Impact of Video Compression Artifacts on Fisheye Camera Visual Perception Tasks

Madhumitha Sakthi[1], Louis Kerofsky[1], Varun Ravi Kumar[1] and Senthil Yogamani[2]
[1]Qualcomm Technologies, Inc., San Diego, California, U.S.
[2]Automated Driving, QT Technologies Ireland Limited.

Abstract

Autonomous driving systems require extensive data collection schemes to cover the diverse scenarios needed for building a robust and safe system. The data volumes are in the order of Exabytes and have to be stored for a long period of time (i.e., more than 10 years of the vehicle's life cycle). Lossless compression doesn't provide sufficient compression ratios, hence, lossy video compression has been explored. It is essential to prove that lossy video compression artifacts do not impact the performance of the perception algorithms. However, there is limited work in this area to provide a solid conclusion. In particular, there is no such work for fisheye cameras, which have high radial distortion and where compression may have higher artifacts. Fisheye cameras are commonly used in automotive systems for 3D object detection task. In this work, we provide the first analysis of the impact of standard video compression codecs on wide FOV fisheye camera images. We demonstrate that the achievable compression with negligible impact depends on the dataset and temporal prediction of the video codec. We propose a radial distortion-aware zonal metric to evaluate the performance of artifacts in fisheye images. In addition, we present a novel method for estimating affine mode parameters of the latest VVC codec, and suggest some areas for improvement in video codecs for the application to fisheye imagery.

1. Introduction

In the recent years, autonomous vehicles are equipped with low-cost camera sensors that provide rich semantic information about the surrounding environment. In order to train robust deep learning algorithms that use camera data for perception tasks, training data is often collected across multiple vehicles and environmental conditions. This has led to a surge in camera data, and associated storage costs, which requires efficient and robust compression strategies. Autonomous driving systems also use other sensors like Lidar but its volume is relatively small due to its sparsity [16].

Prior works [2, 7, 9, 10, 18] have shown the impact of video coding standards such as AVC [32] and HEVC [27] on deep learning tasks. In [2], the authors showed that HEVC and AVC data compression at Quantization Parameter (QP) less than 29 does not significantly affect the Faster R-CNN performance. Actually, they even showed that retraining the model with compressed data improved the Faster R-CNN model precision by 15% compared to the model trained on uncompressed data. However, similar to the other prior works, their tests are limited to undistorted image compression and video input data. Similarly, the authors in [20] tested the impact of image compression across various deep learning tasks such as depth estimation, semantic segmentation, and showed that encoder-decoder architectures were more robust to extreme compression.

The authors [18] applied JPEG compression to the training data and showed negligible drop in performance while fine-tuning with the compressed data. In the case of real-world applications, it is important to train the model directly on compressed images with no prior knowledge about the uncompressed data, such that video compression techniques can be scaled for real-world storage applications. In another study that applied JPEG compression [7], the authors reduced the input image complexity using JPEG, which resulted in similar accuracy with models trained using fewer parameters. Apart from object detection tasks, a recent study [28] evaluated the impact of compression on multi-object tracking accuracy (MOTA) against both Quantization Parameter and Motion Search Range (MSR). They showed significant impact on MOTA at 35 QP, while MSR did not have an impact on the performance.

Although most prior works compressed the image data using HEVC or AVC, the authors in [6] applied the VVC codec and showed that, at specific fine-tuning, the model's weighted average precision increased by 3.68% compared to a model trained on uncompressed data. In addition, data augmentation with JPEG and VVC encoded images also resulted in improved weighted average precision. In case of night vision based pedestrian detection model [9], application of AVC compression to Far Infrared sensor data led to significant storage reduction. The AVC resulted in 0.5 Mbits/s data-rate for negligible loss in performance while JPEG resulted in 1 Mbits/s data-rate. The flexible macro-block segmentation tool of AVC helped in retaining the object details for improved performance, and generated lower data-rates compared to JPEG.

The MPEG Video Coding for Machines (VCM) work [17, 24] differs from the current work in two fundamental ways. First, the data in VCM is all from camera without significant wide angle distortion. Second, the VCM work targets development of a codec with small impact on the visual tasks when the input is compressed. The machine vision model used is developed presumably on uncompressed data. In a primary use case of training data storage, the role of compression is fundamentally altered. The data collected and used for training is compressed while the application of the trained model is generally on uncompressed data. Thus, the impact of compression on training data is of vital concern. Additionally due to the reversal of roles of compression, extremely high image quality may be desired in applying compressed data to the training process when application will be on uncompressed data.

Most of the video compression techniques are tailored for human viewing and they are often applied only on undistorted images with a narrow FOV without modifying the underlying codec which is specifically designed for undistorted images. However, automotive camera suite has very wide angle cameras with high radial distortion due to the needs of a large horizontal field of view of 190° for near-field perception use cases. Four such wide angle fisheye cameras placed around the vehicle cover the full 360° field of view around the vehicle and form the basic sensor set in automotive systems for near-field sensing in combination with Ultrasonics sensors [13, 19]. Relatively, fisheye camera perception has fewer literature as there are only a few public datasets. The limited available literature in various fisheye perception tasks such as object detection [22, 33], semantic segmentation [21, 25], depth estimation [11, 12], localization [29], soiling and weather detection [4, 30], motion segmentation [15], multi-task learning [14, 26], and near-field perception systems [3, 5] indicate that special attention and radial distortion aware design is necessary.

To the best of our knowledge, given the lack of literature in understanding the impact of fisheye image compression on visual perception tasks, our main contributions are:

- The impact of lossy compression of fisheye data on the object detection computer vision task is analyzed across various codecs. Our results show the highest compression that can be achieved without degrading the object detection performance on temporal and non-temporal datasets.
- We emphasise on the necessity to apply lossy compression to the training data, and show the impact of fisheye compression on the object detection task while the model has no prior knowledge about the uncompressed dataset.
- Due to the high radial distortion in the image, unlike prior works that focused on full frame mAP to understand the impact of compression on undistorted images, we propose a radial distortion-aware zonal metric to analyze the impact of fisheye image compression.
- Finally, we present a novel method to improve the existing VVC codec by adapting the camera motion model for the wide FOV camera (fisheye).

Therefore, extending the work of Chan et al. [2] to wide FOV images, we are the first to apply the standard video compression codecs(HEVC, AVC) on wide FOV, fisheye images [8, 34] to quantify both impact of lossy compression on Deep Learning model inference and training. Since standard compression codecs(HEVC, AVC) are designed for human visualization and undistorted images with the exception of VVC[1] that includes a general motion compensated prediction tools that can be applied to common wide FOV images, we present an improved motion model for VVC encoder using camera motion, intrinsics and extrinsics data.

2. Video Compression of Wide FOV imagery

This work studies the use of standardized video codecs on wide FOV imagery. Video codecs utilize temporal prediction from one frame to another and can be greatly effective when the motion of the content matches the motion model of the video codec as the video encoder only needs to encode the motion compensated residual signal. In the case where the camera does not move, as in video surveillance, the background does not change, and hence, a video codec can successfully avoid repetition of unchanged data regardless of the complexity of the codec's motion model. The major components of our evaluation are illustrated in Figure 1. First, original RGB images are converted to YUV color space and sub-sampled to 4:2:0 chroma format as this is efficiently handled by typical video codecs. Second, the images are provided to a lossy video encoder. Several video encoder algorithms and fixed QP values are used to produce compressed bitstreams. Then, the bitstream is decoded and converted to modified YUV' and RGB' pixel values. The RGB' images are provided as input to a Vision Task. The Vision Task may be inference using an already trained model or may consist of training a model on the reconstructed images from the lossy codec.

2.1. Lossy Video Compression impact on inference mAP

In order to understand the effect of the video compression artifacts on the inference mAP, it is necessary to evaluate the images on a standard deep learning network. Given the prevalence of object detection networks in the vision community and the performance vs. computation trade-offs of-

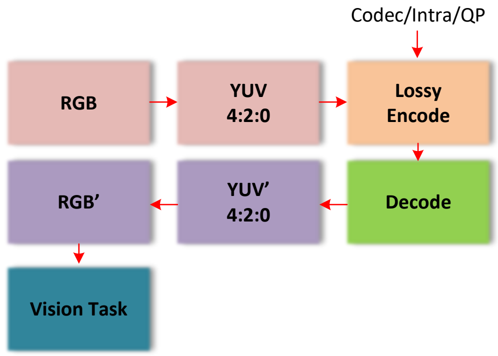


Figure 1. Video compression and evaluation system

fered by the single stage YOLO object detection network, we chose the YOLOv7 object detection network. We applied the AVC and HEVC codecs on the inference images and considered the set of images per camera as a video sequence and encoded the data at various Quantization Parameters (QPs). The frames were compressed using ffmpeg and the decoded images were stored in png, lossless format. AVC and HEVC takes a minimum QP of 0 and a maximum QP of 51. A QP of 4 results in virtually lossless compression and QP of 51 leads to extreme compression.

We test three different encoding methods using ffmpeg. “HEVC intra” uses the HEVC main profile but forces all frames to be intra-coded and hence no temporal prediction. “HEVC main” directly uses the main profile of HEVC for encoding. “AVC” uses the main profile of AVC that allows temporal prediction but is a less capable codec than HEVC. In all cases, ffmpeg defaults of other encoding parameters and complexity settings were used.

2.2. Lossy Video Compression impact on training

Typically, for most deep learning models, few hundred thousand frames of training data is required in order to achieve robust performance. By applying the codecs on inference data, the QP at which the mAP is least affected can be identified. This analysis can partially help in determining the ideal compression such that the inference performance is not affected. In real-world settings, it is not feasible to always store the original data to pre-train the model and then compress the data and use for further fine-tuning either. Therefore, we applied the compression on the training data at different QPs. A model was trained using compressed data (at each QP value) and the inference mAP was recorded on the original data. This analysis helps in identifying the ideal QP to use for compressing and storing only the compressed training data which can lead to valuable storage cost savings.

Zonal Metric: In [23], the authors illustrated the fisheye distortion as a projection of an open cube using the 4th degree radial polynomial distortion model. In the fisheye images, squared grid becomes a curved box towards the periphery and they motivate the need for curved bounding boxes. Therefore, due to the high radial distortion at the periphery, it is important to understand the impact of fisheye image compression particularly at the periphery of an image. We define the zonal metric such that the objects are evaluated in either *central mAP* calculation or *peripheral mAP* calculation. In the FOV of the camera image shown in Figure 2, the straight lines parallel to the y-axis acts as the reference to indicate the curved nature of a straight building and similarly, the straight lines parallel to the x-axis shows the curvature of the windows which would otherwise be straight in a pinhole camera image with no distortion. As the distortion increases towards the periphery of the image, we define the union of the two defined elliptical regions as the least distorted central region while the rest of the area is considered peripheral region. The elliptical region depends on the particular radial distortion of the camera lens. Although to simplify calculations, this region could be approximated by a circle, the ellipse more accurately captures the least distorted regions in the image such as area the close to $y = -H/2$ and $x = 0$ axis.

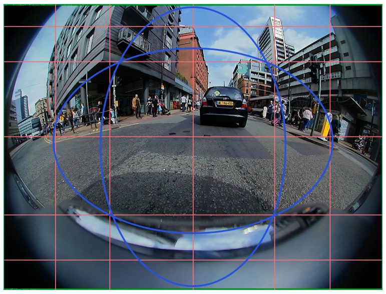
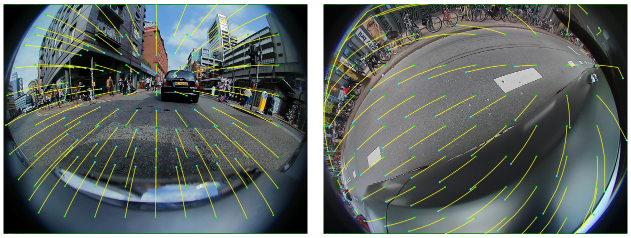


Figure 2. The area with the least distortion is defined as the union of the two ellipses and the objects inside this central region is evaluated in central mAP calculation while the objects outside this region is evaluated as peripheral mAP calculations.

2.3. Improved motion models

Temporal prediction is very effective in reducing the redundancy in video compression where the camera has zero motion. The effectiveness of temporal prediction depends on the underlying motion model of the codec. In situations with significant camera motion, the ability to accurately represent the camera motion becomes important. Traditional codecs use motion models based on 2-D block translation. The recent VVC standard includes an affine mode defined by the motion of control points on the corners of a Coding Unit (CU). Careful analysis of these affine modes indicates the underlying motion is still 4x4 block translation. In the VVC affine modes, a single 2D translation vector for each 4x4 block is determined using a 4-parameter or 6-parameter locally affine model, but the underlying model uses 4x4 block translation. The affine mode provides an ef-



(a) WoodScape FV (b) Woodscape MVR

Figure 3. Epipoles corresponding to ego motion of a camera position over time are shown for Woodscape FV (speed=6.2 m/s, yawrate = -0.6 degrees/sec, dt = 1s), and Woodscape MVR (speed=32.0 m/s, yawrate = 1.3 degrees/sec, dt = 1 s)

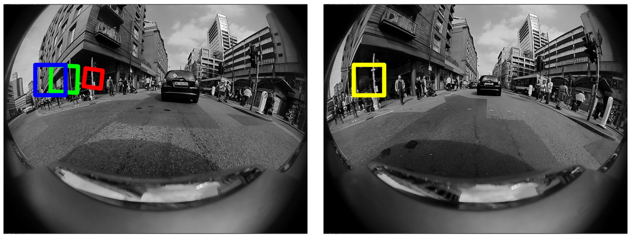
ficient means of signaling a set of translation vectors in a CU composed of 4x4 blocks of pixels.

In datasets with very low frame rate (i.e., where each frame in the scene is one or more seconds apart), motion models in temporal prediction are heavily limited, particularly when considering scale changes and camera lens radial distortion. Therefore, it is important to consider both the capability of the motion model and the practical aspect of selecting parameters for the model. A model with many parameters may provide an excellent motion prediction in theory but it may be impractical to estimate meaningful parameters in practice.

2.3.1 Epipolar geometry guided prediction

Epipolar geometry relates to two overlapping camera views of a scene. In our case, the two views are from the same camera but at different times where the camera has moved position. Given the camera intrinsic, camera extrinsic and camera motion, it is possible to calculate the set of possible pixels in the first frame that correspond to a single pixel in the second frame. With a pinhole camera, this results in a line of possible positions. With a more general camera, the points will lie on a 1-D curve. Examples of epipole curves of the WoodScape dataset are shown in Figure 3 and matched blocks are illustrated in Figure 4. The position on the curve depends on the depth in 3D of the point being imaged. This assumes the scene is static between the two camera images.

Knowledge of camera intrinsic, camera extrinsic, and camera motion greatly reduces the motion parameter estimation problem. Consider a locally affine motion model defined on a block, the motion may be defined by control points, as in VVC with motion vectors (MV), at two or three corners of the block. The number of parameters creates a challenge for motion estimation. We propose to use the epipole geometry to greatly reduce the search space.



(a) Reference ($t-1$) (b) Target (t)

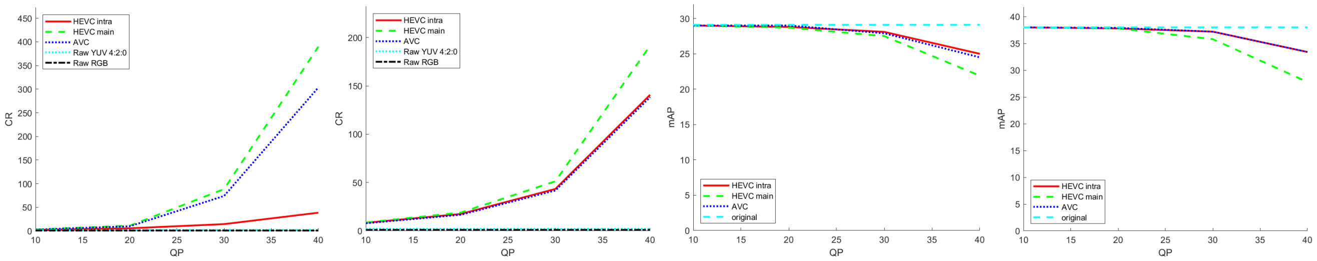
Figure 4. Reference(t) and Target($t+1$) images with overlay of blocks guided by epipole geometry. Block size 128x128 is shown for visual clarity though smaller blocks sizes may be used.

Given a reference and target frame along with the camera and motion information, we select a 1-D list of potential depth candidates. The pixels corresponding to each corner of the block in the target frame at each candidate depth may be pre-computed. For each depth candidate, the corners of the block at a given depth are mapped to a pixel in the reference frame. Given a local block to predict, we form a predicted region on the reference frame by connecting the epipole locations of the corners at the candidate depth. Explicitly, the epipole geometry is used to predetermine the pixel domain displacement of each top corner grid point at a set of candidate depths giving a candidate MV at each grid point and depth $MV_{Epipole}[row][column][depth]$. Given a 1-D list of $n+1$ candidate depth values $\{d_0, d_1, ...d_n\}$, $n+1$ candidate predictors are defined by the VVC motion model and motion vectors corresponding to the block corners in Equation 1. The prediction of a block of pixels P uses the VVC predictor given the current block location (r, c), current block size, reference frame index Idx, and depth d_i, $P_i = VVC(r, c, size, Idx, mv_i[0], mv_i[1], mv_i[2])$.

$$\begin{aligned} mv_i[0] &= MV_{Epipole}[row][column][d_i] \\ mv_i[1] &= MV_{Epipole}[row][column+1][d_i] \\ mv_i[2] &= MV_{Epipole}[row+1][column][d_i] \end{aligned} \tag{1}$$

3. Experiments

In order to evaluate the effect of lossy compression on deep learning models, we compressed the wide FOV images using AVC and HEVC codecs. The Woodscape fisheye camera [34] images were chosen since this dataset consists of scenarios with both ego-vehicle motion and dynamic objects in the scene. Since the publicly available WoodScape dataset is sparsely sampled in time and does not show the true video nature of the camera images that are collected in real-world vehicles, we additionally applied HEVC and AVC compression to the FishEye8K surveillance camera



(a) Fisheye8k

(b) Woodscape

(c) Fisheye8k

(d) Woodscape

Figure 5. (a) and (b): Compression ratio (CR) versus Quantization Parameter (QP) of a video codec applied to YUV 4:2:0 representation of content. (c) and (d): mAP versus QP of lossy video compression for various codecs.

images to evaluate the effectiveness of temporal prediction on the video data.

3.1. Dataset

The WoodScape dataset [34] consists of 10,000 fisheye camera images from 4 surround view cameras: front, rear, mirror left and mirror right. The data was collected across diverse geographical regions including USA, Europe and China. The authors provided 2D bounding box annotations for five classes: pedestrians, vehicles, bicycle, traffic lights and traffic signs. Since the dataset was released as a part of a challenge, the annotations are available only for the training set (8,234 images). Therefore, we split the training data into two sets of 5,762 images and 2,472 images for training and testing by maintaining an equal distribution of surround view cameras in both sets.

The FishEye8K dataset [8] was released with 22 videos (8,000 images) captured across 18 different fisheye cameras for traffic surveillance in Hsinshu, Taiwan. The authors provided 2D bounding box annotations for the entire dataset across Pedestrian, Bike, Car, Bus and Truck classes. Therefore, we used the author's training and validation split for training the YOLOv7 object detection model.

3.2. YOLOv7 object detection model

The YOLO (You Only Look Once) models are single stage object detection networks that predict both bounding boxes and classes. The Non-Maximal Suppression (NMS) post-processing is utilized to finalize the network's predictions. Compared to the latest YOLOR model, YOLOv7[31] achieves 0.4% improved AP while reducing the computations by 15% with 43% fewer parameters. Notably, the authors improved the network's performance by improved model scaling and reparametrization planning techniques.

3.3. Compression ratio results on inference data

An example of the potential of 4:2:0 and video codec is illustrated in Figures 5a, 5b, by the compression ratio achievable on the FishEye8K surveillance dataset and the Woodscape automotive dataset. The chroma sub-sampling of the 4:2:0 format gives a 2:1 compression ratio compared to raw RGB data. The image data is converted to YUV 4:2:0 color space and compressed with ffmpeg using different codecs and intra periods. In all cases, the input consisted of a sequence of frames for a specific camera, and the results show the average compression ratio averaged across cameras for each codec tested at a specific Quantization Parameter (QP). On the FishEye8K data, prediction is effective as can be seen in comparing the HEVC intra (no motion compensation) with the HEVC main that utilises temporal prediction. We see that even the AVC 420 codec, which includes temporal prediction, exceeds the HEVC codec when temporal prediction is removed by forcing all intra frames. Compression ratios over 50:1 can be achieved using video codecs at QP 30 using temporal prediction and the newer HEVC codec but, only about 43:1 using the HEVC-intra in case of the WoodScape dataset. The motion model is not as effective on the Woodscape data due to the camera motion and the large temporal difference between frames (over 1s) even while compressing the same scene. However, the Fisheye8k dataset compression using the temporal prediction results in over 70:1 compression ratio while the all intra configuration results in only 14:1 compression ratio. Therefore, although the WoodScape data consists of repetitive patterns such as road and sky, the compression of the video data with complex Urban scene results in higher compression ratio assuming the motion model efficiently represents the motion in the content.

A central question is the impact of lossy video compression on DL tasks. The first aspect of this evaluation is looking at lossy compression on images used for inference of a model trained without compression. Different video codecs considered have different prediction modes, block size, spatial transforms, etc. (intra coding uses only spatial prediction within a single frame). Despite these differences, the lossy quantization process is similar. We compare the performance of codecs across various QP parameter in Figures 5c and 5d. We see that the difference in mAP between codecs is insignificant for $QP < 20$.

The mAP vs CR results for QP 30 on both Woodscape

Table 1. **The mAP vs CR is captured across various codecs for QP 30 compressed data.**

Dataset	*Codec*	*CR*	*mAP*
Woodscape	Uncompressed	1	38.1
	HEVC-Intra	43.3	37.2
	HEVC-Main	51.2	35.8
	AVC	41.7	37.2
Fisheye8K	Uncompressed	1	29.1
	HEVC-Intra	14.5	28.1
	HEVC-Main	88.6	27.5
	AVC	74.1	27.9

and FishEye8K are reported in Table 1. The benefit of temporal prediction of HEVC-main and AVC can be seen in high compression ratios achieved with small reduction in mAP on the FishEye8K results. Even with a higher compression at QP 30, the drop in mAP is only around 1% with HEVC-intra and AVC codecs. However, at and above QP 30, there is more than 2% drop in mAP with HEVC-main (which may not be desirable for safety critical automotive applications) although HEVC-main consistently achieves better compression ratio then the other codecs/profiles. Therefore, as a trade-off, at QP20, the HEVC-main results in least drop in mAP for the best CR while, above QP30, both AVC and HEVC-intra results in improved trade-off between CR and mAP.

For the Woodscape data, the HEVC main temporal prediction is less effective. Comparing HEVC-Intra and HEVC-Main, the CR is increases slightly from 43.3 to 51.2 while the mAP reduced by 1.4%. The AVC codec also includes temporal prediction and performs worst than the HEVC-Intra codec. In both HEVC-Main and AVC, the motion model is limited to block translation which does not handle the fisheye radial distortion or zooming motion.

3.4. Lossy Video Compression impact on training

Figure 6 shows uncompressed image inference mAP on full frame, central and periphery results on YOLOv7 models trained the Woodscape images compressed using HEVC-main and intra profiles. The model trained and tested on uncompressed images is referenced as the baseline with 37.9% mAP. The HEVC-intra QP 20 compressed trained model has a negligible drop in performance, and the models trained with higher QP compressed data results in lower mAP. The model trained on the original data and evaluated on the QP 40 compressed inference data results in 33.3% mAP. However, the model trained on the QP 30 compressed model and QP 40 compressed inference data results in an increased 34.0% mAP. Therefore, training on compressed data results in the model learning the compression artifacts and helps in recovering the model performance.

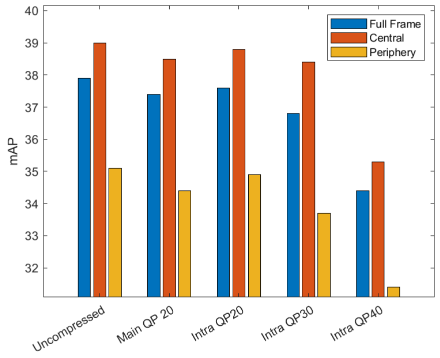


Figure 6. The zonal mAP values for various models trained on the compressed data and evaluated on the uncompressed validation images.

To address the impact of radial distortion, we proposed the union of the elliptical regions as the central region with least distortion and the rest as the peripheral region. However, the region is approximated by a circle for simplifying calculations. Across all the models, the central mAP is better than peripheral mAP. In case of the IntraQP20 model, the central and peripheral mAP almost retains the original uncompressed mAP performance. Therefore, the compression has minimal effect. However at QP30, the central mAP drops only by 0.6% compared to the original model's central mAP but the peripheral mAP drops by 1.4%. Therefore, the compression has a more profound effect on the peripheral region at QP30 and overall mAP drop of 1.1% at QP30 would not completely capture the effect of fisheye image compression and it is necessary to define zonal metric tailored to the camera parameters to identify the trade-off compression ratio.

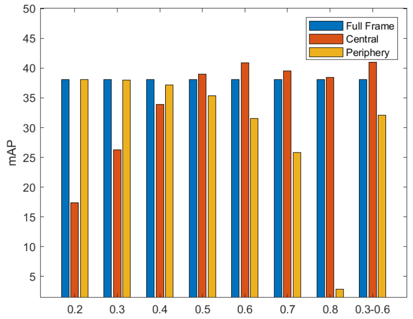


Figure 7. The uncompressed model is evaluated on uncompressed validation images using zonal mAP for different radius values.

As shown in Figure 7, the central mAP increases as the distance increases (i.e., more objects are included in the central region). However, beyond a distance of 0.6, the peripheral mAP starts to drop due to higher distortion on the

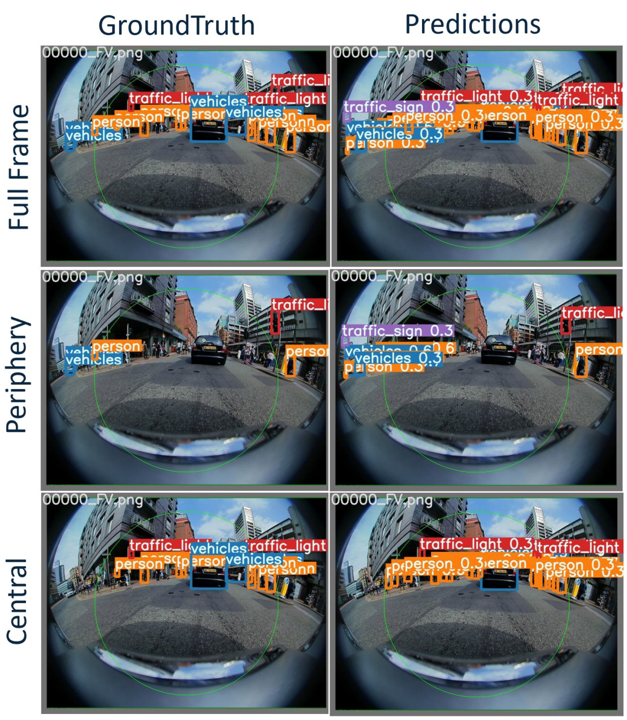


Figure 8. The green circle defines the central regions and any object within the circle is evaluated as a part of central mAP and all the other objects are evaluated as a part of peripheral mAP.

peripheral regions and poorer predictions. At the same time, the central mAP (at lower distances) with very few objects also has a lower mAP, possibly due to the stretched nature of the objects at the center of the image. The rightmost column in Figure 7 only includes the objects between 0.3 and 0.6 and excludes both the stretched part of the image and the periphery. It shows higher mAP compared to other region definitions. Therefore, a more tailored zonal mAP that also considers the central objects stretching could be more informative to decide on the model's performance on the compressed data.

In Figure 8, the objects in the scene are split based on the location into Central or Peripheral regions for the respective mAP calculation. The circle is defined from the center with a radius of 0.5 times the maximum distance (perpendicular distance) and the distortion of the objects at the periphery is significantly higher compared to the central zone. Closer to the peripheral regions in the image, the objects become harder to detect due to distortion along with false positive detection such as the traffic sign prediction.

4. Improving video codec motion models for wide FOV cameras

The frame pairs from the Woodscape fisheye surround view camera images were selected and the epipole geometry was used to guide the target frame prediction based on the reference frame. The result of selecting the optimal value for this 1-D depth list at each block of the reference image provides a prediction image. Table 2 shows the average MSE result using the baseline zero motion model target image prediction and epipole guided motion model prediction of the target image. We tested on frame pairs from the WoodScape dataset across all the four surround view cameras. Due to the random sampling of the available Woodscape data, consecutive frame pairs were not readily available and most pairs had variable vehicle motion between frames. Therefore, given the above limitations and the lack of large scale video fisheye dataset with vehicle motion, we applied our proposed epipole guided search algorithm on the limited frame pairs in the dataset. However, on these challenging, complex Urban scenario images and across surround view cameras, our method resulted in 34.2% MSE reduction while predicting the target image.

Table 2. **The MSE between predicted image and target image using the baseline approach of zero motion model and epipole guided search across camera views.**

Frame	*Baseline*	*Epipole guided*	*MSE change[%]*
WoodScape	3031.4	1994.4	34.2%

An example is shown in Figure 9. The baseline prediction with a zero motion predictor on the front view image 1 results in 2929 MSE (Mean-Squared Error) while the epipole guided prediction results in 1547 MSE. In the second frame pair, due to the lack of shadow and uniform texture of the road surface, the baseline zero motion MSE is 1910 while the epipole guided search results in 1372 MSE. Therefore, especially in case of front camera motion, the epipole guided prediction that takes camera intrinsic, extrinsic, and true motion results in improved prediction. Typically in the video codec, a lower MSE between the reference and the target frame will result in lower bit rate and hence improved compression ratio. In addition, we similarly tested on the mirror view right image (Figure 10) and compared to the baseline with an MSE of 1372. Our guided search results in an MSE of 1269. The MSE reduction with the MVR is less but the image is dominated by road surface and large time difference between the two frames which results in reduced temporal prediction effectiveness.

4.1. Suggestions to improve motion models used in future video codecs

To support cameras with significant lens distortion and camera motion, we need to improve the motion model in order to have accurate temporal prediction. Although the locally affine model of VVC was of interest, but a detailed study indicated that these modes fundamentally rely on tra-

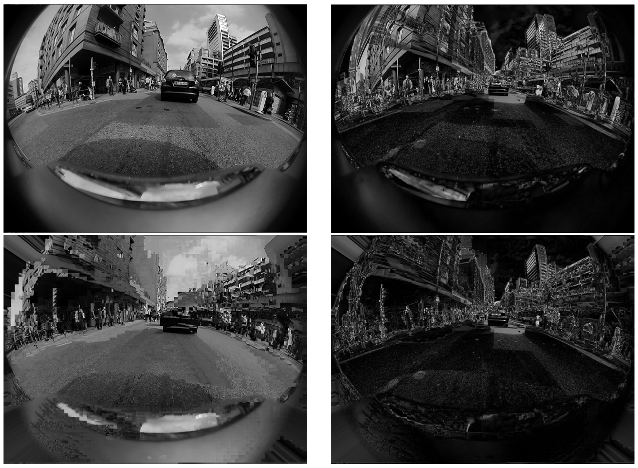


Figure 9. WoodScape FV: Full prediction gray images and errors images corresponding to zero motion predictor and epipole guided locally affine predictor. Top row zero motion predictor with error image (MSE 2939). Second row, epipole guided 16x16 locally affine predictor and error image (MSE 1547).

ditional block translation. Although the next standard development H.267 would include the true affine model, an epipole guided search is still required for improved motion model. Therefore, we suggest the following:

- **True local affine model**: The local adaptivity should be able to support spatially varying camera lens distortion but a true local affine model appears desirable to improve temporal prediction and hence compression.
- **Efficient signaling** of the improved model is desired to avoid overhead in transmitting the motion information. The current VVC syntax for signaling affine mode requires signaling several parameters and multiple motion vectors at each CU and is not expected to be efficient when motion is dominated by affine elements with different parameters such as a zoom due to fast camera motion of a vehicle. Thus, an efficient means of signaling large regions of affine motion is an anticipated need.
- **Epipole guided search**: An efficient means for determining motion parameters, though not officially part of the standard, is essential for the improved motion tool to be useful in practice. The epipole guided search can reduce a 4 or 6 parameter motion search to searching a 1-D list of candidates distances.
- **Dataset**: An important step to move forward with producing a codec for these applications is for the community to develop a dataset with true video motion (15 fps or 30 fps), unlike the WoodScape dataset, and significant camera motion, unlike the FishEye8K dataset.

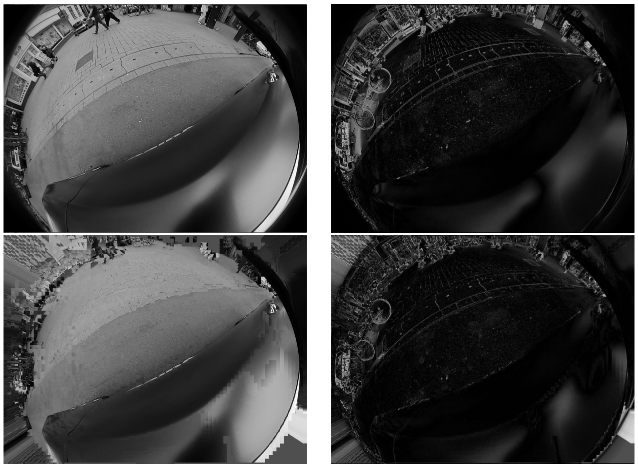


Figure 10. WoodScape MVR: Full prediction gray images and errors images corresponding to zero motion predictor and epipole guided locally affine predictor. Top row zero motion predictor with error image (MSE 1372). Second row, epipole guided 16x16 locally affine predictor and error image (MSE 1269).

5. Conclusion

We presented the study on the impact of lossy fisheye video compression on a camera visual perception task i.e., 2D fisheye object detection. Due to the excessive storage costs involved in saving the automotive driving data, it is important to first understand the extent to which the fisheye data could be compressed without affecting the end task. Our results showed that a minimum of 10x compression ratio is achievable for a negligible drop in mAP and over 80x compression ratio is achievable for a 1-2% drop in mAP for static camera sequences. Although the overall mAP is informative in deciding the ideal compression ratio for pinhole camera models, due to the high distortion at the periphery we present a novel zonal mAP metric. This highlights the effect of compression artifacts on the high distortion regions in the image which ensures that the optimal compression ratio is chosen while the adverse effects of compression artifacts on the performance of the model are avoided. Finally, since video compression achieves remarkable compression rates by exploiting temporal correlations between successive video frames, an accurate motion model is mandatory. The existing codecs rely on a block translation motion models which give sub-optimal temporal prediction with high-speed camera motion and wide-angle camera distortion. Therefore, we present an epipole guided motion prediction model which results in 34% lesser MSE compared to the baseline which translates to lower bitrate requirement for storing and transmitting the compressed data. In the future work, we plan to investigate the impact on a larger set of vision tasks and develop motion models that accounts for both wide FOV and dynamic objects in the scene.

References

[1] Benjamin Bross, Ye-Kui Wang, Yan Ye, Shan Liu, Jianle Chen, Gary J Sullivan, and Jens-Rainer Ohm. Overview of the versatile video coding (vvc) standard and its applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(10):3736–3764, 2021. 2

[2] Pak Hung Chan, Anthony Huggett, Georgina Souvalioti, Paul Jennings, and Valentina Donzella. Influence of avc and hevc compression on detection of vehicles through faster r-cnn. *IEEE Transactions on Intelligent Transportation Systems*, 2023. 1, 2

[3] Ashok Dahal, Jakir Hossen, Chennupati Sumanth, Ganesh Sistu, Kazumi Malhan, Muhammad Amasha, and Senthil Yogamani. Deeptrailerassist: Deep learning based trailer detection, tracking and articulation angle estimation on automotive rear-view camera. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 2

[4] Mahesh M Dhananjaya, Varun Ravi Kumar, and Senthil Yogamani. Weather and light level classification for autonomous driving: Dataset, baseline and active learning. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 2816–2821. IEEE, 2021. 2

[5] Ciarán Eising, Jonathan Horgan, and Senthil Yogamani. Near-field perception for low-speed vehicle automation using surround-view fisheye cameras. *IEEE Transactions on Intelligent Transportation Systems*, 23(9):13976–13993, 2021. 2

[6] Kristian Fischer, Christian Blum, Christian Herglotz, and André Kaup. Robust deep neural object detection and segmentation for automotive driving scenario with compressed image data. In *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5. IEEE, 2021. 1

[7] Gerald Friedland, Rouxi Jia, Jingkang Wang, Bo Li, and Nathan Mundhenk. On the impact of perceptual compression on deep learning. In *2020 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 219–224. IEEE, 2020. 1

[8] Munkhjargal Gochoo, Munkh-Erdene Otgonbold, Erkhembayar Ganbold, Jun-Wei Hsieh, Ming-Ching Chang, Ping-Yang Chen, Byambaa Dorj, Hamad Al Jassmi, Ganzorig Batnasan, Fady Alnajjar, Mohammed Abduljabbar, and Fang-Pang Lin. Fisheye8k: A benchmark and dataset for fisheye camera object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 5305–5313, 2023. 2, 5

[9] Tankred Hase, Wolfgang Hintermaier, Andreas Frey, Tobias Strobel, Uwe Baumgarten, and Eckehard Steinbach. Influence of image/video compression on night vision based pedestrian detection in an automotive application. In *2011 IEEE 73rd Vehicular Technology Conference (VTC Spring)*, pages 1–5. IEEE, 2011. 1

[10] Kristian Kajak. Impact of video compression on the performance of object detection algorithms in automotive applications, 2020. 1

[11] Varun Ravi Kumar, Stefan Milz, Christian Witt, Martin Simon, Karl Amende, Johannes Petzold, Senthil Yogamani, and Timo Pech. Near-field depth estimation using monocular fisheye camera: A semi-supervised learning approach using sparse lidar data. In *CVPR Workshop*, page 2, 2018. 2

[12] Varun Ravi Kumar, Marvin Klingner, Senthil Yogamani, Markus Bach, Stefan Milz, Tim Fingscheidt, and Patrick Mäder. Svdistnet: Self-supervised near-field distance estimation on surround view fisheye cameras. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):10252–10261, 2021. 2

[13] Varun Ravi Kumar, Ciarán Eising, Christian Witt, and Senthil Kumar Yogamani. Surround-view fisheye camera perception for automated driving: Overview, survey & challenges. *IEEE Transactions on Intelligent Transportation Systems*, 24(4):3638–3659, 2023. 2

[14] Isabelle Leang, Ganesh Sistu, Fabian Bürger, Andrei Bursuc, and Senthil Yogamani. Dynamic task weighting methods for multi-task networks in autonomous driving systems. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–8. IEEE, 2020. 2

[15] Eslam Mohamed, Mahmoud Ewaisha, Mennatullah Siam, Hazem Rashed, Senthil Yogamani, Waleed Hamdy, Mohamed El-Dakdouky, and Ahmad El-Sallab. Monocular instance motion segmentation for autonomous driving: Kitti instancemotseg dataset and multi-task baseline. In *2021 IEEE Intelligent Vehicles Symposium (IV)*, pages 114–121. IEEE, 2021. 2

[16] Sambit Mohapatra, Senthil Yogamani, Heinrich Gotzig, Stefan Milz, and Patrick Mader. Bevdetnet: bird's eye view lidar point cloud based real-time 3d object detection for autonomous driving. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 2809–2815. IEEE, 2021. 1

[17] Doc. ISO/IEC JTC1/SC29/WG2 N0190. Use cases and requirements for video coding for machines. 2022. 2

[18] Guru Nayak and Gerald Friedland. Deep layers beware: Unraveling the surprising benefits of jpeg compression for image classification pre-processing. 2023. 1

[19] Maximilian Pöpperli, Raghavendra Gulagundi, Senthil Yogamani, and Stefan Milz. Capsule neural network based height classification using low-cost automotive ultrasonic sensors. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 661–666. IEEE, 2019. 2

[20] Matt Poyser, Amir Atapour-Abarghouei, and Toby P. Breckon. On the impact of lossy image and video compression on the performance of deep convolutional neural network architectures. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 2830–2837, 2021. 1

[21] Hazem Rashed, Senthil Yogamani, Ahmad El-Sallab, Pavel Krizek, and Mohamed El-Helw. Optical flow augmented semantic segmentation networks for automated driving. In *Proceedings of the International Conference on Computer Vision Theory and Applications*, 2019. 2

[22] Hazem Rashed, Eslam Mohamed, Ganesh Sistu, Varun Ravi Kumar, Ciarán Eising, Ahmad El-Sallab, and SK Yogamani. Fisheyeyolo: Object detection on fisheye cameras for autonomous driving. In *Proceedings of the Machine Learning for Autonomous Driving NeurIPS 2020 Virtual Workshop, Virtual*, 2020. 2

[23] Hazem Rashed, Eslam Mohamed, Ganesh Sistu, Varun Ravi Kumar, Ciarán Eising, Ahmad El-Sallab, and Senthil Yogamani. Generalized object detection on fisheye cameras for autonomous driving: Dataset, representations and baseline. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2271–2279, 2021. 3

[24] Sławomir Różek, Olgierd Stankiewicz, Sławomir Maćkowiak, and Marek Domański. Video coding for machines using object analysis and standard video codecs. In *2023 IEEE International Conference on Visual Communications and Image Processing (VCIP)*, pages 1–5. IEEE, 2023. 2

[25] Ganesh Sistu, Isabelle Leang, and Senthil Yogamani. Real-time joint object detection and semantic segmentation network for automated driving. *NeurIPSW on ML on the Phone and other Consumer Devices*, 2018. 2

[26] Ganesh Sistu, Isabelle Leang, Sumanth Chennupati, Senthil Yogamani, Ciarán Hughes, Stefan Milz, and Samir Rawashdeh. Neurall: Towards a unified visual perception model for automated driving. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 796–803. IEEE, 2019. 2

[27] Gary J Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand. Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on circuits and systems for video technology*, 22(12):1649–1668, 2012. 1

[28] Takehiro Tanaka, Alon Harell, and Ivan V Bajić. Does video compression impact tracking accuracy? In *2022 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1517–1521. IEEE, 2022. 1

[29] Nivedita Tripathi and Senthil Yogamani. Trained trajectory based automated parking system using Visual SLAM. In *Proceedings of the Computer Vision and Pattern Recognition Conference Workshops*, 2021. 2

[30] Michal Uricár, Jan Ulicny, Ganesh Sistu, Hazem Rashed, Pavel Krizek, David Hurych, Antonin Vobecky, and Senthil Yogamani. Desoiling dataset: Restoring soiled areas on automotive fisheye cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 2

[31] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7464–7475, 2023. 5

[32] Thomas Wiegand, Gary J Sullivan, Gisle Bjontegaard, and Ajay Luthra. Overview of the h. 264/avc video coding standard. *IEEE Transactions on circuits and systems for video technology*, 13(7):560–576, 2003. 1

[33] Lucie Yahiaoui, Jonathan Horgan, Brian Deegan, Senthil Yogamani, Ciarán Hughes, and Patrick Denny. Overview and empirical analysis of isp parameter tuning for visual perception in autonomous driving. *Journal of Imaging*, 5(10):78, 2019. 2

[34] Senthil Yogamani, Ciaran Hughes, Jonathan Horgan, Ganesh Sistu, Padraig Varley, Derek O'Dea, Michal Uricar, Stefan Milz, Martin Simon, Karl Amende, Christian Witt, Hazem Rashed, Sumanth Chennupati, Sanjaya Nayak, Saquib Mansoor, Xavier Perrotton, and Patrick Perez. Woodscape: A multi-task, multi-camera fisheye dataset for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2, 4, 5