MMPrune4U: Regularizing Multimodal Feature Distortion in Weight Pruning for Deep Neural Network Compression

Sudip Das[1]
sudip.das@valeo.com
Kaixin Xu[2]
kaixin002@e.ntu.edu.sg
Nushrat Hussain[3]
hnushrat_t@isical.ac.in
Ziyuan Zhao[2]
s210088@e.ntu.edu.sg
Arindam Das[1,4]
arindam.das@valeo.com
Weisi Lin[2]
wslin@ntu.edu.sg
Ujjwal Bhattacharya[3]
ujjwal@isical.ac.in

[1] DSW, Valeo India
[2] Nanyang Technological University, Singapore
[3] Indian Statistical Institute, Kolkata
[4] University of Limerick, Ireland

Abstract

Despite the remarkable success of multimodal models in automotive applications, their practical benefits are often accompanied by a large number of parameters, including redundant and excessive weights. This poses hurdles to their deployment on embedded devices due to the substantial computational costs compared to unimodal models. Model sparsification is among the common solutions to reduce the resources required for computation and increase throughput of the system. Although many recent studies in model sparsification and pruning achieve remarkable performance for unimodal models, they overlook capturing the layer-wise sensitivity towards accuracy and behaviors for distinct modalities in response to the pruning, leading to information loss in the downstream tasks of the pruned model. We introduce MMPrune4U, a layer-adaptive weight pruning method explicitly designed to support multimodal 3D scene understanding that incorporates a regularizer based on log-Sobolev inequality. This approach uncovers a crucial property related to the distortion of features resulting from pruning weights across multiple layers while keeping a predefined pruning ratio. As per the changes in the output distribution of the each layer during pruning compared to unpruned model, we regularize the distortion through the functional Fisher information. We formulate our layer-adaptive pruning by considering the layerwise impact to the downstream tasks and optimise the objective function through combinatorial optimization challenge, which we effectively address using dynamic programming techniques. The proposed MMPrune4U method demonstrates superior performance in comparison to the existing state-of-the-art methods, as shown by experimental results on both nuScenes and SemanticKITTI datasets.

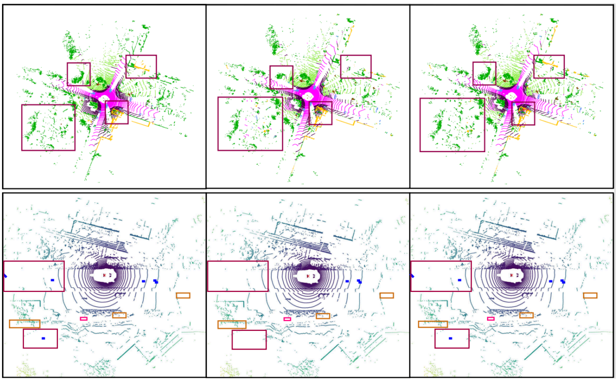


Figure 1: Proposed pruning method, MMPrune4U, when applied on RPVNet [53] model for semantic segmentation using semanticKITTI and object detection using nuScenes datasets, produces comparable predictions with respect to the latest pruning technique RD Prune [54].

1 Introduction

An accurate understanding of visual information from the environment is essential for various applications, including autonomous driving and robotics [8, 25, 29, 34, 36]. The level of comprehension directly impacts the effectiveness of subsequent tasks like path planning and control [10, 21, 52]. To meet the safety standard of autonomous driving systems, it is typical to employ a combination of sensors [34, 36], including cameras and LiDAR, to enhance both reliability and accuracy since LiDAR point cloud provides precise 3D geometric measurements but lacks color and texture information [5, 61, 62]. On the other hand, camera images complement these point cloud views by offering comprehensive semantic information [5, 22, 38], thus maximizing the utilization of available data. PointPainting [47] merges semantic information extracted from 2D images with raw LiDAR points. Further, Wang et al. [49] and Yin et al. [60] have subsequently proposed enhancements to the PointPainting framework. However, most scene understanding studies considered multimodal models [9, 11, 32, 37, 45] with a large number of parameters that result in high energy consumption, delayed system output, and pose challenges for deployment on embedded devices with limited resources. Neural network pruning is a commonly employed approach to reduce the computation complexity by identifying redundant subset of parameters and thereby aiding in the reduction of FLOPs (FLoating-point OPerations) [15, 17, 42, 43] and satisfying the storage requirements [15, 19, 28, 35, 40]. It has been studied as a fundamental technique for a long time, and in most cases, single-modality has been considered as the default scenario. Post-train pruning is among such unimodal pruning schemes for models like CNNs, which prunes weights/parameters from pretrained dense models. Han et al. [18, 19] proposed a few pioneering works in post-train pruning, adopting magnitude-based iterative pruning for simple CNNs such as LeNet and AlexNet. Molchanov et al. [42] adopted Taylor-based criterion as a significance score for intra-layer parameter pruning. Methods such as [44, 63] also leveraged magnitude-based scores, but applied a global threshold for all layers to prune out low-scored parameters. The pruning techniques in [12, 14] determined the layerwise sparsity rate by architectural heuristics. Leet et al. [27] proposed a method to rank magnitude-based scores with inter-layer constraints. Isik et al. [23] derived output distortion-aware layer-wise sparsity ratio from laplacian distribution assumptions of layer weights. Recent advancements in pruning strategy towards task-agnostic pruning avoids the need for network re-pruning for each newly considered task. Further, it can be categorized into two parts: unimodal [6, 7, 41, 55] and multimodal [13] network pruning. These approaches provide a generic sparse model that can be utilized for various unknown downstream tasks, while [57] worked on a structural pruning method aiming at reducing the latency of various components of the Vision Transformer (ViT). Kichler et al. [24] and Wang et al. [50] explored a two-step method to retain knowledge in neural networks. Firstly, it prunes the network, followed by fine-tuning to transfer knowledge from the unpruned model. One limitation of this approach is the disregard for considering the mutual impact of different layers, which makes the pruning process less effective and leads to subpar model accuracy. In some studies involving multimodal networks [1, 39], the pruning method was applied. However, applying high proportions of pruning ratios resulted in deteriorated accuracy.

We observe a significant discrepancy in the information contained within the features of each modality, as quantified by Fisher information, between the features of each layer of the pruned model and those of the unpruned models, resulting in considerable information loss. Particularly in the setup of multimodal models, the impact of information loss is more effective in LiDAR point clouds compared to camera images, as LiDAR provides accurate yet sparse 3D point clouds. During pruning, the information contained in LiDAR features degrades more significantly, limiting its contribution to downstream tasks. Conversely, weights from one modality may contain similar knowledge to that found in another modality. So far, no similar study has been explored to prune the network through the understanding of the information contained within the features of each layer relative to the unpruned model.

In this work, we propose a novel regularizer for a jointly optimized layer-adaptive approach aimed at minimizing the trade-off between FLOPs and accuracy. Specifically, our single-stage pruning approach, along with a regularization term, effectively preserves information loss during the pruning of certain modality branches of the multimodal network, thereby improving the eventual multimodal model performance as shown in Figure 1. The notable contributions of the present study are summarized as follows,

- We formulate a generic post-train pruning scheme for multimodal 3D scene understanding models.
- We present an approach aimed at preventing information loss in the features across all the layers of the pruned model. This involves leveraging the Logarithmic Sobolev Inequality to ensure an equivalent consideration of feature information between each layer of pruned and unpruned model.
- Our extensive evaluation of nuScenes [4] and SemanticKITTI [3] datasets while using MMPrune4U method achieves state-of-the-art performances with significantly less number of FLOPs in comparison to the unpruned models.

2 Proposed Approach

2.1 Preliminaries

We targeted pruning learnable parameters for all feature extraction layers in Multimodal 3D networks. To decide which neurons in a weight tensor need to be pruned, given a layer sparsity ratio α, we rank them by the absolute value and eliminate the bottom-ranked ones. Mathematically, we first obtain the neuron score by the Taylor expansion $\boldsymbol{S} = |\boldsymbol{W}|$ similar to [43]. The above pruning scheme can be formulated as $\widetilde{\mathbf{W}} = \boldsymbol{W} \odot \boldsymbol{M}_\alpha(\boldsymbol{S})$, where $\boldsymbol{M}_\alpha(\boldsymbol{S})$ is the binary mask generated from the ranking score matrix $\boldsymbol{S}$.

We essentially adopt a layerwise sparsity scheme in [54], which provides a rate-distortion-based layerwise pruning ratio allocation algorithm to minimize the output distortion. Given a neural network f, we denote $\boldsymbol{W}^{(1:l)} = \left(\boldsymbol{W}^{(1)}, ..., \boldsymbol{W}^{(l)}\right)$ as all the parameters of f, where l is the total number of layers in f and $\boldsymbol{W}^{(i)}$ is the weights in layer i. When we prune the parameters in layer i to j, we will obtain a new parameter set for those layers $\widetilde{\mathbf{W}}^{(i:j)}$. The objective of model pruning on f can be formulated as to minimize the output distortion caused by pruning $f(x,v;\boldsymbol{W}^{(1:l)}) - f(x,v;\widetilde{\mathbf{W}}^{(1:l)})$:

$$\min \|f(x,v;\boldsymbol{W}^{(1:l)}) - f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\|^2 \;\; s.t. \frac{\|\widetilde{\mathbf{W}}^{(1:l)}\|_0}{\|\boldsymbol{W}^{(1:l)}\|_0} \leq R \tag{1}$$

where R denotes the pruning ratio for the entire network. We exploit the additivity approximation adopted in [54] to leverage the intractable original problem, which approximates the output distortion caused by pruning **all** layers' weights into the sum of the output distortion due to **individually** pruning of each layer:

$$E\left(\|f(x,v;W^{(1:l)}) - f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\|^2\right) = \sum_{i=1}^{l} E(\delta_i^d) \tag{2}$$

where δ_i^d denotes the output distortion when only pruning the weights in layer i:

$$\delta_i^d = f(x,v;\boldsymbol{W}^{(1:i-1)}, \widetilde{\mathbf{W}}^{(i)}, \boldsymbol{W}^{(i+1,l)}) - f(x,v;\boldsymbol{W}^{(1;l)}) \tag{3}$$

2.2 Features Discrepancy-aware Pruning

We devise a pruning framework integrating Logarithmic Sobolev Inequalities [16] to address information loss issues at different layers of the network during parameter pruning. Our aim is to utilize the features at different layers from the multimodal trained model $f(x;\boldsymbol{W}^{(1:l)})$ to assess the information loss of features for $f(x;\widetilde{\mathbf{W}}^{(1:l)})$ as we selectively prune a subset of parameters through regularizer. The training dataset X comprises LiDAR point cloud data x_l^k and multiview camera images x_c^k, with each instance x^i consisting of both types of data, along with their respective ground truth labels y^k. We feed the data X (LiDAR point cloud and Multiview images) into both multimodal models, $f(x;\boldsymbol{W}^{(1:l)})$ and $f(x;\widetilde{\mathbf{W}}^{(1:l)})$, extracting multimodal features represented as Q_t^l and Q_p^l within the probability measure space of each respective model for the sample distribution x. The uncertainty of the random variables, measured by $H(\cdot)$ using Cross Entropy, is quantified by comparing the distributions of $r(\hat{y})$ and $f(\hat{y}|x;\widetilde{\mathbf{W}}^{(1:l)})$, is:

$$H(r,f) = -\sum_{\hat{y}} r(\hat{y}) \log f(\hat{y}|x;\widetilde{\mathbf{W}}^{(1:l)}) \tag{4}$$

where the function $r(\hat{y})$ corresponds to the ground truth. As the optimization progresses, the network runs the risk of losing information across different layers as a consequence of parameter pruning. Therefore, approximating a function $f(x;\widetilde{\mathbf{W}}^{(1:l)})$ with distortion in the parameters to form a pruned model may result in poor performance under multimodality scenario.

To address the disparity between $f(x;\widetilde{\mathbf{W}}^{(1:l)})$ and $f(x;\boldsymbol{W}^{(1:l)})$, we employ Fisher information to quantify the degree of distortion in the weights $\widetilde{\mathbf{W}}$, resulting in information loss within each input distribution (i.e., x_l^k and x_c^k), and the formula for the calculation as outlined in (7). We utilize the Logarithmic Sobolev method to extract equally significant features from the model $f(x;\widetilde{\mathbf{W}}^{(1:l)})$ in comparison to $f(x;\boldsymbol{W}^{(1:l)})$.

$$\begin{aligned}&\int_{\mathbb{R}^n} \|f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\|^2 \log \|f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\|\, du^x(v)\\&\leq \int_{\mathbb{R}^n} \|\nabla f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\|^2\, du^x(v) + \|f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\|_2^2 \log \|f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\|_2\end{aligned} \tag{5}$$

here, $du^x(v)$ represent probability density function where u denotes the Gaussian measure on R^2 and v is used to represent the stochastic variable. The norm $\|f(\cdot)\|$ is defined in the Hilbert space L^2. Precisely,

$$du^x(v) = \frac{1}{\sqrt{(2\pi)^n \det(\Sigma)}} \exp\left(-\frac{1}{2}(\|x\|^2/2)\right) dx \tag{6}$$

We derive the following equation under the condition that the function $f(x,v;\widetilde{\mathbf{W}}^{(1:l)}) \geq 0$,

$$\begin{aligned}&\int_{\mathbb{R}^n} f(x,v;\widetilde{\mathbf{W}}^{(1:l)}) \log f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\, du^x(v)\\&- \int_{\mathbb{R}^n} f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\, du^x(v) \log\left(\int_{\mathbb{R}^n} f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\, du^x(v)\right)\\&\leq \frac{1}{2}\int_{\mathbb{R}^n} f(x,v;\widetilde{\mathbf{W}}^{(1:l)}) \frac{\|\nabla f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\|^2}{f(x,v;\widetilde{\mathbf{W}}^{(1:l)})}\, du^x(v)\end{aligned} \tag{7}$$

The above equation says that the function of entropy remains non-negative owing to the non-negativity inherent in the Fisher information formulation. Additionally, it bounds the functional entropy $E(f(x,v;\widetilde{\mathbf{W}}^{(1:l)}))$ utilizing the Fisher information method through the logarithmic Sobolev inequality. It is expressed as follows,

$$\begin{aligned}E(f(x,v;\widetilde{\mathbf{W}}^{(1:l)})) &\cong \int_{\mathbb{R}^n} f(x,v;\widetilde{\mathbf{W}}^{(1:l)}) \log f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\, du^x(v)\\&- \int_{\mathbb{R}^n} f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\, du^x(v) \log\left(\int_{\mathbb{R}^n} f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\, du^x(v)\right)\end{aligned} \tag{8}$$

where $\int_{\mathbb{R}^n} f(x,v;\widetilde{\mathbf{W}}^{(1:l)}) \log f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\, du^x(v)$ represent the entropy. This expression quantifies the information content associated with the distribution $f(x,v;\widetilde{\mathbf{W}}^{(1:l)})$ and the distortion with respect to the probability measure $du^x(v)$ on R^2. Essentially, it calculates the extent to which the distribution of $f(x,v;\widetilde{\mathbf{W}}^{(1:l)})$ encapsulates information within itself. $\int_{\mathbb{R}^n} f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\, du^x(v) \log\left(\int_{\mathbb{R}^n} f(x,v;\widetilde{\mathbf{W}}^{(1:l)})\, du^x(v)\right)$ refers to expectation of $f(x,v;\widetilde{\mathbf{W}}^{(1:l)})$ under the Gaussian measure $du^x(v)$ over entire space and it captures the uncertainty which reflects deviation of the function $f(x,v;\widetilde{\mathbf{W}}^{(1:l)})$ around its average value and accounts for the spread of $f(x,v;\widetilde{\mathbf{W}}^{(1:l)})$ with respect to the Gaussian measure.

At various layers, in the pursuit of maximizing the information within the latent space for $f(x;\widetilde{\mathbf{W}}^{(1:l)})$, we define $,u_p^X$ and $,u_t^X$ as measures corresponding to the distributions $f(x;\widetilde{\mathbf{W}}^{(1:l)})$ and $f(x;\boldsymbol{W}^{(1:l)})$, respectively. We define two variables - m and v for representing the mean (μ), and variance (σ^2). Throughout optimization, u_t^X and $,u_p^X$ follow Gaussian distributions, characterized as $,u_t^X \sim \mathcal{N}(m^{X^t}, v^{X^t})$ and $,u_p^X \sim \mathcal{N}(m^{X^p}, v^{X^p})$. Here, we denote (m^{X^t}, m^{X^p}) and (v_X^t, v_X^p) as mean and variance of the measures of the pruned and unpruned model. The product measure across distributions in (7) is expressed as $u^X = u_t^X \otimes u_p^X$.

Algorithm 1 MMPrune4U Algorithm

Input: Training dataset $\mathcal{D}_t(X)$, Calibration dataset $\mathcal{D}_c(X)$, model f with l layers, Number of possible pruning ratios for each layer K, Fine-tuning epochs E.

Output: The pruned model $f(\cdot;\widetilde{\boldsymbol{W}}^{(1:l)})$.

Inference $\mathcal{F}$ on $\mathcal{D}_c(X)$ to get output: $\mathbb{Y} \leftarrow \{f(x,v;\boldsymbol{W}^{(1:l)}) \mid \forall X \in \mathcal{D}_c(X)\}$.
for i from 1 to l **do**
  for k from 1 to K **do**
    Calculate $\delta_{i,k}$ following Eqn. 11.
  end for
end for
Obtain layerwise pruning ratios α_i^* using $\delta_{i,k}$ from Eqn. 12.
for i from 1 to l **do**
  Prune $\boldsymbol{W}^{(i)}$ given α_i^* : $\widetilde{\boldsymbol{W}}^{(i)} \leftarrow \boldsymbol{W}^{(i)} \odot \boldsymbol{M}_{\alpha_i^*}(\boldsymbol{S})$.
end for
for e from 1 to E **do**
  Finetune $f(\cdot;\widetilde{\boldsymbol{W}}^{(1:l)})$ on $\mathcal{D}_t(X)$.
end for

We consider a function $S^X(\cdot)$ in Eqn. (9) to calculate the sensitivity of the function, $f(x;\widetilde{\mathbf{W}}^{(1:l)})$, for the neural architecture to the Gaussian measures u_t^X and u_p^X. Specifically, It helps to quantify the changes in the Gaussian measures that affect the $f(x;\widetilde{\mathbf{W}}^{(1:l)})$.

$$S^X(x, u_t^X; u_p^X; \widetilde{\mathbf{W}}^{(1:l)}) = H(f^p(u_t^X; u_p^X; \widetilde{\mathbf{W}}^{(1:l)}), f^p(x; \widetilde{\mathbf{W}}^{(1:l)})) \tag{9}$$

Hence, we substitute the sensitivity function in Eqn. (7) with the following regularization,

$$\lambda^{regu} = \max_{S^X(x, u_t^X; u_p^X; \widetilde{\mathbf{W}}^{(1:l)})} \left[\frac{1}{2} \int_{\mathbb{R}^n} \frac{\|\nabla S^X(x, u_t^X; u_p^X; \widetilde{\mathbf{W}}^{(1:l)})\|^2}{S^X(x, u_t^X; u_p^X; \widetilde{\mathbf{W}}^{(1:l)})} du^x(v) \right] \tag{10}$$

The gradient energy plays a pivotal role by penalizing with large gradients in $f(x;\widetilde{\mathbf{W}}^{(1:l)})$ and encourages to extract equivalent information with respect to a reference model as it is denoted to $f(x;\boldsymbol{W}^{(1:l)})$.

2.3 Final Objectives

From Eqn. 10, We find that the original optimization problem can still be reformulated into a combinatorial problem related to layerwise operands, by amending the Eqn. 3 with the λ_{regu} of i-th layer denoted as λ_{regu}^i. The final layerwise objective is as follows,

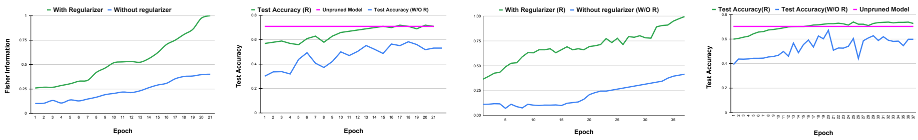
$$\delta_i = \delta_i^d + \lambda_i^{regu} \tag{11}$$

Therefore, we apply the dynamic programming solver as introduced in [54] to finally solve for layerwise pruning ratio allocation for multimodal models:

$$\{\alpha_i^*, \forall 0 \le i \le l\} = \texttt{dp\_solver}(\{(\alpha_{i,k}, \delta_{i,k}), \forall 0 \le i \le l, 0 \le k \le K\}) \tag{12}$$

where K indicates the number of possible discrete pruning rate selections configured as a global constant for all layers, and l is the total number of layers. Algo. 1 shows the holistic pipeline for the proposed pruning scheme, where the calibration set $\mathcal{D}_c(X) \subset \mathcal{D}_t(X)$ is randomly sampled from the training set.

3 Experimentation Details



(a) (b)

Figure 2: Fisher information values proportions throughout the training phase for both (a) BEVfusion [34] and (b) RPVNet [53] models, carried out using the nuScenes [4] SemanticKITTI [3] datasets.

Modality	Dataset	Model	Unpruned	LAMP [43]	ProsPrune [2]	RD Prune [54]	**MMPrune4U**	Sparsity
L+C	N(T)	MSeg3D [30]	73.3	60.09	66.92	70.18	74.89	75%
	S(V)		72.9	61.28	65.96	70.63	74.37	
	N(V)	PMF [54]	76.9	62.12	69.47	72.97	78.82	
	S(V)		63.9	50.37	57.73	60.73	65.56	
L+R	N(V)	CPGNet [51]	76.9	66.68	69.53	74.72	80.11	
	S(T)		68.3	60.49	64.56	65.56	71.13	
	S(T)	RVPNet [53]	70.3	58.97	65.66	68.34	73.09	
	N(V)		77.6	65.22	73.08	74.97	79.93	
L+C	N(T)	MSeg3D [30]	73.3	46.12	55.04	66.86	71.92	83%
	S(V)		72.9	48.96	56.08	68.38	71.87	
	N(V)	PMF [54]	76.9	61.53	61.53	72.59	75.64	
	S(V)		63.9	41.53	48.71	59.77	62.58	
	S(T)	PointPainting [47]	69.86	43.6	53.49	63.44	68.29	
L+R	N(V)	CPGNet [51]	76.9	58.67	66.42	69.04	75.34	
	S(T)		68.3	50.45	58.57	61.91	66.56	
	N(V)	RVPNet [53]	77.6	68.0	69.18	70.32	76.12	
	S(T)		70.3	53.11	61.93	63.28	68.99	

Table 1: Comparison of different pruning strategy vs. MMPrune4U with several segmentation models using multimodal inputs (L+C or L+R) evaluated on nuScenes [4] [validation set "N(V)", test set "N(T)"] and SemanticKITTI [3] [validation set "S(V)", test set "S(T)"] respectively.

3.1 Results

We have considered models and pruning methods for experimentation, all of which have available source code. Figure 2 shows that the proposed regularizer is able to preserve more relevant features measured by Fisher information while comparing with the baseline model without the regularizer. Apparently it also helps to improve the test accuracy (marked by green) evidenced using RPVNet [53] and BEVFusion [34] models experimented on SemanticKITTI [3] and nuScenes [4] datasets respectively.

Table 1 shows the performance of different segmentation models measured using IoU metric and evaluated on the nuScenes dataset, with results reported for the test set "N(T)" and validation set "N(V)", as well as on the SemanticKITTI dataset, with results reported for the test set "S(T)" and validation set "S(V)". It is evident that the MMPrune4U method consistently outperforms various state-of-the-art methods, including the recent RD Prune [54] approach across two different combinations of sensor modalities - LiDAR+Camera (L+C) and LiDAR+Range (L+R). This observation persists even as sparsity increases from 75% to

83%. Table 2 demonstrates the effectiveness of the proposed pruning approach over existing methods for the detection task using LiDAR+Camera (L+C) multimodal inputs evaluated on nuScenes test set (marked as "N(T)"). The enhancement achieved by MMPrune4U over existing state-of-the-art methods is considerable, and this pattern remains steadfast across various levels of sparsity during pruning. As presented in Table 3, our extensive experimentation includes 3D object detection with the recent BEVFusion model [34] using two different backbones - SwinT[33]+Voxelnet[62] and Res101[20]+P.Pillar[26] respectively. The results indicate the superiority of the proposed pruning technique among other approaches in different pruning sparsity.

Model	Unpruned	LAMP [43]	ProsPrune [2]	RD Prune [54]	**MMPrune4U**	Sparsity
BEVFusion [34]	71.3	53.01	62.04	67.53	72.23	
PointPainting [47]	46.6	29.7	38.42	40.24	47.33	
DeepInteraction [58]	70.8	50.1	61.63	62.43	69.92	77%
PointAugmenting [49]	68.8	49.5	57.97	59.82	69.56	
BEVFusion [34]	71.3	42.16	53.82	62.23	71.23	
PointPainting [47]	46.6	13.62	27.41	35.92	45.13	83%
DeepInteraction [58]	70.8	33.1	49.17	61.32	69.77	
PointAugmenting [49]	68.8	32.43	44.87	57.02	66.62	

Table 2: MMPrune4U vs. existing pruning techniques for object detection models using multimodal inputs (L+C) evaluated on nuScenes test set [4].

Backbones	SwinT [33]+Voxelnet[62]				Res101[20]+P.Pillar[26]			
Sparsity	77.5%		89.8%		76.6%		87.7%	
Metrics	mAP	NDS	mAP	NDS	mAP	NDS	mAP	NDS
Unpruned [34]	68.5	71.4	68.5	71.4	53.6	60.6	53.6	60.6
Iterative [18]	60.1	61.9	39.7	40.1	50.1	56.7	42.9	48.5
SynFlow [46]	63.2	65.7	46.3	48.0	50.8	57.4	44.4	50.6
GraSP [48]	63.3	65.9	47.9	49.6	51.1	58.0	44.7	50.7
ProsPr [2]	64.5	67.8	56.4	57.6	51.4	57.9	45.7	51.7
CrossPrune [39]	66.9	69.5	61.8	64.2	52.3	59.3	49.0	55.5
MMPrune4U	69.18	72.23	67.41	69.87	53.9	61.29	50.0	56.41

Table 3: Comparative analysis of BEVFusion models for 3D object detection assessed on nuScenes validation dataset [4] measured using mAP and NDS.

Modality	Model	Unpruned	LAMP [43]	ProsPrune [2]	RD Prune [54]	**MMPrune4U**	Sparsity
C	BevFormerV2 [56]	41.2	15.38	27.1	34.89	40.97	
	DETR3D [51]	55.6	34.5	43.9	49.93	54.9	70%
L	PointPillar [47]	65.5	36.6	48.54	59.74	64.22	
	CenterPoint [59]	60.3	30.92	42.98	53.44	58.86	
C	BevFormerV2 [56]	41.2	22.84	29.07	36.62	42.23	
	DETR3D [51]	55.6	41.73	44.22	51.93	56.79	62%
L	PointPillar [47]	65.5	48.23	50.94	61.91	66.48	
	CenterPoint [59]	60.3	40.92	43.67	55.47	61.22	

Table 4: Effectiveness of MMPrune4U with different sparsity in pruning unimodal model for object detection with solely LiDAR or camera models assessed on nuScenes test set [4].

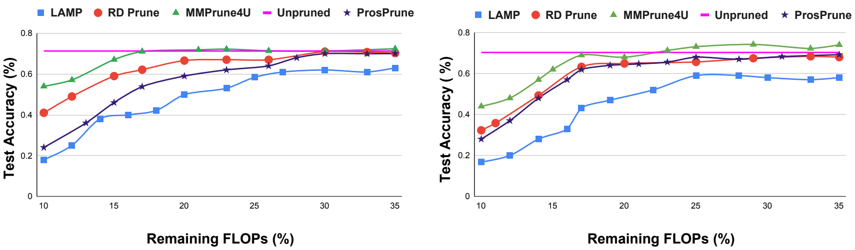


Figure 3: Different pruning methods to reduce the parameters of RVPNet [53] (left) and BEVfusion [34] (right) models.

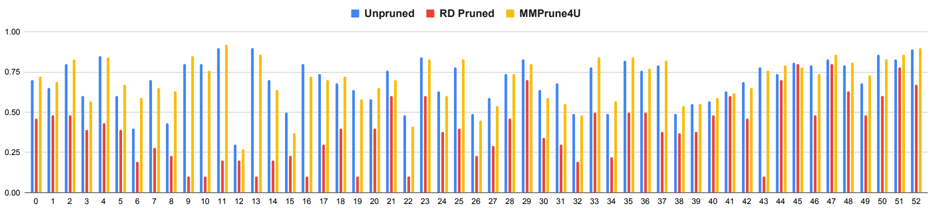


Figure 4: Layerwise Fisher information of RVPNet [53].

The proposed pruning technique can be adapted to accommodate unimodal input as well. Table 4 showcases the substantial margin of improvement achieved when utilizing solely LiDAR or Camera input with the recent BEV models, evidenced in two different pruning sparsity for both modalities separately.

Figure 1 provides a visual comparison between RD Prune [54] and MMPrune4U with respect to ground truth. The top and bottom rows display BEV semantic segmentation and object detection inferences, respectively. In terms of segmentation performance, MMPrune4U accurately preserves predictions of buildings (marked by yellow) that are false negatives by RD Prune. With respect to the detection task, two pedestrians (marked by blue) are missed by RD Prune but correctly preserved by the proposed pruning technique. Generally, MMPrune4U exhibits reduced susceptibility to generating false positives. One of the main aspects of pruning is to achieve comparable performance with fewer FLOPs. Figure 3 shows the Pareto-frontier of test accuracy vs. FLOPs while using various state-of-the-art pruning techniques including MMPrune4U on RVPNet [53] (left) using SemanticKITTI [3] and BEVfusion [34] (right) model using nuScenes [4] datasets. The frontier line at left shows that the proposed pruning method achieves the same performance remarkably with only 17% of the FLOPs of the unpruned model. For the other pattern, with just 23% FLOPs of the unpruned model, MMPrune4U can match the same performance. Notably, even with 17% of the FLOPs, the proposed pruning method delivers competitive performance compared to the unpruned model, and the highest performance, surpassing even the unpruned model, is achieved with only 29% of the FLOPs.

In network pruning, it is essential to verify if the layers in the pruned model can provide comparable information. We calculate the Fisher information for each layer in the unpruned model (RVPNet [53]) and compare it with two pruned models: one generated using the latest RD Prune [54] method and another employing MMPrune4U. Figure 4 presents a detailed analysis in which MMPrune4U demonstrates information levels that are nearly on par with the reference model across layers, surpassing the RD Prune method by a significant margin.

4 Conclusion

While deep learning models excel in automotive applications, their excessive parameters hinder their deployment on embedded devices due to computational constraints. In this work, we introduce a novel regularizer based on the log-Sobolev inequality, integrating the properties of functional Fisher information and functional entropy to minimize feature distortion during pruning across layers. By considering layer-wise sensitivity and optimizing with dynamic programming, our approach outperforms existing methods, as validated through extensive experiments using different pruning methods applied on various state-of-the-art models with multiple pruning sparsity in both multimodal and unimodal setup on complex automotive datasets. Our ablation study underscores the effectiveness of the proposed regularizer in mitigating feature distortions present in the pruned network. In future study, we plan to address the issue of modality imbalance in the context of multimodal network pruning.

References

[1] Simegnew Yihunie Alaba and John E Ball. Transformer-based optimized multimodal fusion for 3d object detection in autonomous driving. IEEE Access, 2024.

[2] Milad Alizadeh, Shyam A Tailor, Luisa M Zintgraf, Joost van Amersfoort, Sebastian Farquhar, Nicholas Donald Lane, and Yarin Gal. Prospect pruning: Finding trainable weights at initialization using meta-gradients. In *International Conference on Learning Representations*, 2021.

[3] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9297–9307, 2019.

[4] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[5] Hongxiang Cai, Zeyuan Zhang, Zhenyu Zhou, Ziyin Li, Wenbo Ding, and Jiuhua Zhao. Bevfusion4d: Learning lidar-camera fusion under bird's-eye-view via cross-modality guidance and temporal aggregation. *arXiv preprint arXiv:2303.17099*, 2023.

[6] Tianlong Chen, Jonathan Frankle, Shiyu Chang, Sijia Liu, Yang Zhang, Zhangyang Wang, and Michael Carbin. The lottery ticket hypothesis for pre-trained bert networks. *Advances in neural information processing systems*, 33:15834–15846, 2020.

[7] Tianlong Chen, Jonathan Frankle, Shiyu Chang, Sijia Liu, Yang Zhang, Michael Carbin, and Zhangyang Wang. The lottery tickets hypothesis for supervised and self-supervised pre-training in computer vision models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16306–16316, 2021.

[8] Sanjoy Chowdhury, Subhrajyoti Dasgupta, Sudip Das, and Ujjwal Bhattacharya. Listen to the pixels. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 2568–2572. IEEE, 2021.

[9] Arindam Das, Sudip Das, Ganesh Sistu, Jonathan Horgan, Ujjwal Bhattacharya, Edward Jones, Martin Glavin, and Ciarán Eising. Revisiting modality imbalance in multimodal pedestrian detection. In *2023 IEEE International Conference on Image Processing (ICIP)*, pages 1755–1759. IEEE, 2023.

[10] Sudip Das, Perla Sai Raj Kishore, and Ujjwal Bhattacharya. An end-to-end framework for pose estimation of occluded pedestrians. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 1446–1450. IEEE, 2020.

[11] Kinjal Dasgupta, Arindam Das, Sudip Das, Ujjwal Bhattacharya, and Senthil Yogamani. Spatio-contextual deep network-based multimodal pedestrian detection for autonomous driving. *IEEE transactions on intelligent transportation systems*, 23(9): 15940–15950, 2022.

[12] Utku Evci, Trevor Gale, Jacob Menick, Pablo Samuel Castro, and Erich Elsen. Rigging the lottery: Making all tickets winners. In *International Conference on Machine Learning*, pages 2943–2952. PMLR, 2020.

[13] Matteo Farina, Massimiliano Mancini, Elia Cunegatti, Gaowen Liu, Giovanni Iacca, and Elisa Ricci. Multiflow: Shifting towards task-agnostic vision-language pruning. *arXiv preprint arXiv:2404.05621*, 2024.

[14] Trevor Gale, Erich Elsen, and Sara Hooker. The state of sparsity in deep neural networks. *arXiv preprint arXiv:1902.09574*, 2019.

[15] Xue Geng, Zhe Wang, Chunyun Chen, Qing Xu, Kaixin Xu, Chao Jin, Manas Gupta, Xulei Yang, Zhenghua Chen, Mohamed M Sabry Aly, et al. From algorithm to hardware: A survey on efficient and safe deployment of deep neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 2024.

[16] Leonard Gross. Logarithmic sobolev inequalities. *American Journal of Mathematics*, 97(4):1061–1083, 1975.

[17] Jinyang Guo, Wanli Ouyang, and Dong Xu. Multi-dimensional pruning: A unified framework for model compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1508–1517, 2020.

[18] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

[19] Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28, 2015.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[21] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17853–17862, 2023.

[22] Bin Huang, Yangguang Li, Enze Xie, et al. Fast-bev: Towards real-time on-vehicle bird's-eye view perception. *arXiv preprint arXiv:2301.07870*, 2023.

[23] Berivan Isik, Tsachy Weissman, and Albert No. An information-theoretic justification for model pruning. In *International Conference on Artificial Intelligence and Statistics*, pages 3821–3846. PMLR, 2022.

[24] Neil Kichler, Sher Afghan, and Uwe Naumann. Towards sobolev pruning. *arXiv preprint arXiv:2312.03510*, 2023.

[25] Perla Sai Raj Kishore, Sudip Das, Partha Sarathi Mukherjee, and Ujjwal Bhattacharya. Cluenet: A deep framework for occluded pedestrian pose estimation. In *BMVC*, page 245, 2019.

[26] Alex H Lang et al. Pointpillars: Fast encoders for object detection from point clouds. In *IEEE/CVF CVPR*, pages 12697–12705, 2019.

[27] Jaeho Lee, Sejun Park, Sangwoo Mo, Sungsoo Ahn, and Jinwoo Shin. Layer-adaptive sparsity for the magnitude-based pruning. *arXiv preprint arXiv:2010.07611*, 2020.

[28] Jong-Ryul Lee and Yong-Hyuk Moon. Rethinking group fisher pruning for efficient label-free network compression. In *BMVC*, page 693, 2022.

[29] Hongyang Li, Chonghao Sima, Jifeng Dai, Wenhai Wang, Lewei Lu, Huijie Wang, Jia Zeng, Zhiqi Li, Jiazhi Yang, Hanming Deng, et al. Delving into the devils of bird's-eye-view perception: A review, evaluation and recipe. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[30] Jiale Li, Hang Dai, Hao Han, and Yong Ding. Mseg3d: Multi-modal 3d semantic segmentation for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21694–21704, 2023.

[31] Xiaoyan Li, Gang Zhang, Hongyu Pan, and Zhenhua Wang. Cpgnet: Cascade point-grid fusion network for real-time lidar semantic segmentation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 11117–11123. IEEE, 2022.

[32] Zhenxin Li, Shiyi Lan, Jose M Alvarez, and Zuxuan Wu. Bevnext: Reviving dense bev frameworks for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20113–20123, 2024.

[33] Tingting Liang, Xiaojie Chu, Yudong Liu, Yongtao Wang, Zhi Tang, Wei Chu, Jingdong Chen, and Haibin Ling. Cbnet: A composite backbone network architecture for object detection. *IEEE Transactions on Image Processing*, 31:6893–6906, 2022.

[34] Tingting Liang, Hongwei Xie, Kaicheng Yu, Zhongyu Xia, Zhiwei Lin, Yongtao Wang, Tao Tang, Bing Wang, and Zhi Tang. Bevfusion: A simple and robust lidar-camera fusion framework. *Advances in Neural Information Processing Systems*, 35:10421–10434, 2022.

[35] Liyang Liu, Shilong Zhang, Zhanghui Kuang, Aojun Zhou, Jing-Hao Xue, Xinjiang Wang, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. Group fisher pruning for practical network compression. In *International Conference on Machine Learning*, pages 7021–7032. PMLR, 2021.

[36] Youquan Liu, Runnan Chen, Xin Li, Lingdong Kong, Yuchen Yang, Zhaoyang Xia, Yeqi Bai, Xinge Zhu, Yuexin Ma, Yikang Li, et al. Uniseg: A unified multi-modal lidar segmentation network and the openpcseg codebase. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21662–21673, 2023.

[37] Zhijian Liu, Haotian Tang, Alexander Amini, Xingyu Yang, Huizi Mao, Daniela Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[38] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE international conference on robotics and automation (ICRA)*, pages 2774–2781. IEEE, 2023.

[39] Yantao Lu, Bo Jiang, Ning Liu, Yilan Li, Jinchao Chen, Ying Zhang, and Zifu Wan. Crossprune: Cooperative pruning for camera-lidar fused perception models of autonomous driving. page 111522. Elsevier, 2024.

[40] Jian-Hao Luo, Jianxin Wu, and Weiyao Lin. Thinet: A filter level pruning method for deep neural network compression. In *Proceedings of the IEEE international conference on computer vision*, pages 5058–5066, 2017.

[41] Arun Mallya, Dillon Davis, and Svetlana Lazebnik. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *Proceedings of the European conference on computer vision (ECCV)*, pages 67–82, 2018.

[42] Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. Pruning convolutional neural networks for resource efficient inference. In *International Conference on Learning Representations*, 2016.

[43] Pavlo Molchanov, Arun Mallya, Stephen Tyree, Iuri Frosio, and Jan Kautz. Importance estimation for neural network pruning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11264–11272, 2019.

[44] Ari Morcos, Haonan Yu, Michela Paganini, and Yuandong Tian. One ticket to win them all: generalizing lottery ticket initializations across datasets and optimizers. *Advances in neural information processing systems*, 32, 2019.

[45] Ziying Song, Lei Yang, Shaoqing Xu, Lin Liu, Dongyang Xu, Caiyan Jia, Feiyang Jia, and Li Wang. Graphbev: Towards robust bev feature alignment for multi-modal 3d object detection. *arXiv preprint arXiv:2403.11848*, 2024.

[46] Hidenori Tanaka, Daniel Kunin, Daniel L Yamins, and Surya Ganguli. Pruning neural networks without any data by iteratively conserving synaptic flow. *Advances in neural information processing systems*, 33:6377–6389, 2020.

[47] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential fusion for 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4604–4612, 2020.

[48] Chaoqi Wang, Guodong Zhang, and Roger Grosse. Picking winning tickets before training by preserving gradient flow. In *International Conference on Learning Representations*, 2019.

[49] Chunwei Wang, Chao Ma, Ming Zhu, and Xiaokang Yang. Pointaugmenting: Cross-modal augmentation for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11794–11803, 2021.

[50] Maolin Wang, Yao Zhao, Jiajia Liu, Jingdong Chen, Chenyi Zhuang, Jinjie Gu, Ruocheng Guo, and Xiangyu Zhao. Large multimodal model compression via efficient pruning and distillation at antgroup. *arXiv preprint arXiv:2312.05795*, 2023.

[51] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning*, pages 180–191. PMLR, 2022.

[52] Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21729–21740, 2023.

[53] Jianyun Xu, Ruixiang Zhang, Jian Dou, Yushi Zhu, Jie Sun, and Shiliang Pu. Rpvnet: A deep and efficient range-point-voxel fusion network for lidar point cloud segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16024–16033, 2021.

[54] Kaixin Xu, Zhe Wang, Xue Geng, Min Wu, Xiaoli Li, and Weisi Lin. Efficient joint optimization of layer-adaptive weight pruning in deep neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17447–17457, 2023.

[55] Runxin Xu, Fuli Luo, Chengyu Wang, Baobao Chang, Jun Huang, Songfang Huang, and Fei Huang. From dense to sparse: Contrastive pruning for better pre-trained language model compression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11547–11555, 2022.

[56] Chenyu Yang, Yuntao Chen, Hao Tian, et al. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. In *IEEE/CVF CVPR*, pages 17830–17839, 2023.

[57] Huanrui Yang, Hongxu Yin, Maying Shen, Pavlo Molchanov, Hai Li, and Jan Kautz. Global vision transformer pruning with hessian-aware saliency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18547–18557, 2023.

[58] Zeyu Yang, Jiaqi Chen, Zhenwei Miao, Wei Li, Xiatian Zhu, and Li Zhang. Deepinteraction: 3d object detection via modality interaction. *Advances in Neural Information Processing Systems*, 35:1992–2005, 2022.

[59] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11784–11793, 2021.

[60] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Multimodal virtual point 3d detection. *Advances in Neural Information Processing Systems*, 34:16494–16507, 2021.

[61] Yuanxin Zhong, Minghan Zhu, and Huei Peng. Vin: Voxel-based implicit network for joint 3d object detection and segmentation for lidars. *BMVC 2023*, 2021.

[62] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *IEEE CVPR*, pages 4490–4499, 2018.

[63] Michael Zhu and Suyog Gupta. To prune, or not to prune: exploring the efficacy of pruning for model compression. *arXiv preprint arXiv:1710.01878*, 2017.

[64] Zhuangwei Zhuang, Rong Li, Kui Jia, Qicheng Wang, Yuanqing Li, and Mingkui Tan. Perception-aware multi-sensor fusion for 3d lidar semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16280–16290, 2021.