TEMPORAL AXIAL ATTENTION FOR LIDAR-BASED 3D OBJECT DETECTION IN AUTONOMOUS DRIVING

Manuel Carranza-García[⋆] *José C. Riquelme*[⋆] *Avideh Zakhor*[†]

[⋆] Division of Computer Science, University of Sevilla, Spain
[†]Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, USA
mcarranzag@us.es, riquelme@us.es, avz@berkeley.edu

ABSTRACT

3D object detection is a core problem of the perception systems of autonomous vehicles. Despite recent progress in the field, the temporal aspect of LiDAR data has not been fully explored in current state-of-the-art detectors. This work proposes a modified CenterPoint architecture that uses temporal axial attention to exploit the sequential nature of autonomous driving data for 3D object detection. The last ten LiDAR sweeps are split into three groups of frames, and the axial attention transformer block captures both spatial and temporal dependencies among the features extracted from each group. Our proposal is evaluated using the nuScenes dataset. With this novel approach, we obtain an average mAP improvement of 3.8 and 2.3 points over the original CenterPoint in the fine/coarse pillar settings, respectively.

Index Terms— autonomous driving, attention, deep learning, LiDAR, object detection

1. INTRODUCTION

The accurate perception of the environment remains a challenging problem for autonomous vehicles. Surrounding traffic elements must be perceived robustly and efficiently under different driving scenarios and weather conditions. Therefore, 3D object detection is a crucial task in this field that has received considerable attention over the last few years.

Many autonomous driving companies such as Waymo [1] and nuTonomy [2] have released high-quality multi-modal perception datasets. LiDAR has proven to be a reliable and accurate distance sensor for 3D object detection, which can also work under low illumination conditions such as nighttime. The availability of these large-scale sequential LiDAR datasets has opened many research opportunities for developing temporal algorithms for 3D detection.

In the recent literature, many LiDAR-based detection architectures have been proposed. Popular models such as PointPillars [3] or VoxelNet [4] voxelize the space around the vehicle and learn feature representations of the point cloud, which are then efficiently processed by 2D convolutions to obtain detections. Other methods operate directly on points [5] or range image views [6]. Furthermore, there have also been efforts to develop sensor fusion methods [7]. Currently, CenterPoint (CP) [8] is the state-of-the-art for 3D object detection in the nuScenes and Waymo datasets.

Despite the recent progress in the field, little work has been done in the area of extracting temporal information from point clouds [9, 10, 11]. The above-mentioned architectures only merge multiple sweeps into a single point cloud and append a timestamp feature. This approach does not fully exploit the temporal nature of LiDAR data, which could significantly help to improve the detection performance.

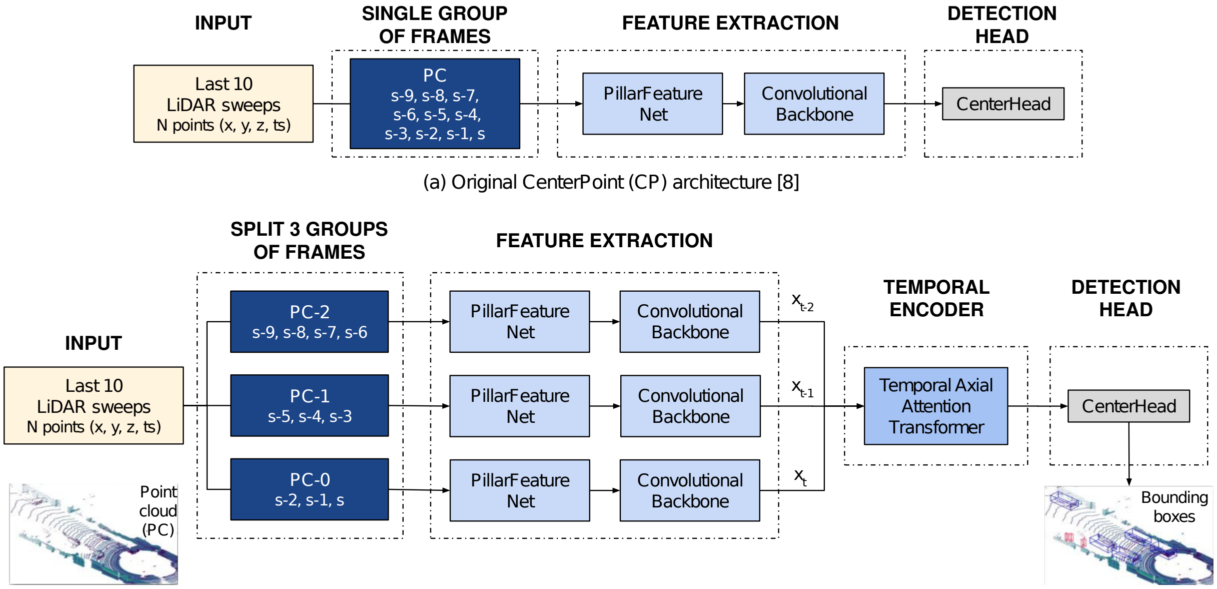
Therefore, in this work, we propose a modified CP architecture that aims to leverage the temporal information in sequential LiDAR frames. Firstly, the input pipeline is modified by splitting the last sweeps into multiple groups of frames. The temporal dimension is created by stacking features extracted from each group. Then, a temporal encoder, placed between the convolutional backbone and the detection head, explicitly models spatio-temporal dependencies in the point cloud representations. This methodology follows our prior work on temporal data for 3D detection [12]. In this study, we update the framework from PointPillars to CenterPoint, and replace the former recurrent ConvLSTM temporal encoder with a Transformer network that uses axial self-attention.

Our proposal is validated using the nuScenes dataset. The experimental study compares the performance with the original CP architecture under three different pillar size settings. The rest of the paper is organized as follows: Section 2 presents the background of this study; Section 3 describes the proposed method; Section 4 presents the experimental results; Section 5 presents the conclusions.

2. BACKGROUND

CenterPoint The CenterPoint framework, presented in Figure 1a, is an effective center-based 3D object detector [8]. CP relies on existing standard backbones, such as

The authors gratefully acknowledge funding from Berkeley DeepDrive, Fulbright, IMFAHE, and the projects PID2020-117954RB-C22, P18-RT-2778, PYC20 RE 078 US.



(a) Original CenterPoint (CP) architecture [8]

(b) CenterPoint with Temporal Axial Attention (CP-TAA) architecture

Fig. 1: (a) Original CenterPoint framework. (b) Our proposed CP-TAA architecture leverages the temporal nature of LiDAR data by modifying the input pipeline and including a Transformer-based temporal encoder between the convolutional backbone and the detection head.

VoxelNet [4] or PointPillars [3], to extract 2D map-view feature representations from 3D point clouds. In this feature extraction process, a grid of voxels/pillars around the vehicle is generated, and PointNet [13] is used to extract an *n*-dimensional feature vector describing each of them. These scene representations are fed to a 2D convolutional backbone with a feature pyramid network architecture. The extracted 2D feature map is used to generate 3D bounding boxes.

CP replaces the traditional anchor-based detection head with a center-based approach. CenterHead uses a keypoint detector to find the center of objects and then regresses other attributes such as size and orientation. This method aims to alleviate issues with axis-aligned anchors for locating rotated objects accurately. Since points have no intrinsic orientation, the backbone can learn the rotational invariance of objects and obtain more accurate bounding box predictions.

Temporal LiDAR data for detection Although CP uses multiple sweeps to enrich the point cloud, it does not fully take advantage of the temporal information. As in the anchor-based PointPillars framework [2], CP transforms the last ten sweeps to the current reference frame using the vehicle's ego-motion, and appends a timestamp feature to each point. In nuScenes, the LiDAR sensor runs at 20Hz, hence these sweeps correspond to 0.5 seconds of driving data. Therefore, this accumulation into a single large point cloud may not be optimal since moving objects can be smeared.

To address these issues, our previous work [12] extended the PointPillar detection framework to explicitly capture temporal information among successive groups of LiDAR frames. This initial proposal uses a recurrent ConvLSTM network to process the non-overlapping groups of featurized point clouds and improve the detection precision.

Axial attention Self-attention transformers have obtained success in sequence processing and computer vision problems [14]. Furthermore, recent works in 2D object detection, such as DETR [15], have proposed combining convolutional backbones with attention-based networks. Motivated by these results, we aim to explore the suitability of Transformers for 3D object detection using temporal LiDAR data. However, given the large dimensionality of the voxelized LiDAR inputs, the use of standard self-attention is computationally prohibitive. The axial attention approach provides a solution to deal with high dimensional tensors [16]. Axial attention allows to efficiently integrate global information by processing each individual axis independently in a sequential manner. It has been successfully applied in other domains such as protein language models [17] or precipitation forecasting [18].

3. METHODOLOGY

Inspired by our prior work on 3D object detection using temporal LiDAR data [12], our proposal is to extend the original CenterPoint with a novel temporal encoder that uses axial attention to better capture spatio-temporal dependencies in the data. Figure 1b presents the CenterPoint with temporal axial attention (CP-TAA) and the modified input pipeline.

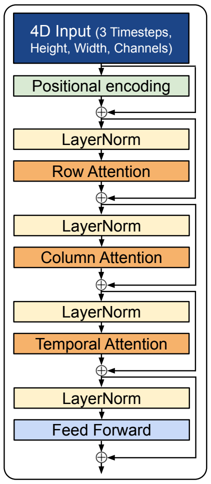


Fig. 2: Transformer block with axial attention used as temporal encoder in our proposed CP-TAA architecture.

Given an input LiDAR point cloud (PC) with N points defined by (x, y, z) coordinates and a timestamp feature, the aim of the proposed object detection model is to output 3D bounding boxes of the detected objects. As in [12], the last ten LiDAR sweeps are divided into three non-overlapping groups of frames (GoFs) called PC-0, PC-1, PC-2, to leverage the temporal information from successive point clouds. In this work, we use the CP-Pillar version rather than CP-Voxel, since it runs faster and eliminates the need for tuning the size of the vertical dimension of voxels. Pillars are created for each resulting point cloud, and features are extracted using the shared convolutional backbone. The temporal dimension is then created by concatenating the feature representations from the three GoFs. The temporal encoder processes the sequence and outputs a single 2D feature map, which is used by the CenterHead detector to generate bounding boxes. The division into three GoFs provides a good efficiency/accuracy trade-off since more groups would require an excessive amount of memory.

The advantage of this input data decomposition is that it reduces the smearing of objects in motion compared to a dense cloud with all LiDAR frames. Furthermore, since the GoFs are independent, the feature extraction process could happen in parallel or follow a rolling window approach. With that scheme, only the most recent point cloud (PC-0) would need to pass through the backbone network, which could save computation and significantly speed up the pipeline.

Besides using CenterPoint rather than PointPillars, the novelty with respect to the work in [12] is that we design a Transformer-based temporal encoder that outperforms the previous ConvLSTM proposal. Transformer networks have recently shown promising results in many computer vision problems [14, 15] and are well suited for the temporal nature of sequential LiDAR data. The proposed self-attention Transformer block is able to integrate global information and maintain long-distance temporal and spatial dependence. In contrast, the receptive field in the ConvLSTM encoder is local and limited to the size of the kernel, typically 1×1 or 3×3. To expand the receptive field, it is necessary to stack several layers, which increases the amount of computation.

Nevertheless, standard global self-attention is prohibitively expensive to compute for large inputs, which is the case for our three sequential feature maps with large spatial dimensions, e.g. 128×128. Therefore, we employ the axial attention approach developed in [16, 19]. The core idea is to separate the desired 3D attention into three sequential steps that apply 1D attention in the temporal, height, and width axes. This formulation enables the model to efficiently learn global interactions with a reasonable computational cost.

Figure 2 illustrates the proposed Transformer block. First, learned positional encodings are used along the temporal and spatial dimensions to preserve the order of inputs. Then, row, column, and temporal attentions are sequentially applied. Residual connections and layer normalization are employed before attention, and a feed-forward layer is used at the end. The temporal encoder outputs three attention maps corresponding to each timestep. The last map, which corresponds to the most recent GoFs, is used by the CenterHead detector.

4. EXPERIMENTAL RESULTS

This section presents the experimental results obtained on the 3D object detection task with the nuScenes dataset. This dataset contains 1000 driving scenes of 20 seconds length in Boston and Singapore. In this study, we use the train and validation split provided when downloading the dataset. The LiDAR data is recorded at 20Hz, but only keyframes sampled at 2 Hz have annotations. There are 700 scenes in the training set with about 28,000 keyframes and 150 scenes in the validation set with about 6,000 keyframes. For each keyframe, the ten preceding LiDAR sweeps are also used as input to the model.

The nuScenes detection task comprises ten object classes. The detection performance is evaluated according to the per-class and mean average precision (mAP) and the nuScenes detection score (NDS) [2]. AP calculates the normalized area under the precision-recall curve. In this task, instead of using intersection over union (IoU), the matching is defined by thresholding the 2D center distance d on the ground plane. The mAP is calculated as the average AP over matching thresholds $\mathbb{D} = \{0.5, 1, 2, 4\}$ meters and the set of classes $\mathbb{C}$:

$$mAP = \frac{1}{|\mathbb{C}||\mathbb{D}|} \sum_{c \in \mathbb{C}} \sum_{d \in \mathbb{D}} AP_{c,d} \tag{1}$$

While AP is the most popular object detection metric, it cannot capture all aspects of the detection task. Therefore, together with mAP, the NDS includes other five metrics based on location, size, orientation, attributes, and velocity. A more detailed explanation of the NDS metric can be found in [2].

Table 1: Precision and efficiency of the studied architectures on the nuScenes validation set with different pillar size. The reported metrics are the per-class AP, mAP, NDS, speed in frames per seconds, memory usage (GB), and number of parameters (millions) of the models. Best results are highlighted in bold.

	Pillar	Car	Ped.	Truck	Bus	Trail.	C. Veh.	Moto.	Bicy.	TC	Barr.	mAP	NDS	FPS	Mem.	# Param.
Original CP	0.15	80.9	77.4	51.3	59.7	25.8	13.5	49.9	23.6	51.2	51	48.4	58.5	**23.4**	**1.9**	**5.9**
CP-ConvLSTM		81.0	78.6	49.1	60.0	26.5	12.5	51.9	27	54.1	53.3	49.4	59.6	12.2	5.4	7.2
CP-TAA		**81.9**	**79.1**	**52.2**	**63.6**	**34.1**	**14.8**	**57.0**	**27.2**	**54.7**	**57.1**	**52.2**	**60.5**	10.3	8.4	7.5
Original CP	0.2	79.9	73.1	48.4	60.2	**33.6**	11.6	49.6	22.4	45.1	46.3	47.0	57.1	**30.1**	**1.2**	**5.9**
CP-ConvLSTM		80.1	**75.7**	**50.7**	59.8	31.7	**13.1**	51.4	**24.7**	51.4	48.7	48.7	58.3	17.8	3.1	7.2
CP-TAA		**80.3**	75.6	47.7	**64.7**	33.0	13.0	**53.0**	**24.7**	**51.5**	**52.4**	**49.6**	**58.6**	16.2	4.7	7.5
Original CP	0.3	78.0	64.5	41.1	57.0	29.5	11.1	42.5	13.4	41.6	41.8	42.0	53.0	**42.2**	**0.8**	**5.9**
CP-ConvLSTM		**78.5**	67.6	**44.5**	56.5	28.3	**13.8**	44.8	17.1	**42.9**	40.5	43.5	54.4	28.9	1.8	7.2
CP-TAA		78.4	**67.7**	43.3	**60.3**	**32.8**	10.5	**46.1**	**17.2**	41.6	**45.0**	**44.3**	**54.7**	24.8	2.3	7.5

The implemented code, using the MMDetection3D toolbox [20], is publicly available[1]. Besides the modifications explained in Section 3, the rest of the design of the architecture and hyper-parameters are kept as in the original CP implementation [8]. The hidden dimension of the temporal encoder is kept to 384 channels, which is the dimension of the output of the convolutional backbone. The models are trained for 30 epochs, with batch size 8 on two Nvidia Titan RTX 24GB GPUs. In all experiments, the AdamW optimizer is used with a one-cycle learning rate policy, max value 1e-4, weight decay 0.01, and momentum 0.85 to 0.95.

Given this experimental setup, the original CP detector is compared to our modified architecture with a temporal encoder, either using the prior ConvLSTM module (CP-ConvLSTM) or the novel Transformer block (CP-TAA). The performance is analyzed for different degrees of voxelization of the space around the vehicle. This parameter has a significant impact on the accuracy, computation time, and memory requirements of the models. While the original CP-Pillars used pillar size $(0.2m, 0.2m)$ [8], we explore two additional grid settings, a finer $(0.15m, 0.15m)$ and a coarser $(0.3m, 0.3m)$ configuration.

Table 1 presents the precision and efficiency results obtained over the nuScenes validation set with different pillar size settings. As can be seen, our proposed modification with a temporal encoder significantly enhances the detection precision under all three pillar size settings. From finer to coarser pillars, respectively, the mAP improvement of the CP-TAA with respect to the original CP is 3.8, 2.6, and 2.4 points. In terms of NDS, our proposal outperforms the baseline by 2.0, 1.5, and 1.7 points, respectively. Furthermore, the axial attention approach presents an advantage over the recurrent ConvLSTM encoder. Specifically, the mAP is enhanced by 2.8, 0.9, and 0.8 points when using the proposed transformer block. These improvements can also be seen in nearly all classes individually, regardless of the instance distribution. For instance, with pillar size 0.15, the best mAP increases are obtained in classes such as trailers (+8.3), motorcycles (+7.1), or barriers (+6.1).

These results show that developing methods with the temporal aspect in mind is essential for 3D object detection using LiDAR. Merging all LiDAR sweeps into a single point cloud is not optimal and does not fully leverage the temporal information. The experiments also confirm the capacity of Transformer networks to process temporal data for 3D object detection. The axial attention approach allows to efficiently capture global spatio-temporal dependencies in sequential point cloud representations, outperforming the previous recurrent network approach.

Table 1 also reports the speed, memory, and the number of parameters of the studied architectures for batch size of one. Both CP-ConvLSTM and CP-TAA have comparable speed and run near real-time, although axial attention is more memory intensive. As expected, the temporal encoder introduces a computational overhead and reduces the speed compared to the original CP. Note that FPS are reported considering that GoFs features are extracted sequentially, and a speedup could be obtained with a parallel/streaming approach.

5. CONCLUSIONS

In this paper, we modify the state-of-the-art CenterPoint 3D object detection framework to fully exploit the temporal information present in autonomous driving LiDAR data. Our method modifies the input pipeline, splitting recent sweeps into groups of frames to create the temporal dimension. Then, a Transformer network based on axial self-attention extracts spatio-temporal features before the detection head.

We experimented with three different pillar sizes and obtained a significant precision improvement over the original CenterPoint baseline with all configurations. These findings demonstrate the importance of developing detection models with a temporal approach, and the suitability of Transformers as temporal encoders for this problem.

In future work, we aim to extend the framework with multi-sensor fusion approaches using temporal data. Furthermore, we plan to focus on improving the computational efficiency of the proposed architecture, exploring the performance in a streaming scenario.

[1] https://github.com/carranza96/mmdetection3d/tree/cp-taa

6. REFERENCES

[1] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, et al., "Scalability in Perception for Autonomous Driving: Waymo Open Dataset," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 2443–2451.

[2] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom, "nuScenes: A Multimodal Dataset for Autonomous Driving," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11618–11628.

[3] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom, "PointPillars: Fast Encoders for Object Detection From Point Clouds," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 12689–12697.

[4] Yin Zhou and Oncel Tuzel, "VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4490–4499.

[5] Charles R. Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J. Guibas, "Frustum PointNets for 3D Object Detection from RGB-D Data," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 918–927.

[6] Lue Fan, Xuan Xiong, Feng Wang, Naiyan Wang, and Zhaoxiang Zhang, "RangeDet: In Defense of Range View for LiDAR-based 3D Object Detection," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 2898–2907.

[7] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl, "Multimodal virtual point 3d detection," *NeurIPS*, 2021.

[8] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl, "Center-based 3D Object Detection and Tracking," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 11779–11788.

[9] Ahmad El Sallab, Ibrahim Sobh, Mahmoud Zidan, Mohamed Zahran, and Sherif Abdelkarim, "Yolo4d: A spatio-temporal approach for real-time multi-object detection and classification from lidar point clouds," *Neural Information Processing Systems (NeurIPS) Workshop MLITS*, 2018.

[10] David Deng and Avideh Zakhor, "Temporal lidar frame prediction for autonomous driving," in *2020 International Conference on 3D Vision (3DV)*, 2020, pp. 829–837.

[11] K.S. Chidanand Kumar and S. Al-Stouhi, "Real-time spatial-temporal context approach for 3D object detection using LiDAR," in *VEHITS 2020 - 6th International Conference on Vehicle Technology and Intelligent Transport Systems*, pp. 432–439.

[12] Scott McCrae and Avideh Zakhor, "3D Object Detection For Autonomous Driving Using Temporal Lidar Data," in *2020 IEEE International Conference on Image Processing (ICIP)*, pp. 2661–2665.

[13] R. Qi Charles, Hao Su, Mo Kaichun, and Leonidas J. Guibas, "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 77–85.

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *9th International Conference on Learning Representations, ICLR 2021.*

[15] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko, "End-to-end object detection with transformers," in *Computer Vision – ECCV 2020*, pp. 213–229.

[16] Jonathan Ho, Nal Kalchbrenner, Dirk Weissenborn, and Tim Salimans, "Axial attention in multidimensional transformers," *ArXiv*, vol. abs/1912.12180, 2019.

[17] Roshan M Rao, Jason Liu, Robert Verkuil, Joshua Meier, John Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives, "MSA Transformer," in *38th International Conference on Machine Learning (ICML)*, 2021, vol. 139, pp. 8844–8856.

[18] Casper Kaae Sønderby, Lasse Espeholt, Jonathan Heek, Mostafa Dehghani, Avital Oliver, Tim Salimans, Shreya Agrawal, Jason Hickey, and Nal Kalchbrenner, "MetNet: A Neural Weather Model for Precipitation Forecasting," *ArXiv*, vol. abs/2003.12140, 2020.

[19] H. Wang, Y. Zhu, B. Green, H. Adam, A. Yuille, and L.-C. Chen, "Axial-DeepLab: Stand-Alone Axial-Attention for Panoptic Segmentation," in *16th European Conference on Computer Vision, ECCV 2020*, 2020, pp. 108–126.

[20] MMDetection3D Contributors, "MMDetection3D: OpenMMLab next-generation platform for general 3D object detection," https://github.com/open-mmlab/mmdetection3d, 2020.