# Semantic-aware Video Compression for Automotive Cameras

Yiting Wang, Pak Hung Chan, Valentina Donzella

***Abstract*—Assisted and automated driving functions in vehicles exploit sensor data to build situational awareness, however, the data amount required by these functions might exceed the bandwidth of current wired vehicle communication networks. Consequently, sensor data reduction, and automotive camera video compression need investigation. However, conventional video compression schemes, such as H.264 and H.265, have been mainly optimised for human vision. In this paper, we propose a semantic-aware (SA) video compression (SAC) framework that compresses separately and simultaneously region-of-interest and region-out-of-interest of automotive camera video frames, before transmitting them to processing unit(s), where the data are used for perception tasks, such as object detection, semantic segmentation, etc. Using our newly proposed technique, the region-of-interest (ROI), encapsulating most of the road stakeholders, retains higher quality using lower compression ratio. The experimental results show that under the same overall compression ratio, our proposed SAC scheme maintains a similar or better image quality, measured accordingly to traditional metrics and to our newly proposed semantic-aware metrics. The newly proposed metrics, namely SA-PSNR, SA-SSIM, and iIoU, give more emphasis to ROI quality, which has an immediate impact on the planning and decisions of assisted and automated driving functions. Using our SA-X264 compression, SA-PSNR and SA-SSIM have an increase of 2.864 and 0.008 respectively compared to traditional H.264, with higher ROI quality and the same compression ratio. Finally, a segmentation-based perception algorithm has been used to compare reconstructed frames, demonstrating a 2.7% mIOU improvement, when using the proposed SAC method versus traditional compression techniques.**

***Index Terms*—Automotive camera data, video compression, semantic segmentation, machine learning, Intelligent Vehicles, Automated and Assisted driving.**

## I. Introduction

VEHICLES equipped with assisted and automated driving (AAD) functions can significantly enhance the transportation system in several countries, increasing the safety of journeys, providing more flexible and comfortable mobility options, and reducing pollution and greenhouse gas emissions [1]. In particular, for higher levels of driving automation, the perception sensor suite is essential for assisting the vehicle decision-making process in AAD systems (L3-L5) [2], [3], [4]. These higher levels of driving automation require more perception sensors to provide the required spatial and temporal coverage, and also to ensure redundancy, robustness and safety [5], [6], [7]. This requirement results in a remarkable increase in the amount of generated data by the perception sensors, especially from the widely deployed camera sensors. The video data generated from cameras can be used by safety critical features or through the cloud for data analysis and decision-making. Both of these use cases involve the need to transmit a large amount of sensor data [8]. Assuming 8 cameras are used in an automated vehicle with the possibility to transmit 200 Megabyte/second each, the required bandwidth can not be supported by traditional vehicle communication networks [9]. Moreover, computational power and wired transmission bandwidth are limited resources on vehicles, and a bandwidth bottleneck caused by too high data rates may result in unacceptable delays, and as a consequence, safety threats to the vehicle and other road stakeholders.

The reduction of camera data, i.e. video compression, needs to be assessed in order to enable the real-time transfer of videos from the sensors to the vehicle processing unit(s), without compromising the perception steps consuming the sensor data (see Fig. 1). Higher levels of compression can reduce the volume of data and latency, but they can also decrease the video quality (e.g. introducing distortions and artefacts in reconstructed frames) that could impair the information extracted from the data and therefore the decision-making process and the vehicle safety. Chan *et al.* have demonstrated that object detection based on compression-tuned neural networks can have improved performance in relation to neural networks trained on uncompressed data (in terms of average precision and recall) when lossy compressed video data are transmitted over the vehicle data communication networks [10], [11]. Recent work by Wang *et al.* has proposed some preliminary results on H.264 compression of a user-defined region-of-interest using video data [12]. However, the amount of work on AAD-specific compression is very limited, and additional research is necessary to fully explore the connection between video compression and machine learning (ML)-based perception for automated driving. In order to increase the compression ratio of automated vehicles (AVs) video data without lowering the quality of the perception algorithms, this work proposes a new semantic-aware compression (SAC) method based on a frame-by-frame identification of the region-of-interest (ROI), preserving higher image quality for this region. This novel semantic video compression splits each frame in the video into two areas: the areas holding less crucial information (such as sky, nature, trees, and foliage) are compressed more heavily than regions containing fundamental information from driving (e.g. the road, vehicles, pedestrians). The selection of important and less important regions can be tuned depending on the specific implementation of our proposed technique.

Manuscript received ** **, ****. Yiting Wang, Pak Hung Chan and Valentina Donzella are with WMG, University of Warwick, CV4 7AL UK (e-mail: Yiting.Wang.1@warwick.ac.uk, pak.chan.1@warwick.ac.uk, v.donzella@warwick.ac.uk).

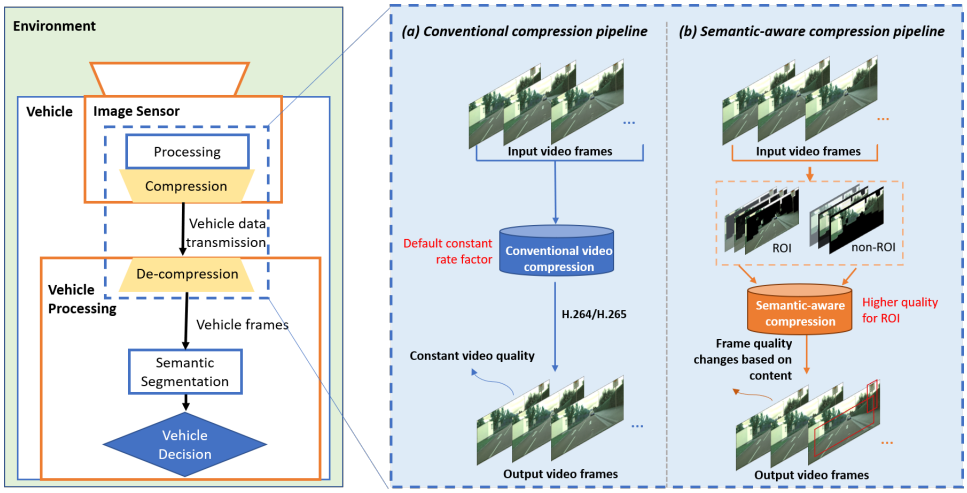


Fig. 1. Schematic view of the architecture of a vehicle with a camera sensor transmitting compressed data (via vehicle wired communication networks) to the processing unit(s) responsible for data consumption and perception. On the right, there is a zoom-in into the functions to show a comparison between traditional (a) vs our proposed semantic-aware video compression (b).

Specifically, as a part of this work, ML-based pre-compression semantic segmentation is used to identify ROI and enable sufficient flexibility to cope with the complexity and variability one can have between consecutive frames in automotive [13]. Following the settings from [12], ROI is identified to include the following labels: road, object, human, parking, vehicle, unlabeled, dynamic, ground; construction/structures, nature, sky are regarded as non-ROI; compression to the different regions is applied using H.264 and H.265 compliant codecs. The contributions of this work are as follows:

- we introduce a new semantic-aware video compression method for automotive cameras using segmentation to extract dynamically, real-time, ROI and non-ROI for each video frame;
- we apply different compression ratios to these two regions, therefore better maintaining the image quality of the ROI, which is more important for imminent navigation decisions;
- we propose three new metrics for quantifying the quality of reconstructed frames (i.e. semantic- aware (SA-) Structural Similarity Index (SA-SSIM), semantic- aware Peak Signal to Noise Ratio (SA-PSNR), and iIOU) that weights image quality based on region importance. The proposed metrics give more emphasis to regions that are critical to driving (ROI), so they can provide a more meaningful evaluation of the quality of images specifically for automotive applications.

The experimental work demonstrates that the proposed approach achieves better results with respect to traditional compression in terms of SA-SSIM and SA-PSNR, and state-of-the-art performance even when combined with a perception task, i.e. semantic segmentation.

## II. Related Work

### A. Video Compression Standards

To date, H.264 and H.265 are two widely adopted compression standards for videos used for digital television and streaming [14]. These standards consider data redundancy in time and space to decrease the size of the compressed files. H.265 improves the compression performance significantly by introducing some more flexibility to H.264 (e.g. more macroblock sizes). Moreover, H.265 can achieve up to a 50% reduction in bit rate maintaining the same perceptual video quality of H.264 [15]. These traditional video compression techniques benefit from being established and widely used in a variety of applications. Their effectiveness, however, has not been optimised for usage in video transmission specifically for AV and AAD functions. Some concepts related to rate control in the above mentioned standards are covered in [12], [16]. Recently, newer codecs (e.g. AOMedia Video 1, AV1, first released in 2018) and video compression standards (e.g. H.266, also known as Versatile Video Coding, VVC, first released in 2020), have been developed to enhance compression efficiency [17]. These codecs and standards are more advanced and computationally complex with respect to their predecessors; they can provide higher quality videos for comparable levels of compression, and support high dynamic range videos and 8K resolution. However, the computational complexity of these novel techniques hinders real-time processing, and hardware acceleration solutions are not mature enough (contrary to H.264 and H.265) [18]. Hence, these standards are considered outside the scope of assisted and automated functions and not covered in this paper. Moreover, some standards have been adopted for automotive (e.g. VESA Display Stream Compression, DSC, and VESA Display Compression-M, VDC-M), but they are optimised for displays (and human vision), and not for machine learning based perception, and they offer inadequately low compression ratios, i.e. 3:1-5:1 [11]; therefore they are also deemed outside the main topic of this paper. In this study, well-known compression algorithms (based on the H.264 and H.265 standards) are coupled with a pre-segmentation phase in order to increase compression performance in terms of higher compression ratios and better image quality in the case of automotive applications.

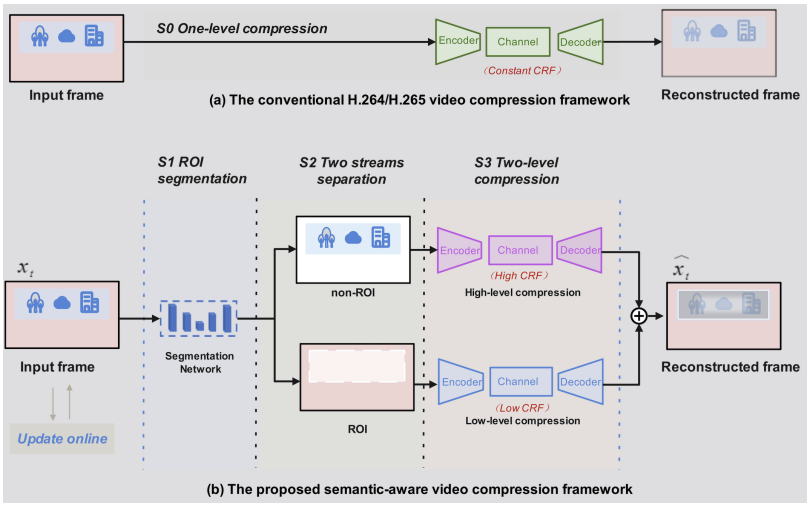


Fig. 2. Illustration of : (a) traditional compression; (b) the proposed semantic-aware video compression framework (SAC). S1 block implements the ROI-based semantic segmentation; S2 is responsible for the separation of the two streams; S3 represents the two-level compression. The constant rate factor (CRF) is used to change the compression level.

### B. Content-aware Compression

The compression strategy proposed in this paper belongs to content-aware compression, which entails compressing the original video frames according to their information content. In Li *et al.* the concept of an "importance map" is proposed to emphasise the sharp edges or rich textures with more bit-rate allocation [19]. Similarly, recent work has proposed a "critical pixel mask" learned from an attention-based convolution neural network (CNN) module to identify the pixels in ROI that are more important for the perceived quality, then a refinement neural network is used to improve the image quality, especially for these masked areas [20]. Another type of image compression involves the preservation of visual information by changing geometric integrity [21]. This process can result in a more "compressible" input image and therefore better compression visual quality. To maximise performance on the given analytic tasks with compression, it has been recently proposed a semantic preserving compression framework where the most relevant features for both classification and compression tasks are jointly learnt with multi-task neural network [22]. By introducing the multi-task loss, the accuracy of the tasks for classification and decoding is boosted. However, most of these proposed methods only compress based on spatial information, and they are not optimised in the case of information redundancy in the temporal domain. Video segmentation is used to differentiate foreground objects from the background in a number of content-aware compression methods. Early investigations use deep learning-based segmentation to classify blocks in the video frame as "texture" or "non-texture" to apply different compression codecs [23]. In Chen *et al.* the block-based compression is improved into pixel-level texture segmentation to have even further bit-rate reduction [24]. Sengar *et al.* proposed a motion segmentation-based video compression method based on the optimisation of adaptive particle swarm for surveillance videos [25]. However, the CNN detector in these methods is likely to detect only one or a few types of textures based on perceptual significance for human vision. Moreover, the foreground extraction used in these papers is only valid for static backgrounds, therefore is not applicable to AVs. Some recent work has proposed to compress road scene maps to solve the lack of large storage in AVs for high-precision localization, but the Authors consider data from the point cloud and not videos [6], [26]. Some content-adaptive video compression techniques for AVs have been recently proposed. However, the work from Dror *et al.* is for remote AV control, whereas Wang *et al.* only presents some preliminary results with semantic-aware compression and H.264 [27], [12].

There are some recent works focusing on end-to-end techniques to implement DNN-based semantic segmentation [28], [29], [30]. However, they do focus on image compression and not video compression as proposed in this work. Furthermore, they do not consider the specific automotive requirements, so they mainly use traditional metrics to evaluate their results. One of the strengths of the techniques proposed in this work is that they combine traditional compression techniques, so they are fully deterministic and allow for higher compression ratios to be achieved (1:1000 and above) with the flexibility of DNN-based segmentation. Moreover, they have a higher degree of flexibility, as depending on the specific application, ROI and non-ROI can be defined differently, and the compression ratio for the two regions can be also tailored.

### C. Semantic Segmentation

Considering the *sense-perceive-plan-control* pipeline in AVs, the quality of transmitted and decompressed sensor data is of foremost importance for the following steps, see Fig.1. Nonetheless, previous work mainly considers traditional image quality metrics, even if some recent works have analysed the implications of compression on object detection [10], [31], [32]. However, there are further vision tasks, e.g. segmentation, that have not been properly investigated. Semantic segmentation means that each pixel in an image is labelled with a specific class (e.g. vehicle, road, sky, etc.) [33]. Many datasets,

such as the KITTI, Cityscapes and ApolloScape, have helped the improvement of the automated driving segmentation task [34]. Building on previous works on segmentation [35], [36], [37], this paper proposes an attention-based semantic segmentation neural network to implement two key tasks: 1) ROI extraction, as the first step of our semantic aware compression; 2) evaluation of the quality of the reconstructed frames. Related to the first task, i.e. ROI extraction, a neural network based solution can offer flexibility and robustness to identify the ROI with high accuracy and to cope in real-time with the significant variability between automotive video frames (e.g. vehicles, pedestrians, bicyclists, animals in different positions, orientations, with different colours, materials, textures, levels of obstruction, etc.).

## III. Model Architecture

Fig.2 shows a comparison of conventional video compression and our SAC architecture. The proposed framework mainly contains three steps (see Fig.2 b). S1 is the semantic segmentation; S2 represents the stream separation (stream I, ROI, and stream N, non-ROI); S3 performs the two-level compression. The initial video frames are described as X = $\{x_1, x_2, ..., x_t\}$, where $x_t$ indicates the frame at time t. The decompressed output stream is $\hat{X} = \{\hat{x_1}, \hat{x_2}, ..., \hat{x_t}\}$.

### A. Overall Procedure

The proposed semantic-aware compression consists of the following steps.

**S1: ROI segmentation.** The masks for both ROI and non-ROI are generated to allow for stream separation in step 2. This step is based on a threshold segmenting neural network, with a similar structure to [38]. The specific segmentation details will be introduced in Sec. III.B.

**S2: Stream separation.** After segmentation, the region boundaries are fitted to a $16 \times 16$ pixel block grid to match the macroblocks in the compression standards (macroblock filter). The masks will be used for generating the streams of ROI and non-ROI. The specific separation details will be introduced in Sec. III.C.

**S3: Two-level compression.** In this step, the ROI stream will go through low-level compression and the non-ROI stream will go through high-level compression. The encoder-decoder structure follows conventional H.264 and H.265 standards. After adding the two compressed streams, the output stream of the compressed video $\widehat{X}$ will be produced. The specific two-level compression details will be introduced in Sec. III.D.

### B. ROI Segmentation

Since the segmentation masks will inform all the process, more accurate segmentation will contribute to improved semantic-aware compression. The segmentation network architecture is described in Fig. 3 [38].

Firstly, the sensor generated video sequences X are converted to lower resolution images and transformed to greyscale through a deep neural network. Secondly, the Residual Network (ResNet) 101 is used as the feature extraction network

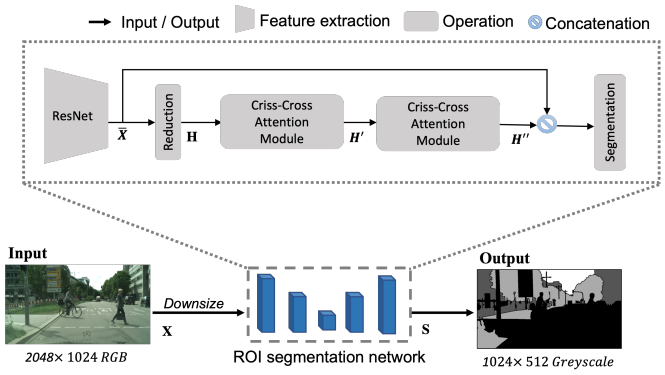


Fig. 3. The architecture of the Segmentation Neural Network, highlighting its different modules, i.e. ResNet, criss-cross modules.

due to its high efficiency [38], [39]. It can obtain the feature map $\bar{X}$ with the spatial size of $h \times w$. Benefiting from its "shortcut connections" structure, this backbone architecture can address degradation problems in deeper convolution layers [39]. Thirdly, two linked criss-cross attention modules are used to get the new feature map $H^{''}$ [38]. Following that, a concatenation of the original feature map $\bar{X}$ with the dense contextual feature $H^{''}$ is performed. Finally, the segmentation results S can be obtained. We set the feature extraction network as *R* and the segmentation network to be *f*, therefore the segmentation result after step one is shown in (1).

$$S = f\left(R\left(X\right) + H''\right) \tag{1}$$

Four tones of grey are used to represent the four output categories of the segmentation pixels: ROI, sky, construction, and nature. As it is hard to define what is "important" (i.e. ROI) and there are constant changes for the stakeholders belonging to ROI, this segmentation algorithm only predicts the non-ROI; all the parts not identified as non-ROI are defined as ROI. This process can produce a more robust 'automotive compliant' segmentation.

### C. Stream Separation

As shown in Fig. 4, given the segmentation results, *S*, a binary mask is generated to further reduce the classification of pixels into two regions: non-ROI and ROI. The macroblocks used in the H.264 and H.265 standards for motion composition and prediction are then simulated using a $16 \times 16$ macroblock filter. Any $16 \times 16$ block that has at least one ROI pixel is treated as an ROI block during filtering; in this way the loss of any ROI pixels belonging to critical objects in the frames is reduced. Using the binary ROI and non-ROI masks ($M_i$, $M_n$), the two streams ($S_i$) and ($S_n$) are generated according to (2) and (3).

$$S_i = M_i \odot X \tag{2}$$

$$S_n = M_n \odot X \tag{3}$$

Where $\odot$ indicates the Hadamard product. The subscripts or prefixes of $i$ and $n$ are used to indicate variables related to the ROI or the non-ROI respectively.

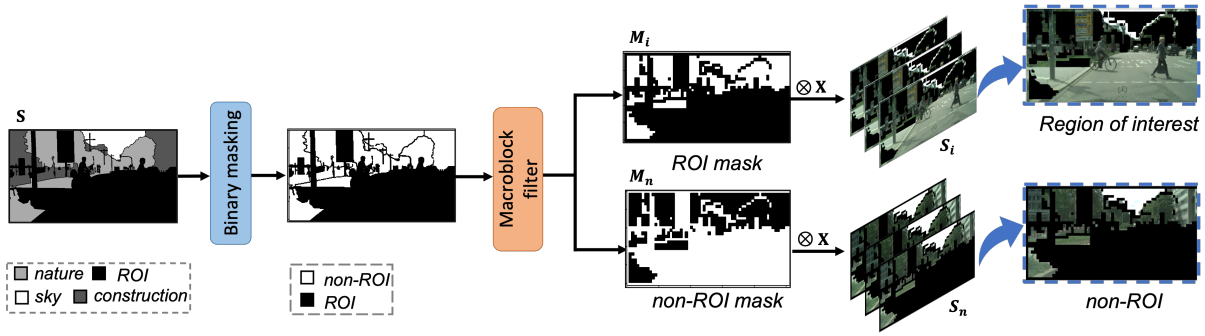


Fig. 4. Separation of ROI and non-ROI streams. Firstly, a binary mask is achieved via pre-processing the segmented frame *S*; then, important and non-important macroblocks are generated to get the two masks, $M_i$ and $M_n$; lastly, the masks are applied to the original frame to generate the two streams.

### D. Two-level Compression-decompression

The frames at the time t from the video sequences $S_n$ and $S_i$ are set as $nm_t$ and $im_t$ respectively, according to (4)-(5).

$$nm_t \subset S_n \ , \ im_t \ \subset \ S_i \tag{4}$$

$$x_t = nm_t \ + \ im_t \tag{5}$$

The process of coding-decoding based on H.264 or H.265 can be divided into seven phases, as described in [12] and schematically represented in Fig. 5. At the end, the predicted ROI frame $\hat{im}_t$ and the reconstructed residual $\overline{r}_t$ are combined to achieve the decompressd ROI area frame $\overline{im}_t$, as shown in (6).

$$\overline{im}_t = \hat{im}_t + \overline{r}_t \tag{6}$$

The non-ROI area frame $\overline{nm}_t$ can be reconstructed in parallel following a similar process. In the end, the reconstructed whole frame $\overline{x}_t$ is generated through the equation (7).

$$\overline{x}_t = \overline{im}_t + \overline{nm}_t \tag{7}$$

The novelty of the proposed technique lies in the quantisation step. In fact, in this work, the constant rate factor (CRF) parameter can be set differently for the two streams, to obtain various quality levels for ROI and non-ROI. The key feature of the proposed SAC compression is that the CRF of the stream $S_n$ is greater than the stream $S_i$, and therefore the quality of $\overline{im}_t$ (reconstructed ROI) is better than the $\overline{nm}_t$ (reconstructed non-ROI) one.

## IV. Implementation

The experiments are conducted under the following assumptions based on the labels from the Cityscapes dataset: 1) the non-ROI area is achieved by combining three classes: construction (i.e. building, wall, fence, guard rail, bridge, tunnel), nature (i.e. vegetation, terrain) and sky; 2) the ROI area is the area of the frame not classified as non-ROI. Based on the previously described network architecture, we have designed and evaluated our experiments as detailed below.

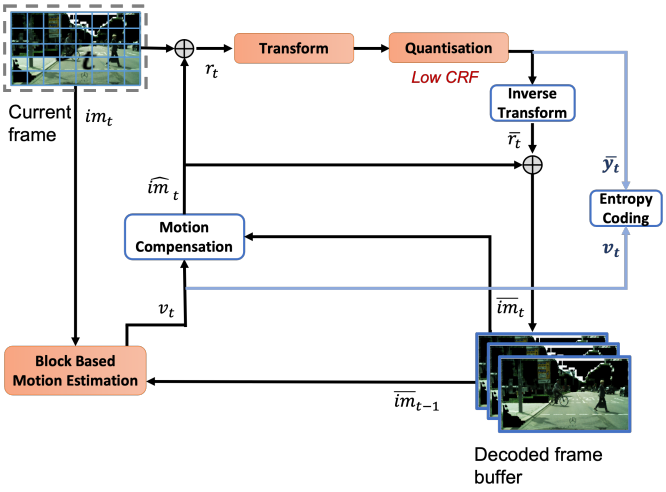


Fig. 5. The process of H.264 or H.265 compression on ROI. An identical process is applied in parallel (at the same time) to the non-ROI; in this case the quantisation will have higher CRF (higher compression).

### A. Training Optimisation

The optimisation method for gradient descend in step 1 is the Adaptive Moment Estimation (Adam) [40]. It is computationally efficient with low memory requirements. A learning rate of $3 \times 10^{-4}$ was used, the *betas* are selected as (0.9, 0.999), and the weight decay is $1 \times 10^{-5}$. The training process has 47 epochs, 2974 iterations per epoch, and batch size of 4.

### B. Loss Function

The binary cross-entropy loss (BCE) and dice loss are combined to create the loss function, hereby called "BCE-Dice loss" [12], [41], [42]. This compounded loss allows for compensating unbalanced classes and has good perfomance with small objects [12]. Assuming that the distribution of the classes is $s$ and the prediction of this distribution is $\hat{s}$, one can use (8) to define the BCE loss.

$$L_{BCE}\left(s, \hat{s}\right) = \ -\left(slog\left(\hat{s}\right) + \left(1-s\right) log\left(1-\hat{s}\right)\right) \tag{8}$$

The similarity between the two distributions is computed via the dice coeffcient [42]. It can be used here to calculate the dissimilarity between labelled image $s$ and predicted results

$\hat{p}$, as described in (9).

$$L_{Dice}\left(s,\hat{p}\right) = \ 1-2\left(s\cap\hat{p}\right)+\frac{1}{\left(s\cup\hat{p}\right)}+1 \tag{9}$$

Here, the smooth index 1 is added to avoid the extreme case when $s=\hat{p}=0$. The unique loss function that we employed in our experiments combining BCE loss with dice loss is defined in (10).

$$L_{BCE-Dice} = L_{BCE}\left(s,\hat{s}\right)+\ L_{Dice}\left(s,\hat{p}\right) \tag{10}$$

### C. Evaluation of Compression and Segmentation

Compression distortion can be evaluated by computing traditional quality metrics, namely PSNR and SSIM [43]. Additionally, the compression ratio can give an indication of the overall compression per frame. However, the above-mentioned quality metrics measure the quality of overall the frames, despite the importance of the content of different areas in each frame. Thus, this work provides a unique semantic-aware assessment, leveraging that the two streams have different compression ratios and different importance to the aim of imminent decisions for *planning* and *control* of the vehicle; this SA assessment is inspired by [44]. The novel evaluation metrics, named SA-SSIM and SA-PSNR, emphasise the importance of the quality of the ROI and are used to evaluate the presented experimental results, Sec. V. These newly defined quality indicators take performance and compression ratio into consideration with a new concept of "compression ratio indexes" as the weights for the two regions. The functions will give higher weight to the ROI, as their quality is crucial for the following navigation steps. We define the CRF, SSIM and PSNR values for stream I and N individually as ($C_{roi}$, $C_{non}$), ($S_i$, $S_n$) and ($P_i$, $P_n$). The compression ratio indexes, estimating the ratios between the stream I and stream N compression ($r_{roi}$, $r_{non}$ ), can be calculated as in (11)-(12).

$$r_{roi} = \frac{C_{roi}}{C_{non}+\ C_{roi}} \tag{11}$$

$$r_{non} = \frac{C_{non}}{C_{non}+\ C_{roi}} \tag{12}$$

As the CRF for ROI is always smaller than non-ROI, the ratio will always have $r_{non} > r_{roi}$. The SA-PSNR and SA-SSIM are formulated as the (13)-(14).

$$\text{SA-SSIM} \ = \ r_{non}S_i+\ r_{roi}S_n \tag{13}$$

$$\text{SA-PSNR} \ = \ r_{non}P_i+\ r_{roi}P_n \tag{14}$$

The accuracy of the "Intersection-over-Union" will serve as the measurement metric for ROI segmentation (IOU). The IOU can be expressed as in Eq. (15).

$$IOU = \frac{GT\cap Pre}{GT\cup Pre} \tag{15}$$

Where GT represents the ground truth area, Pre as the predicted area, $GT\cap Pre$ illustrates the intersection region and $GT\cup Pre$ is the union. The average IOU (mIOU) for m categories of labels, can be computed as in (16).

$$mIOU = \frac{\sum_1^m IOU_m}{m} \tag{16}$$

TABLE I
SELECTION OF CRF VALUES FOR TWO-LEVEL COMPRESSION

| Method | CRF default settings |
|---|---|
| H.264 | $I_{crf}$ = $N_{crf}$ = a |
| H.265 | $I_{crf}$ = $N_{crf}$ = b |
| SA-X264 | $I_{crf}$ = a-c, $N_{crf}$ = a+d |
| SA-X265 | $I_{crf}$ = b-e, $N_{crf}$ = b+f |

Note: a,b are integers and between 0-51, c,d,e,f are integers and equal or above zero. We name our SAC methods SA-'codec', for example, SA-X264 is the X264 codec-based semantic compression.

This work also defines a "Region-of-interest IOU" (iIOU) in (17), for the benefit of our task, to evaluate the quality of IoU specifically for the ROI.

$$iIOU = \frac{GT_{roi}\cap Pre_{roi}}{GT_{roi}\cup Pre_{roi}} \tag{17}$$

### D. Experiment Setup

The experiments were performed using a virtual machine running Ubuntu 20 (100GB of storage), a Quadro P5000 GPU serving as the tensor core, and a Conda environment with Python 3.8. The neural networks were trained and tested using PyTorch.

1) Dataset: the main experiments used the Cityscape dataset [45], which is an AV open benchmarking semantic dataset made up of 50 sequences of annotated videos. To the aim of this work, the Cityscape dataset was the only automotive dataset that comprised temporal sequences with segmentation masks (which are key for video compression techniques based on inter-frame prediction). Part of the experiments was also carried out using the KITTI-STEP dataset [46], which contains diverse driving scenarios (e.g. rural, urban). This dataset was designed for segmenting and tracking every pixel (STEP) and contains labels for non-ROI for consecutive frames in the video sequences.

2) Compression: FFmpeg X264 and X265 codecs were used to carry out the experiments, selecting variable bit rate compression based on CRF [12]; the higher the CRF the higher the compression of the output videos, and therefore the lower the video quality.$I_{crf}$ and $N_{crf}$ are the CRF values for stream I and stream N. A possible way of selecting CRF values is summarised in Table I. In this work, the values of $c, d$, and $e, f$ have been selected to obtain the same compression ratios achieved when using X264 with CRF $= a$ and X265 with CRF $= b$. Specifically, in the results reported in the next section, $a = 23$ and $b = 28$ have been selected, as the default CRF values in H.264 and H.265, respectively. These CRF values correspond to compression ratios of around 1:250 and 1:375 respectively [11], and similar compression ratios can be achieved with $c = 18, d = 27$ and $e = 23, f = 32$. Having similar compression ratios (when using the same compression standard) allows for a fairer comparison of the quality of output videos in case of uniform or semantic-aware compression.

## V. RESULTS

After 47 epochs (2974 iterations each) the computed mIOU reaches 87.97% and 80.89% in training and validation, and

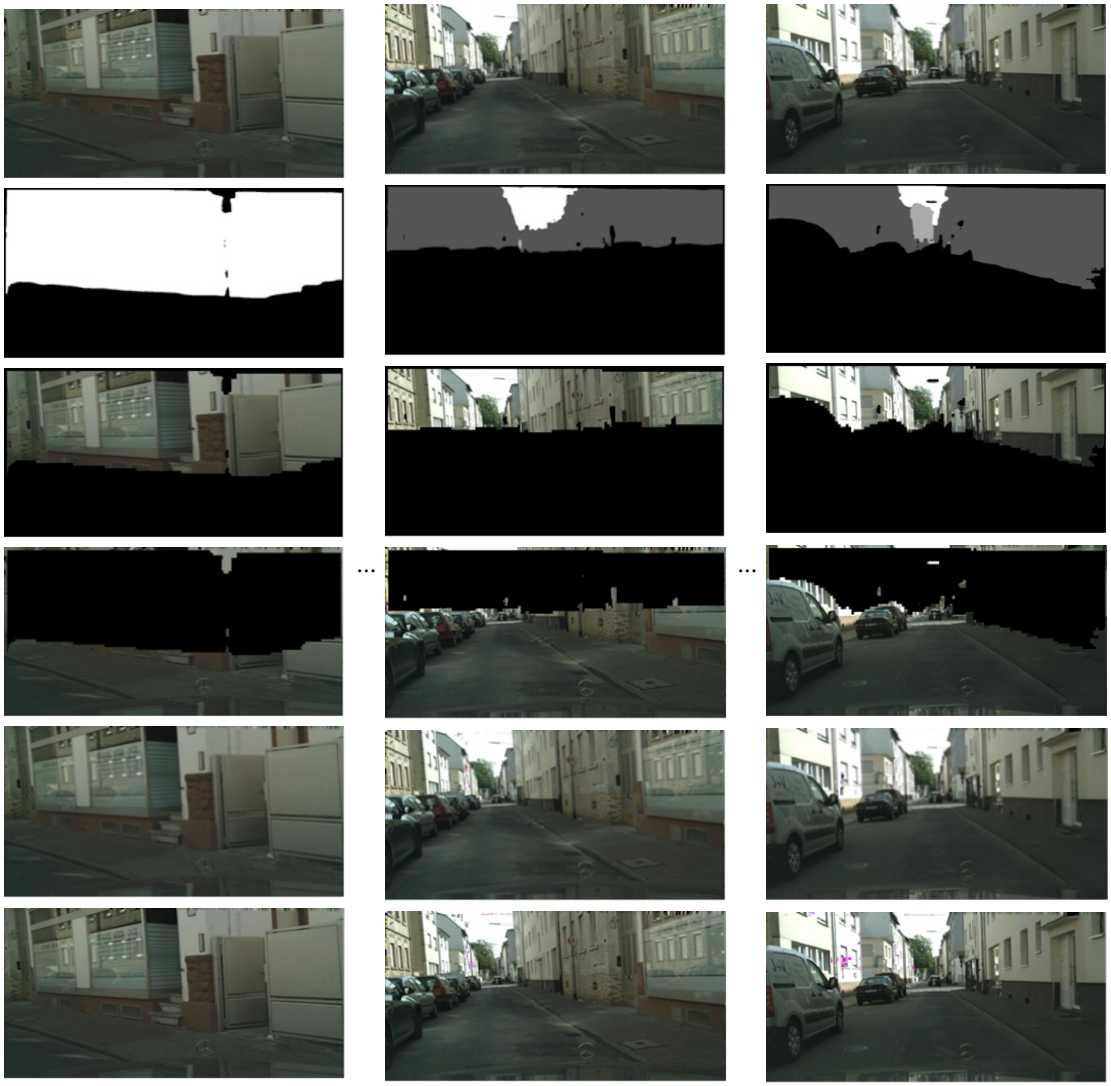


Fig. 6. Video compression visual results. From top to down: original frames, their semantic segmentation prediction, the non-ROI, the ROI, the reconstructed frames by traditional X265, the reconstructed frames by SA-X265. CRF is 23 for the ROI, 28 for the non-ROI.

79.08% for testing. As a first consideration, the proposed SAC method can allow for higher compression ratios while preserving ROI quality, e.g. we can achieve about a 6% reduction in the size of the compressed videos when using SA-X265 compared to traditional X265. This size reduction is achieved by using for the ROI the same CRF used for traditional X265, while the non-ROI is compressed using a higher CRF (i.e. $\text{CRF}_{non-ROI}$= $\text{CRF}_{ROI}$+ 5) for the ROI. A higher size reduction can be achieved further by increasing $\text{CRF}_{non-ROI}$. When using the X265 codec, the average processing time is 0.211 s per frame. In the case of SA-X265, it takes 0.103 - 0.177 s to code/decode the two streams and segmentation takes 0.081 seconds per frame on average. So, without any specific optimisation or hardware acceleration, SA compression has an overall processing time that is comparable to uniform compression (i.e. 0.184-0.258 s versus 0.211 s per frame respectively).

### A. Compression and Artefacts

The experimental results of each step for frames 55, 80 and 170 are displayed in Fig. 6. The second row shows the non-ROI segmentation results. In the third and fourth rows there are the non-ROI and ROI frames, the abrupt boundaries are due to the macro-block filter. The fifth and sixth rows are the reconstructed frames by the conventional H.265 method, and by the proposed SA-X265, respectively.

The semantic-aware video compression shows comparable visual performance and values for SSIM compared with traditional H.264 and H.265 under same compression ratio, Table III. There is a small decrease in PSNR when using the SA techniques (around 1 dB), and this decrease may be due to some artefacts in the reconstructed frames (as shown in Fig. 7), randomly appearing at the boundaries of ROI and non-ROI. As an example, some artefacts have been identified using red rectangles in Fig. 7, with unexpected violet and blue coloured pixels in the reconstructed frames. It has also been observed

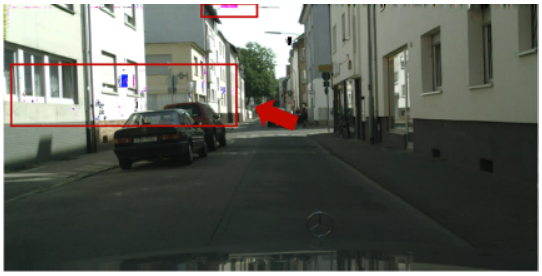

Fig. 7. Flash artefact in one of the reconstructed frames after using our proposed compression scheme SA-X265.

(but not reported here) that with a larger difference between the CRF values deployed for ROI and non-ROI more artefacts are arising at the boundaries. We believe that by applying different compression ratios, these artefacts might be the result of sharp changes at the boundaries. However, as explained in the following sections, this kind of artefacts has little influence on the selected perception task, i.e. segmentation.

To further analyse the image quality in the two different regions (without considering the artefacts), we propose to evaluate the PSNR and SSIM in these different areas separately, as explained in Sec. IV.C.

### B. Image Quality and Semantic-aware Evaluation

Tables II and III present the quality of the reconstructed frames after compression via traditional and newly proposed semantic-aware techniques, namely PSNR, SSIM, and SA-PSNR, SA-SSIM for the Cityscapes and KITTI-STEP datasets respectively. These last two metrics emphasise the quality of the ROI. The selected CRF values are stated in the Table (CRF values for ROI and non-ROI are presented as $C_{roi}$ and $C_{non}$). For the compared compression methods, a clear decreasing trend can be observed for all quality metrics as the value of CRF increases. Moreover, the evaluated video quality of the traditional uniform compression methods has lower SA-SSIM and SA-PSNR than the SAC when achieving the same compression ratio (e.g. H.264 at CRF=23 compared to SA-X264 at $C_{roi}$=18 and $C_{non}$=27, or H.265 at CRF=28 compared to SA-X265 at $C_{roi}$=23 and $C_{non}$=32), and even when the SA-based methods have 'worse' CRF settings (e.g. same CRF for ROI and higher CRF for non-ROI). For example, comparing uniform H.265 compression with CRF = 28 and SAC with $C_{roi}$ and $C_{non}$ equal to 28 and 32 respectively, their PSNR, SSIM and SA-SSIM are comparable, however, the SA compressed files have smaller size and higher SA-PSNR. From the Table, it is also possible to observe that the H.264-based methods mainly perform better in terms of the evaluated metrics, but only because the compression ratio achieved with H.265 at CRF =28 is higher with respect to the one achieved for H.264 at CRF=23. It is worth noting that the mean percentage of the ROI per frame is higher (62%) than of the non-ROI (38%) (obviously these percentages change for each image), so this proportion has an impact on the calculated semantic-aware metrics. From Tables II and III, we can conclude that our proposed semantic-aware compression method can better estimate the quality of a compressed frame, giving more weight to the regions which are crucial for immediate and safe driving. In real-world applications, the CRF values for the ROI and non-ROI can be fine-tuned depending on the specific driving function that will consume the data.

TABLE II
CALCULATED PERFORMANCE METRICS FOR TRADITIONAL VS SA COMPRESSION TECHNIQUES BASED ON H.264 AND H.265 ON THE CITYSCAPES DATASET, WITH CRF SETTINGS. ↑ DENOTES THAT LARGER VALUES LEAD TO BETTER QUALITY.

| Method | $C_{roi}$ | $C_{non}$ | PSNR (dB)↑ | SSIM ↑ | SA-PSNR (dB)↑ | SA-SSIM ↑ |
|---|---|---|---|---|---|---|
| H.264 | 18 | 18 | 46.76 | 0.98 | 46.76 | 0.98 |
| SA-X264 | 18 | 23 | 45.99 | 0.99 | 48.79 | 0.99 |
| H.264 | 23 | 23 | 45.18 | 0.98 | 45.18 | 0.98 |
| SA-X264 | 18 | 27 | 43.94 | 0.98 | 48.05 | 0.99 |
| H.264 | 27 | 27 | 43.69 | 0.98 | 43.70 | 0.98 |
| SA-X264 | 23 | 27 | 44.33 | 0.98 | 47.07 | 0.99 |
| H.265 | 23 | 23 | 45.76 | 0.98 | 45.76 | 0.98 |
| SA-X265 | 23 | 28 | 43.98 | 0.99 | 47.76 | 0.99 |
| H.265 | 28 | 28 | 43.80 | 0.98 | 43.80 | 0.98 |
| SA-X265 | 23 | 32 | 43.77 | 0.98 | 47.01 | 0.99 |
| H.265 | 32 | 32 | 42.01 | 0.97 | 42.01 | 0.97 |
| SA-X265 | 28 | 32 | 43.97 | 0.98 | 45.20 | 0.98 |

TABLE III
CALCULATED PERFORMANCE METRICS FOR TRADITIONAL VS SA COMPRESSION TECHNIQUES BASED ON H.264 AND H.265 ON THE KITTI-STEP DATASET, WITH CRF SETTINGS. ↑ DENOTES THAT LARGER VALUES LEAD TO BETTER QUALITY.

| Method | $C_{roi}$ | $C_{non}$ | PSNR (dB)↑ | SSIM ↑ | SA-PSNR (dB)↑ | SA-SSIM ↑ |
|---|---|---|---|---|---|---|
| H.264 | 18 | 18 | 31.73 | 0.89 | 31.73 | 0.89 |
| SA-X264 | 18 | 23 | 28.62 | 0.86 | 33.60 | 0.93 |
| H.264 | 23 | 23 | 29.19 | 0.84 | 29.19 | 0.84 |
| SA-X264 | 18 | 27 | 27.75 | 0.84 | 32.98 | 0.92 |
| H.264 | 27 | 27 | 27.57 | 0.79 | 27.57 | 0.79 |
| SA-X264 | 23 | 27 | 26.91 | 0.81 | 31.44 | 0.90 |
| H.265 | 23 | 23 | 30.29 | 0.86 | 30.29 | 0.86 |
| SA-X265 | 23 | 28 | 26.76 | 0.80 | 32.17 | 0.91 |
| H.265 | 28 | 28 | 27.57 | 0.79 | 27.57 | 0.79 |
| SA-X265 | 23 | 32 | 25.83 | 0.78 | 31.31 | 0.89 |
| H.265 | 32 | 32 | 25.82 | 0.73 | 25.82 | 0.73 |
| SA-X265 | 28 | 32 | 24.95 | 0.74 | 29.78 | 0.87 |

Fig. 8 compares part of a reconstructed frame after applying the four compression methods. We can see that our proposed methods show better quality in the important area, maintaining clear license plate numbers.

### C. Influence of Compression on Semantic Segmentation

After compression, it is important to evaluate its implications on the perception tasks implemented in the vehicle processing units after the transmission of the data, Fig. 1. Here, we use the same semantic segmentation network model, described in Sec. III.B, to evaluate the videos compressed via traditional and semantic-aware methods. The performance of segmentation are evaluated based on the mean IOU (mIOU) and important area IOU (iIOU), Sec. IV.C, as shown in Table IV. The results demonstrate that decompressed frames with enhanced SA-PSNR and SA-SSIM bring to better segmentation output in terms of both mIOU and iIOU. Our proposed SAC techniques outperform conventional compression in terms of segmentation mIOU, with SA-X264 having the


Fig. 8. Visual comparison of a frame compressed with the 4 different techniques (one per row) with equivalent compression ratio. The plate is clearly readable when using the proposed semantic aware compression techniques.

TABLE IV
EVALUATED SEGMENTATION-BASED PERCEPTION RESULTS FOR TRADITIONAL AND SEMANTIC-AWARE COMPRESSION

| Method | $C_{roi}$ | $C_{non}$ | SA-PSNR (dB)↑ | SA-SSIM ↑ | mIOU (%)↑ | iIOU (%) ↑ |
|---|---|---|---|---|---|---|
| H.264 | 23 | 23 | 45.18 | 0.98 | 87.86 | 92.00 |
| SA-X264 | 18 | 27 | 48.05 | 0.99 | 90.56 | 92.45 |
| H.265 | 28 | 28 | 43.80 | 0.98 | 85.71 | 91.43 |
| SA-X265 | 23 | 27 | 47.01 | 0.99 | 87.48 | 92.04 |

best performance with 90.56% in mIOU and 92.45% in iIOU. Fig. 9 compares visually the segmentation output after the different compression techniques, and shows that, for example, the segmentation of the traffic signal is preserved only when using SA-X265.

## VI. DISCUSSION

Our method presents an improved performance in terms of traditional and *ad hoc* metrics, such as SA-PSNR, SA-SSIM, mIOU and iIOU. For example, the SA-X264 has achieved an increment of 2.864 in SA-PSNR and 0.008 in SA-SSIM when compressed with the same ratio compared with H.264 compliant codec. Moreover, the semantic segmentation on the SAC-based compressed videos performs better than on the data compressed with H.264, with an increase of 2.7% on mIOU and of 0.45% on iIOU. These results have been validated by applying our proposed techniques also on KITTI-STEP dataset (see Sec. IV.B and V.B). The evaluation of image quality and semantic segmentation (quantitatively and qualitatively) shows the same trends of the complete results presented in the paper with the Cityscape dataset. In fact, using KITTI-STEP, the novel SA-X264 shows the best performance, e.g. with CRF values of 18 and 23 for ROI and non-ROI


Fig. 9. Visual comparison of the segmentation results on the original frame (a)-(b), and the compressed frame under same compression ratio, but different compression techniques: traditional X264 (c), traditional X265 (d), proposed SA-264 (e), and proposed SA-265 (f).

respectively, it has the highest SA-PSNR (i.e. 33.598) and SA-SSIM (i.e. 0.933) values, showing an increase of about 1% in terms of iIOU compared with traditional H.264. The results validate the flexibility and wider applicability of our method under different driving scenarios and different datasets. In the future, more efficient segmentation algorithms and codecs may potentially be investigated to enhance the performance and speed of our SAC method. Further research is required on compression artefacts, such as those shown in Fig. 7. Overall, the results show that these minor artefacts are not frequent and the performance of the segmentation based on the decompressed data is not affected by these artefacts. Moreover, our proposed method can leverage the maturity of compression techniques developed to comply with H.264 and H.265 standards, but overall achieves better results than the traditional uniform compression techniques.

## VII. CONCLUSION

This paper proposes a novel automotive specific technique to compress video camera data by pre-segmenting each frame into important area and not-important area, and applying different levels of compression to the two areas, in order to better preserve the quality of safety critical regions. In addition, three new performance metrics have been discussed and applied, as a way to give more weight to the quality of the regions in the frames that are critical for driving applications. Experimental findings indicate that our method outperforms other uniform compression algorithms in terms of our weighted evaluation metrics SA-SSIM and SA-PSNR, which take into account that different regions of the frames are compressed at different levels and therefore have different quality. The results presented in this paper open the possibility for new research in the field of sensor data compression and particularly in how to optmise the compression based on the importance of the data content and in combination with specific perception

tasks, such as object detection, segmentation, etc. Future work will include optimising the proposed technique for bandwidth utilisation and for real-time performance; however, given that the proposed semantic aware compression processes in parallel with the ROI and non-ROI frames (which are easier to compress due to the uniform masked areas), the compression time can be reduced up to almost 50% with respect to uniform compression, therefore compensating the extra time added by the pre-segmentation step. Furthermore, novel low-latency compression techniques and compression on raw data might be considered.

## ACKNOWLEDGMENTS

Dr Donzella thanks the Royal Academy of Engineering as this project was partially supported under the Industrial Fellowships' programme. The work was also partially supported by High Value Manufacturing CATAPULT. This research is partially sponsored by the Centre for Doctoral Training to Advance the Deployment of Future Mobility Technologies (CDT FMT) at the University of Warwick. The Authors acknowledge Dr Huggett, ON Semiconductor, for the interesting and useful conversations.

## REFERENCES

[1] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda, "A survey of autonomous driving: Common practices and emerging technologies," *IEEE access*, vol. 8, pp. 58 443–58 469, 2020.

[2] SAE J3016_202104, "Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," Sociaty of Automotive Engineers, Warrendale (PA), USA, Standard, 2021.

[3] L. Steckhan, W. Spiessl, N. Quetschlich, and K. Bengler, "Beyond sae j3016: New design spaces for human-centered driving automation," in *HCI in Mobility, Transport, and Automotive Syst.* Springer, 2022, pp. 416–434.

[4] G. Cheng, Z. Wang, and J. Y. Zheng, "Modeling weather and illuminations in driving views based on big-video mining," *IEEE Trans. Intell. Veh.*, vol. 3, no. 4, pp. 522–533, 2018.

[5] C.-P. Hsu, B. Li, B. Solano-Rivas, A. R. Gohil, P. H. Chan, A. D. Moore, and V. Donzella, "A review and perspective on optical phased array for automotive lidar," *IEEE J. Sel. Topics Quantum Electron.*, vol. 27, no. 1, pp. 1–16, 2020.

[6] L. Li, M. Yang, H. Li, C. Wang, and B. Wang, "Robust localization for intelligent vehicles based on compressed road scene map in urban environments," *IEEE Trans. Intell. Veh.*, 2022.

[7] Q. Yang, S. Fu, H. Wang, and H. Fang, "Machine-learning-enabled cooperative perception for connected autonomous vehicles: Challenges and opportunities," *IEEE Netw.*, vol. 35, no. 3, pp. 96–101, 2021.

[8] X. Jiang, F. R. Yu, T. Song, and V. C. M. Leung, "Intelligent resource allocation for video analytics in blockchain-enabled internet of autonomous vehicles with edge computing," *IEEE Internet Things J.*, vol. 9, no. 16, pp. 14 260–14 272, 2022.

[9] G. R. Bagwe, "Video frame reduction in autonomous vehicles," 2018.

[10] P. H. Chan, G. Souvalioti, A. Huggett, G. Kirsch, and V. Donzella, "The data conundrum: compression of automotive imaging data and deep neural network based perception," in *London Imag. Meet.*, vol. 2021, no. 1. Society for Imaging Science and Technology, 2021, pp. 78–82.

[11] P. H. Chan, A. Huggett, G. Souvalioti, P. Jennings, and V. Donzella, "Influence of avc and hevc compression on detection of vehicles through faster r-cnn," *Techxiv*, 2022.

[12] Y. Wang, P. H. Chan, and V. Donzella, "A two-stage h.264 based video compression method for automotive cameras," in *2022 IEEE 5th Int. Conf. on Industrial Cyber-Physical Systems*, 2022, pp. 01–06.

[13] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021.

[14] G. Lu, R. Yang, S. Wang, S. Liu, and R. Timofte, "Deep learning for visual data compression," in *in Proc. 29th ACM Int. Conf. on Multimedia*, 2021, pp. 5683–5685.

[15] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, 2012.

[16] Q. Cai, Z. Chen, D. O. Wu, S. Liu, and X. Li, "A novel video coding strategy in hevc for object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 12, pp. 4924–4937, 2021.

[17] B. Bross, Y.-K. Wang, Y. Ye, S. Liu, J. Chen, G. J. Sullivan, and J.-R. Ohm, "Overview of the versatile video coding (vvc) standard and its applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 10, pp. 3736–3764, 2021.

[18] M. Saldanha, M. Corrêa, G. Corrêa, D. Palomino, M. Porto, B. Zatt, and L. Agostini, "An overview of dedicated hardware designs for state-of-the-art av1 and h. 266/vvc video codecs," in *2020 27th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*. IEEE, 2020, pp. 1–4.

[19] M. Li, W. Zuo, S. Gu, D. Zhao, and D. Zhang, "Learning convolutional networks for content-weighted image compression," in *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3214–3223.

[20] X. Zhang and X. Wu, "Attention-guided image compression by deep reconstruction of compressive sensed saliency skeleton," in *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 13 354–13 364.

[21] T. R. Shaham and T. Michaeli, "Deformation aware image compression," in *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2453–2462.

[22] N. Patwa, N. Ahuja, S. Somayazulu, O. Tickoo, S. Varadarajan, and S. Koolagudi, "Semantic-preserving image compression," in *in Proc. IEEE Int. Conf. on Image Process.* IEEE, 2020, pp. 1281–1285.

[23] C. Fu, D. Chen, E. Delp, Z. Liu, and F. Zhu, "Texture segmentation based video compression using convolutional neural networks," *J. Electron. Imaging*, vol. 2018, no. 2, pp. 155–1, 2018.

[24] D. Chen, Q. Chen, and F. Zhu, "Pixel-level texture segmentation based av1 video compression," in *in Proc. IEEE Conf. IEEE Int. Conf. on Acoustics, Speech and Signal Process.* IEEE, 2019, pp. 1622–1626.

[25] S. S. Sengar and S. Mukhopadhyay, "Motion segmentation-based surveillance video compression using adaptive particle swarm optimization," *Neural Comput. Appli.*, vol. 32, no. 15, pp. 11 443–11 457, 2020.

[26] X. Sun, M. Wang, J. Du, Y. Sun, S. S. Cheng, and W. Xie, "A task-driven scene-aware lidar point cloud coding framework for autonomous vehicles," *IEEE Trans. Industr. Inform.*, pp. 1–11, 2022.

[27] I. Dror, R. Birman, and O. Hadar, "Content adaptive video compression for autonomous vehicle remote driving," in *Applications of Digital Image Process. XLIV*, vol. 11842. Int. Society for Optics and Photonics, 2021, p. 118420Q.

[28] J. Wang, Y. Duan, X. Tao, M. Xu, and J. Lu, "Semantic perceptual image compression with a laplacian pyramid of convolutional networks," *IEEE Trans. on Image Proc.*, vol. 30, pp. 4225–4237, 2021.

[29] K. Liu, D. Liu, L. Li, N. Yan, and H. Li, "Semantics-to-signal scalable image compression with learned revertible representations," *Int. J. Comput. Vis.*, vol. 129, no. 9, pp. 2605–2621, 2021.

[30] R. Wang, Z. Sun, and S.-i. Kamata, "Adaptive image compression using gan based semantic-perceptual residual compensation," in *in Proc. 25th Int. Conf. on Pattern Recognit.* IEEE, 2021, pp. 9030–9037.

[31] W. Xu, N. Souly, and P. P. Brahma, "Reliability of gan generated data to train and validate perception systems for autonomous vehicles." in *in Proc. IEEE Workshop on Appl. of Computer Vision*, 2021, pp. 171–180.

[32] U. Michieli, M. Biasetton, G. Agresti, and P. Zanuttigh, "Adversarial learning and self-teaching techniques for domain adaptation in semantic segmentation," *IEEE Trans. Intell. Veh.*, vol. 5, no. 3, pp. 508–518, 2020.

[33] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[34] Y. Kang, H. Yin, and C. Berger, "Test your self-driving algorithm: An overview of publicly available driving datasets and virtual testing environments," *IEEE Trans. Intell. Veh.*, vol. 4, no. 2, pp. 171–185, 2019.

[35] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Int. Conf. on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[36] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2881–2890.

[37] D. Nilsson and C. Sminchisescu, "Semantic video segmentation by gated recurrent flow propagation," in *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6819–6828.
[38] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "Ccnet: Criss-cross attention for semantic segmentation," in *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 603–612.
[39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[40] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.
[41] Z. Zhang and M. R. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *In Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018.
[42] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *In Proc. 4th Int. Conf. on 3D Vision*. IEEE, 2016, pp. 565–571.
[43] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.
[44] L. Wu, K. Huang, H. Shen, and L. Gao, "Foreground-background parallel compression with residual encoding for surveillance video," *IEEE Trans. Circuits Syst. Video Technol.*, 2020.
[45] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3213–3223.
[46] M. Weber, J. Xie, M. Collins, Y. Zhu, P. Voigtlaender, H. Adam, B. Green, A. Geiger, B. Leibe, D. Cremers, A. Osep, L. Leal-Taixe, and L.-C. Chen, "Step: Segmenting and tracking every pixel," in *Neural Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks*, 2021.