationship between the objects conforms better to a graph format as discussed above. Through the GNN model 304, the corresponding probability of each object can be generated (312).

To integrate the generated output (312) with the extracted feature maps obtained from the CNN models (326, 328), the semantic attention module maps the corresponding probability 312 and position information of each object into a 2D Gaussian map 310. Different ROI-wise Gaussian maps 310 are combined to induce the model decoding. Finally, the deconvolutional model 320 is leveraged to decode the multi-level visual content to estimate the driver attention map, which is described as follows:

$$X' = G_s \otimes (F_r \oplus F_f \oplus X) \quad (4)$$

where X represents the output of one deconvolutional layer, F_r and F_f are the residual connections of the corresponding encoding layers of raw image and optical flow map, G_s denotes the semantic attention map, $\otimes$ and $\oplus$ denote elementwise multiplication and element-wise sum, respectively, and X' is the updated X that is used as the input of the next deconvolutional layer. Thus, step 310, being integration of the image feature map and the optical flow map, comprises an element wise addition operation.

A heterogeneous model is proposed to handle the MVC that includes different types of formats, such as the semantic attention module to integrate the non-Euclidean output of GNN with the Euclidean feature maps extracted by the convolutional neural network (CNN). The proposed heterogeneous architecture includes two CNN-based encoders, one GAT model with a semantic attention module and a deconvolutional-based decoder. There are two outputs: the output of the decoder, the final driver attention map, and the output of the GAT module, the response probability graph. The corresponding loss function is defined as follows:

$$\mathcal{E}_{sal} = \lambda \mathcal{E}_{kl}(\tilde{y}, y_{sal}) + \lambda_2 \mathcal{E}_{cc}(\tilde{y}, y_{sal}) + \lambda_2 \mathcal{E}_{kl}(\vec{h}, y_{node})$$

$$\begin{aligned}
&= \lambda_1 \frac{1}{N} \sum_i y_{sal_i} \log \left(\frac{y_{sal_i}}{\tilde{y}_i + \epsilon} + \epsilon \right) \\
&+ \lambda_2 \frac{1}{N} \sum_i \frac{\sigma(\tilde{y}_i, y_{sal_i})}{\sigma(\tilde{y}_i) \cdot \sigma(y_{sal_i})} \\
&+ \lambda_3 \frac{1}{N} \sum_i y_{node_i} \log \left(\frac{y_{node_i}}{\tilde{h}_i + \epsilon} + \epsilon \right) \tag{5}
\end{aligned}$$

5 where $\tilde{y}_i$ represents the estimated driver attention map and y_{sal} is the
corresponding labeled attention maps, $\sigma()$ is used to calculate the covariance,
 $\mathcal{L}_{kl}$ and $\mathcal{L}_{cc}$ denote the loss functions of Kullback-Leibler Divergence (KL-Div) and
correlation coefficient (CC), respectively, and $\lambda_{1,2,3}$ are used to balance different
loss functions. Notice that y_{node} is the ground truth of the nodes, which is
10 generated as follows:

$$y_{node_i} = \frac{\sum_{(x,y) \in (ROI_i)} y_{sal}(x,y)}{\sum y_{sal}} \tag{6}$$

where ROI_i represents the region that belongs to the bounding box of node i , as
15 shown in Figure 5. It leverages the ratio of the sum of values in the ROI area to
the sum of the entire labelled attention map to obtain the response probability
of the node.

The above methods can be implemented on various systems. One such system
20 is schematically represented in Figure 6. Figure 6 shows an end-user computing
device or system referred to in this disclosure comprises a smartphone device,
a tablet device, a laptop device etc. that is used by an end user to estimate a
driver attention map. Computing device 600 comprises one or more processing
units 610 with access to one or more pre-processors 602 (if used) for pre-
25 processing input data – e.g. selecting raw images, removing motion artefacts,
and so on – and having a communication channel to a camera/external device(s)
604 for collecting input data (e.g. image capture device(s), vehicle monitoring

sensors for speed, yaw and others), a neural network module 606 comprising a first CNN network 608, a second CNN network 610 and GNN module 612, corresponding to CNN 328, 326 and 304, respectively.

- 5 The external device(s) 604 may be integral with, or unitary with, system 600, or may be separate.

System 600 may be in communication (e.g. over network 618) with one or more server system 616 that serve as a back end system for an application executing
10 on the system 600. For example, in embodiments where system 600 is a smartphone, the server system 616 may be a backend application server of a relevant application for CNN implementation on the system 600. The server system 616 may transmit code or information to the system 600 and may receive information from system 600 obtained after pre-processing input data
15 captured by external device(s) 604.

The code running the methodology, and/or input data whether before or after pre-processing, may be stored in memory 620. Thus, memory 620 may constitute or comprise a computer readable storage medium with instructions
20 stored thereon for implementing method 100.

The present method and system were evaluated using various datasets that can also be used (in full or by splitting between training and testing and/or validation data) for training the GNN, first and second CNNs. In particular, the datasets
25 are BDD-A, DR(eye)VE, and DADA2000. The first database includes images for driving events involving braking, the second database includes images at various weather conditions and times of day, and the third dataset includes images leading to crash situations.

30 To evaluate the proposed MVC concept and corresponding MVC-Net, there are four protocols used for comparisons: I, I+F, I+S, I+F+S, which are tested on different datasets with two backbones: ResNet18 and ResNet34, and the other

network architecture parameter can be found in Table I. The I , F , and S represent the raw image, optical flow map, and semantic information, respectively. In the comparison process, the model of I is utilized as the baseline, and $I+F+S$ denotes the proposed MVC-Net – i.e. present method. The decoders of these protocols have the same structure. The comparison results are illustrated in Figures 7 to 9, where each sub-figure represents the comparison of different protocols using two encoders under different evaluation metrics, and the vertical caption indicates the corresponding metric. The results of the error bars in these figures show that the semantic information can slightly reduce the variance of the CC of the prediction, as shown in Figure 9.

Module		Parameter
Convolutional Encoder		Resnet18/Resnet34
Deconvolutional Decoder	deconv_1	$[4 \times 4, 512, 256, 2]$
	deconv_2	$[4 \times 4, 256, 128, 2]$
	deconv_3	$[4 \times 4, 128, 64, 2]$
	deconv_4	$[4 \times 4, 64, 64, 2]$
Graph Module	gat_1	$[6, h]$
	gat_2	$[h, 1]$
Final Fusion	conv_1	$[3 \times 3, 64, 64, 1]$
	conv_2	$[3 \times 3, 64, 1, 1]$

Table I

Both optical flow and semantic information can improve the estimation performance on all three datasets with respect to the primary metric CC, with semantic information outperforming optical flow. High positive CC values occur at locations where the magnitudes the prediction map and ground truth map are similar, thus, CC symmetrically penalizes the false positive and false

negative. From this perspective, the results preliminarily demonstrate the effect and capability of the MVC, which is further validated using other metrics on the BDD-A dataset. However, note that the optical flow does not have a positive correlation with the KL and SIM metrics tested on the DADA2000 and DR(eye)VE datasets. This is because of two factors. The first factor is that the correlation between the movement status of the surrounding agents and ground truth is weak on these two datasets. For example, the DR(eye)VE driver focusing on the vanishing point during routine driving and traffic accident in DADA disrupting the normal usage of the optical flow. To validate this, the correlation between the semantic information and ground truth saliency map of each sample is calculated. Then, a statistical evaluation of the result is performed as depicted in Figure 10, which shows the percentage of different correlation values. The correlation between semantic information and ground truth in the DR(eye)VE dataset is found to be the lowest, followed by the DADA-2000 dataset. The other factor is that the introduced semantic information allows the model estimation to cover a larger area compared to the ground truth. According to the definitions, the NSS measures whether the prediction covers all ground truth points, but the excess area reduces the value of true positive points after dividing the sum of the entire prediction map, which results in a lower SIM value. Therefore, the MVC input is more inclined to expand the attention area, which in higher NSS but lower SIM values. According to the comparison results, the backbones do not exert a significant effort, especially when compared to the various input protocols. This demonstrates the importance of the proposed MVC inputs.

Overall, the comparison verifies the feasibility and efficiency of the proposed MVC-Net.

As set out above, there are three main contributions of the present method and system. Considering human visual mechanisms, the MVC is proposed for driver attention estimation. Rather than implicitly utilizing the semantic segmentation map by existing works, the semantic information is utilized in an explicit and

interpretable manner. In addition, a novel heterogeneous network integrating GNN module with the CNN model is proposed to handle different types of visual content. Also, an innovative semantic attention module is designed to transform the non-Euclidean graph nodes into a semantic attention map, which can be
5 further combined with the Euclidean feature maps extracted by CNN. These features combine to produce a driver attention map with greater likelihood of practically map to a human driver's attention applied to a field of view from a vehicle, than previously known methods.

- 10 The reference in this specification to any prior publication (or information derived from it), or to any matter which is known, is not, and should not be taken as an acknowledgment or admission or any form of suggestion that that prior publication (or information derived from it) or known matter forms part of the common general knowledge in the field of endeavor to which this
15 specification relates.

Throughout this specification and the statements/claims that follow, unless the context requires otherwise, the word "comprise", and variations such as "comprises" and "comprising", will be understood to imply the inclusion of a
20 stated integer or step or group of integers or steps but not the exclusion of any other integer or step or group of integers or steps.

The scope of this disclosure encompasses all changes, substitutions, variations, alterations, and modifications to the example embodiments described or
25 illustrated herein that a person having ordinary skill in the art would comprehend. The scope of this disclosure is not limited to the example embodiments described or illustrated herein. Moreover, although this disclosure describes and illustrates respective embodiments herein as including particular components, elements, feature, functions, operations, or steps, any of these
30 embodiments may include any combination or permutation of any of the components, elements, features, functions, operations, or steps described or illustrated anywhere herein that a person having ordinary skill in the art would

comprehend. Although this disclosure describes or illustrates particular embodiments as providing particular advantages, particular embodiments may provide none, some, or all of these advantages.

Claims

1. A system for estimating a driver attention map, the system comprising:
one or more processor (processor(s));
5 a memory comprising instructions that when executed by the processor(s) cause the processor(s) to:
receive a stream of images captured from a camera directed towards a field of view of a driver of a vehicle;
process at least one image of the stream of images using a first
10 neural network to determine an image feature map;
process the at least one image using an object detection model to detect objects in the at least one image;
define a graph representation of the detected objects, wherein each detected object is represented as a node of the graph and
15 connections between the nodes represent relationships between the detected objects; and
integrate the image feature map and the graph representation of the detected objects to obtain a driver attention map corresponding to the at least one image;
20 wherein the driver attention map estimates regions in the at least one image corresponding to regions of attention of a driver's visual perception.
2. The system of claim 1, wherein integration of the image feature map and
25 the graph representation comprises:

- processing the graph representation of the detected objects by a graph neural network (GNN) to transform the graph representation into a two dimensional (2D) Gaussian map for each detected object;
- 5 combining the 2D Gaussian maps to obtain a semantic attention map;
- integrating the semantic attention map with the image feature map to obtain the driver attention map.
- 10 3. The system of claim 2, wherein integrating the semantic attention map with the image feature map comprises:
- performing element wise multiplication operation to obtain a multiplication result; and
- decoding the multiplication result using a deconvolution model to
- 15 obtain the driver attention map.
4. The system of claim 3, wherein the deconvolution model mirrors the structure of the first neural network; and
- an output of at least one layer of the first neural network is provided as
- 20 a residual input to a corresponding mirrored layer of the deconvolution model.
5. The system of any one of claims 1 to 4, wherein the instructions that when executed by the processor(s) further cause the processor(s) to:

process using a second neural network, a subset of the stream of images comprising the at least one image to determine an optical flow map, the optical flow map indicating a movement of image features over the subset of the stream of images;

5 wherein the image feature map is integrated with the optical flow map and the graph representation of the detected objects.

6. The system of claim 5, wherein integration of the image feature map and the optical flow map comprises element wise addition operation.

10

7. The system of any one of claims 1 to 6, wherein each node of the graph representation comprises features relating to the corresponding detected object, the features comprise: location coordinates of the detected object, a class index of the detected object, and a prediction confidence probability value.

15

8. The system of any one of claims 1 to 7, wherein the graph representation of the detected objects is a fully connected graph structure, and the representations of each node are computed as follows:

$$\begin{aligned}
e_{ij} &= f([Wh_i \parallel Wh_j]) \\
\alpha_{ij} &= \frac{\exp(e_{ij})}{\sum_{k \in \mathcal{N}_i} \exp(e_{ik})} \\
h'_i &= \sigma \left(\sum_{j \in \mathcal{N}_i} \alpha_{ij} Wh_j \right) \\
h'_i(K) &= \parallel_{k=1}^K \sigma \left(\sum_{j \in \mathcal{N}_i} \alpha_{ij}^k W^k h_j \right)
\end{aligned}$$

where e_{ij} represents an attention coefficient regarding the neighbour nodes, W denotes a shared and learnable weight matrix, h_i corresponds to node i , h_j corresponds to node j , $f()$ is a mapping function, $\mathcal{N}_i$ denotes a neighbourhood of node i , K is a number of multi-head attention, and $\parallel$ indicates the concatenation operation.

9. The system of claim 2, wherein the combining the 2D Gaussian maps to obtain a semantic attention map comprises computing:

$$\begin{aligned}
G(x, y) &= \sum_{i \in \mathcal{N}} \vec{h}_i * g_i(x, y) \\
g_i(x, y) &= \frac{1}{2\pi\epsilon^2 b_w^i b_h^i} \exp \left(- \left(\frac{(x - b_x^i)^2}{2\epsilon^2 b_w^{i^2}} + \frac{(y - b_y^i)^2}{2\epsilon^2 b_h^{i^2}} \right) \right)
\end{aligned}$$

where $\vec{h}_i$ represents a final response probability of node i , $(b_x^i, b_y^i, b_w^i, b_h^i)$ are features of a bounding box of a detected object, and ϵ is a hyper-parameter to regulate the synthetic Gaussian map g_i .

10. A computer implemented method for estimating a driver attention map comprising:

- receiving a stream of images captured from a camera directed towards a field of view of a driver of a vehicle;
- processing at least one image of the stream of images using a first neural network to determine an image feature map;
- 5 processing the at least one image using an object detection model to detect objects in the at least one image;
- defining a graph representation of the detected objects, wherein each detected object is represented as a node of the graph and connections between the nodes represent relationships between the detected objects;
- 10 and
- integrating the image feature map and the graph representation of the detected objects to obtain a driver attention map corresponding to the at least one image;
- wherein the driver attention map estimates regions in the at least one
- 15 image corresponding to regions of attention of a drivers visual perception.
11. The method of claim 10, wherein integration of the image feature map and the graph representation comprises:
- 20 processing the graph representation of the detected objects by a graph neural network (GNN) to transform the graph representation into a two dimensional (2D) Gaussian map for each detected object;
- combining the 2D Gaussian maps to obtain a semantic attention map;

integrating the semantic attention map with the image feature map
to obtain the driver attention feature map.

12. The method of claim 11, wherein integrating the semantic attention map
5 with the image feature map comprises:
performing element wise multiplication operation to obtain a
multiplication result; and
decoding the multiplication result using a deconvolution model to
obtain the driver attention feature map.
- 10
13. The method of claim 12, wherein the deconvolution model mirrors the
structure of the first neural network; and
an output of at least one layer of the first neural network is
provided as a residual input to a corresponding mirrored layer of the
15 deconvolution model.
14. The method of any one of claims 10 to 13 further comprising:
processing using a second neural network, a subset of the stream
of images comprising the at least one image to determine an
20 optical flow map, the optical flow map indicating a movement of
image features over the subset of the stream of images;
wherein the image feature map is integrated with the optical flow
map and the graph representation of the detected objects.

15. The method of claim 14, wherein integration of the image feature map and the optical flow map comprises element wise addition operation.
16. The method of any one of claims 10 to 15, wherein each node of the graph comprises features relating to the corresponding detected object, the features comprise: location coordinates of the detected object, a class index of the detected object, and a prediction confidence probability.
17. The method of any one of claims 10 to 16, wherein the graph representation of the detected objects is a fully connected graph structure, and the representations of each node are computed as follows:

$$\begin{aligned}
 e_{ij} &= f([Wh_i \parallel Wh_j]) \\
 \alpha_{ij} &= \frac{\exp(e_{ij})}{\sum_{k \in N_i} \exp(e_{ik})} \\
 h'_i &= \sigma \left(\sum_{j \in N_i} \alpha_{ij} Wh_j \right) \\
 h'_i(K) &= \parallel_{k=1}^K \sigma \left(\sum_{j \in N_i} \alpha_{ij}^k W^k h_j \right)
 \end{aligned}$$

where e_{ij} represents an attention coefficient regarding the neighbour nodes, W denotes a shared and learnable weight matrix, h_i corresponds to node i , h_j corresponds to node j , $f()$ is a mapping function, N_i denotes a neighbourhood of node i , K is a number of multi-head attention, and $\parallel$ indicates the concatenation operation.

18. The method of claim 11, wherein the combining the 2D Gaussian maps to obtain a semantic attention map comprises computing:

$$G(x, y) = \sum_{i \in N} \vec{h}_i * g_i(x, y)$$

$$g_i(x, y) = \frac{1}{2\pi\epsilon^2 b_w^i b_h^i} \exp\left(-\left(\frac{(x - b_x^i)^2}{2\epsilon^2 b_w^{i2}} + \frac{(y - b_y^i)^2}{2\epsilon^2 b_h^{i2}}\right)\right)$$

where $\vec{h}_i$ represents a final response probability of node i , $(b_x^i, b_y^i, b_w^i, b_h^i)$ are

- 5 features of a bounding box of a detected object, and ϵ is a hyper-parameter to regulate the synthetic Gaussian map g_i .

19. One or more non-transitory computer-readable storage media storing instructions that when executed by one or more processors cause the one
10 or more processors to perform the method of any one of claims 10 to 18.