Supplementary Material
GliaNet: Adaptive Neural Network Structure Learning with Glia-Driven

Mengqiao Han[1]   Liyuan Pan[*,2]   Xiabi Liu[*,2]

[1] College of Information Engineering, Northwest Agricultural and Forestry (A&F) University
[2] School of computer science, Beijing Institute of Technology

hanmq@nwafu.edu.cn, {liyuan.pan, liuxiabi}@bit.edu.cn

Abstract

In this supplementary material, we provide more details of our GliaNet in Sec. 1, more ablation studies and discussions in Sec. 2, and more experiments and visualization results are shown in Sec. 3. We also provide the source code at: https://github.com/hanmengqiaoyx/GliaNet.

1. Implementation and Training Details

Our GliaNet consists of a Neural Module (NM) and a Glia Module (GM) that optimizes the communication of NM, where the Glia is composed of Oligodendrocyte (Oli) and Astrocyte (Ast). In the GM, Oli identifies the winning neurons to participate in subsequent propagation, and their weights are modulated by Ast.

To apply the contrastive learning loss on the GM, *i.e.*, Oli with $\Phi = \{\phi_{\mathcal{H}}, \phi_g, \phi_{\mathcal{U}}\}$ and Ast with $\Phi = \{\theta_{\mathcal{H}}, \theta_f, \theta_{\mathcal{U}}\}$, Take the interaction between neuron states and Oli as an example, we first build the positive pairs. Followed by different clipping, grayscale and affine transformations on the training data, we generated matching pairs from the training data and then obtained the two views of neuron states $\mathbf{X}$ and $\widehat{\mathbf{X}}$ for NM. By feeding $\mathbf{X}$ and $\widehat{\mathbf{X}}$ to our GM, we obtained $\mathbf{X}^* = \{w_k\}_{k=1}^M$ and $\widehat{\mathbf{X}}^* = \{w_{M+k}\}_{k=1}^M$. Note, x_k and x_{M+k} are regarded as a positive pair. Denote $\mathbf{z} = c(\text{Cancat}(\mathbf{X}^*, \widehat{\mathbf{X}}^*))$ as a normalized projection, we finally get the comparative learning loss (Eq. (11) in our main paper).

$$\mathcal{L}_k^{\text{reg}}(\Phi, \Theta) = -\sum_{i=1}^N \log \frac{\exp(z_i \cdot z_{i+N})/\tau}{\sum_{p=1}^{2N} \mathbb{1}_{[p \neq i]} \exp(z_i \cdot z_p)/\tau}, \tag{11}$$

where $\mathbb{1} \in \{0, 1\}$ is an indicator function evaluating to 1 if $l \neq q$ and τ is a temperature. Here, z_j is the kth element
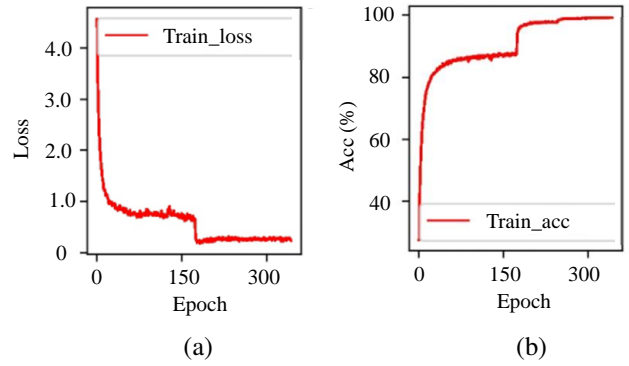
---

*corresponding authors

Figure 1. *The evolution of the GliaNet during training is illustrated in the change curve. (a) displays the variation in the difference between train loss and epoch, while (b) exhibits the fluctuation in the difference between train accuracy and epoch.*
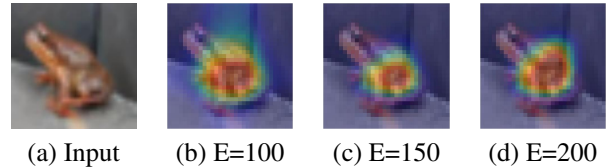

(a) Input   (b) E=100   (c) E=150   (d) E=200

Figure 2. *We visualize the interaction between Glia and neurons at different epochs (E) during training on images (a).*

of $\mathbf{z}$. The $\mathcal{L}_k^{\text{reg}}(\Phi, \Theta)$ encourages Oli in the kth Glia unit, denoted as $\mathbf{O}_k$, to learn similar representations of different neuron states features from the same neuron state $\mathbf{X}$ of NM. Similarly, the relationship between Ast, $\mathbf{A}_k$, and winning neuron weights follows a comparable pattern.

Fig. 1 depicts the change curve when employing NM as ResNet50 on imageNet-1k [4]. The decreasing loss curve signifies the convergence achieved by our training scheme. The GliaNet undergoes training using the SGD optimizer with a learning rate of $1e$-1.

We also provide a clearer explanation and visualization to elucidate the role of the interaction between Glia and neurons in optimizing GliaNet. Fig. 2 illustrates the increasing

Table 1. *Comparison of different sizes of parameters of the Neural Module (NM) in GliaNet on CIFAR10. We set the NM as ResNet18 with different sizes. Taking '64-128-256-512' as an example, it denotes the output of the four dense blocks, respectively. We highlight the best and the second-best numbers in* **bold** *and* **bold**, *respectively.*

Pattern	Architecture	Acc (%)	Params (M)	Time Cost (GPU days)
Tiny-ResNet18	64-128-128-256	96.60	**6.15**	**0.06**
Orgn-ResNet18	64-128-256-512	96.72	6.72	0.11
Large-ResNet18	64-256-256-512	**96.86**	7.00	0.14

Table 2. *Experiments on classification with different joint functions $\mathcal{U}()$ on CIFAR10.*

Manners	Optimization for Glia units	Interaction	Acc (%)	Params (M)
ResNet18 [6]	-	-	94.96	11.17
Case 1	Independent	Current	96.57	7.35
Case 2	Independent	History	96.59	7.38
Case 3	Chain	Current	96.81	**7.00**
Case 4 (Ours)	Chain	History	**96.86**	**7.00**

Table 3. *Experiments on classification with different adaptive functions $\mathcal{H}()$ on CIFAR10.*

Manners	$\mathcal{H}()$	Acc (%)	Params (M)
ResNet18 [6]	-	94.96	11.17
Case 1	Compression	96.68	7.35
Case 2	Coarse-global compression	96.82	7.06
Case 3 (Ours)	Fine-global compression	**96.86**	**7.00**

epoche (E) of Glia-Neuron interactions, leading to a heightened focus on target regions through the discovery of an optimal sub-network and its parameters by the Glia.

2. More Ablation Studies and Discussions

This section discusses the rationality of our method, wherein the Neural Module (NM) in GliaNet is consistently configured as ResNet18 across all experiments and evaluated on the CIFAR10 dataset.

2.1. The Number of Parameters

Regarding the varying parameters, the capacity of our GliaNet is intricately related to the size of the selected NM. To this end, we present the results with different numbers of parameters (Tiny, Organ, and Large) based on ResNet18 in Tab. 1. Our method consistently demonstrates competitive performance even when employing the tiny ResNet18, where the capacity is significantly reduced.

Table 4. *The accuracy of our GliaNet with respect to the number of NM neurons from 25% to 100% that participate in the interaction with Glia.*

Num of Neurons	ResNet18	25%	50%	75%	100%
Acc (%)	94.96	95.50	96.25	96.72	**96.86**
Params (M)	11.17	9.86	8.28	7.75	**7.00**

2.2. The architecture of U() in GliaNet

The joint function $\mathcal{U}()$ facilitates collaborations in the Glia, *i.e.*, $\mathbf{G}_{\{M\}}$. We conduct progressive ablation experiments with four cases to validate the effectiveness of our chain optimization strategy. Case 1: Independent optimization for multiple Glia units, where each Glia unit independently optimizes the communication of a subset of pre-neurons. Case 2: Independent optimization of Glia units based on their output at different iterations. Case 3: Chain optimization for a group of Glia units to establish interactions in $\mathbf{G}_{\{M\}}$. Case 4 (Ours): Interaction based on the output of Glia units at different iterations. Tab. 2 shows the performance with four cases. Note that all cases achieve better accuracy compared to the ResNet18 baseline, and our designed chain optimization achieves the best performance.

2.3. The architecture of H() in GliaNet

We perform incremental ablation experiments on the integration function $\mathcal{H}()$ with three cases to verify the necessity of making Oli- and Ast-input global. Case 1: Compression only, where connections in the compressed weights are considered irrelevant. Case 2: Coarse-global compression, involving the establishment of correlations among all connections in the weight during compression. Case 3 (Ours): Fine-global compression, which entails compression while initially establishing correlations between the input connections of Oli and Ast, followed by correlating the input of every Oli and Ast. Tab. 3 reports the results, indicating superior performance with Fine-global compression. The process of establishing correlations between connections exhibits a significant improvement in the optimized capacity of Glia.

2.4. Number of Interactions Neurons

We investigate the results of randomly selecting 25%, 50%, 75% and 100% of neurons in NM to establish the interaction with Glia, while the remaining neurons maintain one-direction propagation. Tab. 4 reports that as more NM neurons establish interactions with Glia, the adaptive ability of GliaNet improves, resulting in increased accuracy. Fig. 3 illustrates that the more neurons in NM participate in the interaction, the more attention of NM covers the target region on the image. The visualization follows the Grad-CAM [13].
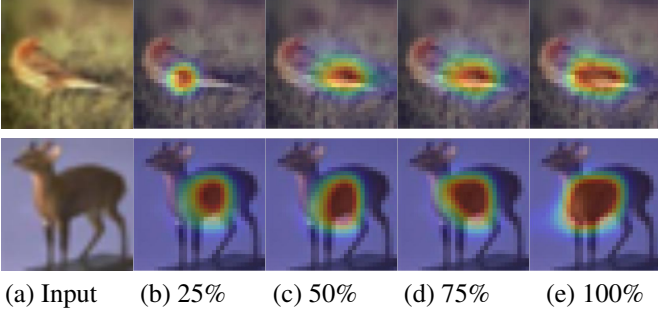
(a) Input   (b) 25%   (c) 50%   (d) 75%   (e) 100%

Figure 3. *We display the attention of NM on images (a) with different percentages of NM neurons that are involved varying from 25% to 100% (b)-(e).*
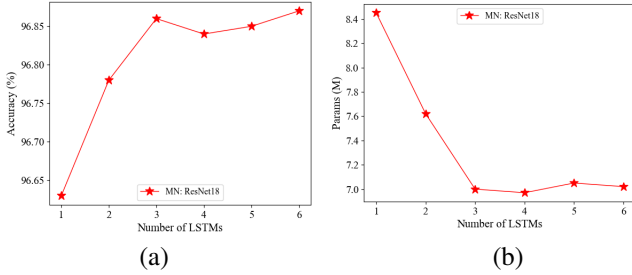

(a)   (b)

Figure 4. *Comparison of accuracy and parameters concerning different numbers of LSTM stacking in each Oli and Ast on CIFAR10. We set NM in our GliaNet to ResNet18. (a) Results of Acc. (b) The final NM parameters.*

2.5. Number of LSTM in each Oli and Ast

As the number of Glia increases, the input of each Oli and Ast will decrease, allowing us to design their capacity to be smaller. Considering the time cost, we set the number of Glia to 4, *i.e.*, $\mathbf{G}_{\{4\}} = \{\mathbf{O}_{\{4\}}, \mathbf{A}_{\{4\}}\}$, in the main paper. Building upon this configuration, we further study the impact of varying amounts of LSTM contained in the Oli and Ast on accuracy and final parameters. We set NM in our GliaNet to ResNet18 and evaluated on CIFAR10. Fig. 4a illustrates the relationship between test accuracy and the numbers of LSTM, and Fig. 4b displays the parameters concerning the number of LSTM. We can observe that the performance has stabilized using three LSTMs to construct each Oli and Ast.

Fig. 5 depicts the transfer of attention regions on the input image with NM (ResNet18) under the stack of different numbers of LSTM within each Oli and Ast. By increasing the number of LSTM among them, the receptive field of NM for the target region on the input image gradually expands, further enhancing the valuable information that the NM focuses on. When the number of LSTM is 3, NM's attention on the image primarily centers on the target region.

Table 5. *The accuracy of our GliaNet with respect to different values of the contrastive loss coefficient λ in Eq. (11).*

Hyperparameters λ	Acc (%)	Params (M)
$\lambda = 1e-1$	**96.86**	**7.00**
$\lambda = 1e-2$	**96.80**	**7.06**
$\lambda = 1e-3$	96.75	7.18
$\lambda = 0$	96.68	7.32

Table 6. *Comparison with the tate-of-the-art NAS methods with contrastive learning regularization term in their evaluation strategy. GliaNet and NAS architectures are searched and evaluated on CIFAR10. Time cost only denotes the search time using our method and the 'CL + NAS' methods. Ours (DenseNet-BC) denotes that the NM in our GliaNet is set to DenseNet-BC.*

Architecture	Acc (%)	Params (M)	Time Cost (GPU days)
DenseNet-BC [8]	96.54	25.60	-
CL + NAS-LID [7]	97.59	**6.87**	**2.87**
CL + HOTNAS [17]	**97.62**	4.**72**	3.96
Ours (DenseNet-BC)	**97.82**	7.49	**0.70**

For example, the 1st row in Fig. 5 labeled as 'Bird', the optimization of Glia in the first ($Q = 1$), the Glia increases the attention of NM on the input image. Then, Glia gradually guides NM to diffuse attention to other locations within the target region.

2.6. Hyperparameter Selections

Hyper-parameter selections are typically conducted on small datasets and subsequently applied to all datasets to verify their scalability [1]. Accordingly, we perform the experiment on CIFAR10. The NM in our GliaNet is set to ResNet18, with the remaining settings identical to those in the main paper.

We study λ in Eq. (11) in the main paper, where λ is the coefficient of contrastive loss. Tab. 5 reports that our method achieves the best accuracy when setting $\lambda = 1e$-1. Notably, even without the contrastive loss, our results (95.68%) surpass those of the State-of-the-Art (SOTA) methods CDS [19] and AstroNet [5] (95.49% and 96.52%, respectively, as shown in Tab. 1, main paper) on CIFAR10.

2.7. The Effectiveness of Contrastive Learning

We allow the SOTA Neural Architecture Search (NAS) methods, NAS-LID [7] and HOTNAS [17], to use the contrastive learning (CL) regularization term [19] in their evaluation strategies, to more fairly compare the performance of our method with 'CL + NAS-LID' and 'CL + HOTNAS'.

Tab. 6 reports the performance results of our GliaNet and 'CL + NAS'. Both of them are searched and evaluated on CIFAR10 [9]. Compared with 'CL + NAS-LID' and 'CL

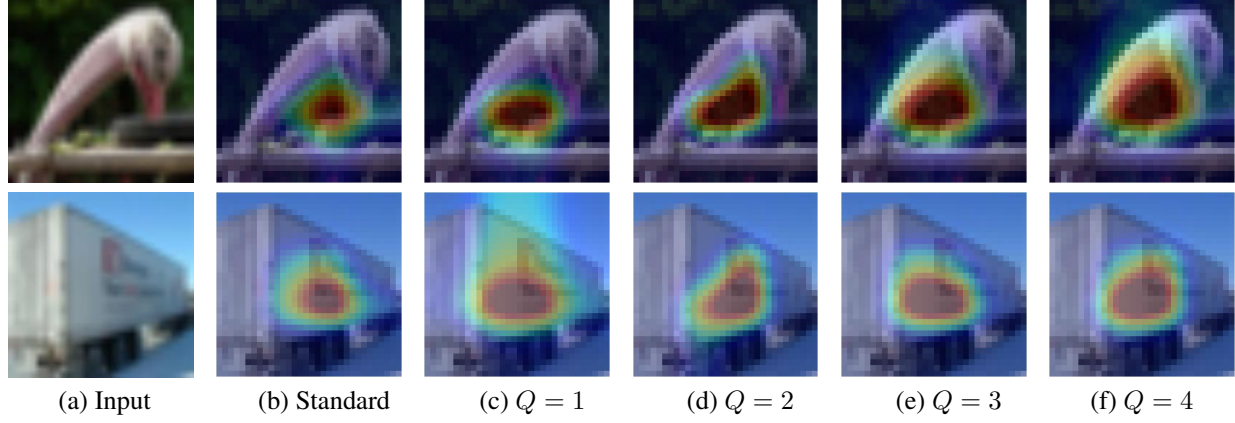|					
(a) Input	(b) Standard	(c) $Q = 1$	(d) $Q = 2$	(e) $Q = 3$	(f) $Q = 4$

Figure 5. *We display the NM's (ResNet18 on CIFAR10) attention, optimized by our proposed Glia, on the input image (a) with different numbers of LSTM varying from 1 to 4 (c)-(f). From the $1^{st} - 2^{nd}$ rows, the labels are 'Bird' and 'Truck'. (b) Standard ResNet18. The attention map adaptively focuses on the target region. Q denotes the number of LSTM within each Oli and Ast.*

Table 7. *Experiments on classification with different methods on ImageNet-1k. With the same settings, and CL regularization term, compare networks constructed by our G-N model and artificial Astrocyte-Neuron model. Our (ResNet50) denote that the NM in GliaNet is set to ResNet50.*

Architecture	Acc (%)	Params (M)
ResNet50 [6]	75.30	26.11
AstroNet [5]	78.31	8.70
AstroNet* [5]	**78.54**	**8.54**
Ours⁻ (ResNet50)	78.49	**8.54**
Ours (ResNet50)	**78.98**	**8.20**

+ HOTNAS', our method improves the accuracy by 0.23% and 0.20%, respectively. In addition, we achieve a significant advantage in the time cost. Our advantages in computational resources and time costs are more obvious.

We also compare our GliaNet with AstroNet [5], with the CL regularization term, denoted as AstroNet*, and ours without the CL regularization term Ours⁻. The comparison results on ImageNet-1k are demonstrated in Tab. 7, with the CL regularization term, our GliaNet outperforms AstroNet* by 0.44% and Ours⁻ by 0.49% in accuracy. This demonstrates the effectiveness of the group of Glia units cooperating with each other compared to a single astrocyte.

3. More Experiments and Visualization Results

3.1. Compared to SOTA Dropout and Pruning Methods

As shown in Tab 8, our method demonstrates superior performance compared to the latest early dropout and pruning techniques on the ImageNet-1k. The proposed approach not only consistently surpasses the accuracy of the top-three

Table 8. *Experiments comparing with dropout and pruning methods on ImageNet.*

Architecture	Acc (%)	Compression/Dropout Rate (%)	Change (%)
ViT-Tiny/32	76.30	-	-
+ early dropout [12]	76.37	5	↑ 0.07
+ early dropout [12]	**76.70**	10	↑ 0.40
+ early dropout [12]	76.60	**20**	↑ 0.30
Ours (ViT-Tiny)	**77.45**	**14.21**	↑ **1.15**
SWIN-S	**83.00**	-	-
NViT-H [16]	82.95	**40**	↓ 0.05
Ours (SWIN-S)	**84.02**	**38**	↑ **1.02**

early dropout methods but also exceeds the current SOTA accuracy achieved by the SWIN-S.

3.2. More SOTA Segmentation Methods

We compare our method with two masked-attention-based segmentation methods, Mask2Former [3] and MP-Former [18] on the COCO dataset [11]. Note, the backbone, *i.e.*, ResNet50, of the standard segmentation method is pre-trained on ImageNet-1k. Keeping the same as other settings in Mask2Former and MP-Former methods, we only replace ResNet50 with ResNet50-Ast and ResNet50†, which are pre-trained by AstroNet and our method.

Tab. 9 reports the comparison of our method with others on COCO. The pre-trained G-N model-based backbone (ResNet50†) with our method, compared to the second-best method, *i.e.*, ResNet50 in Mask2Former (44.3%) and AstroNet (ResNet50-Ast) in MP-Former (45.5%), our method improves the AP accuracy of 0.5% and 0.4%.

Table 9. *Experiments on segmentation with various backbones on COCO. ResNet50, ResNet50-Ast and ResNet50† pre-trained on ImageNet-1k by standard training, AstroNet and Ours.*

Model	backbone	AP	AP$_S$	AP$_M$	AP$_L$
Mask2Former [3]	ResNet50	**44.3**	**24.0**	**47.6**	65.2
AstroNet [5]	ResNet50-Ast	44.1	23.8	**47.6**	65.0
Ours	ResNet50†	**44.8**	**24.5**	**47.9**	**65.4**
MP-Former [18]	ResNet50	45.5	**25.2**	48.7	**66.2**
AstroNet [5]	ResNet50-Ast	**45.7**	25.0	**48.8**	**66.5**
Ours	ResNet50†	**45.9**	**25.6**	**49.1**	**66.5**

Table 10. *Experiments on object detection with various backbones on COCO. ResNet50, ResNet50-Ast and ResNet50† pre-trained on ImageNet-1k by standard training, AstroNet and Ours.*

Model	backbone	AP	AP$_S$	AP$_M$	AP$_L$
DEQDet [15]	ResNet50	48.6	**31.6**	51.8	**62.9**
AstroNet [5]	ResNet50-Ast	**48.8**	31.4	**52.0**	62.6
Ours	ResNet50†	**49.0**	**31.9**	**52.4**	**63.0**
DiffusionDet [2]	ResNet50	46.6	28.9	**49.2**	**62.1**
AstroNet [5]	ResNet50-Ast	**46.7**	**29.0**	49.0	50.0
Ours	ResNet50†	**46.9**	**29.2**	**49.5**	**61.8**

3.3. More SOTA Object Detection Methods

Tab. 10 shows our object detection performance compared with DEQDet [15] and DiffusionDet [2] on the COCO dataset. We employ the same method for acquiring backbones as in the segmentation task and then fine-tune them with identical settings for the object detection task. Experimental results indicate that utilizing the pre-trained backbone (ResNet50†) from our method results in AP improvements of 0.4% and 0.3% compared to the second-best AstroNet (ResNet50-Ast) in DEQDet (48.8%) and Diffusion-Det (46.7%), respectively.

3.4. Results on Neuromorphic Dataset

The comparison between our GliaNet and other methods on the neuromorphic dataset CIFAR10-DVS [10] is shown in Tab. 11. Our GliaNet exhibits the superior performance. For CIFAR10-DVS, our accuracy surpasses the second-best AstroNet (78.67% and 81.38%) by 0.15% and 0.12% on ResNet18 and ResNet50, respectively. Our method achieves a relatively smaller model capacity.

3.5. Results on Few-shot Dataset

We verify the effectiveness of our method on the few-shot dataset, Mini-ImageNet [14]. The Mini-ImageNet dataset is

Table 11. *Experiments on classification with different methods on neuromorphic dataset CIFAR10-DVS. With the same settings, compare networks constructed by our G-N model and other models.*

Architecture	Acc (%)	Params (M)
ResNet18 [6]	77.40	11.68
AstroNet [5]	**78.67**	**7.84**
Ours (ResNet18)	**78.82**	**7.52**
ResNet50 [6]	80.27	26.11
AstroNet [5]	**81.38**	**9.25**
Ours (ResNet50)	**81.50**	**8.15**

Table 12. *Experiments on classification with different methods on Mini-ImageNet. With the same settings, compare networks constructed by our G-N model and other models.*

Architecture	Acc (%)	Params (M)
ResNet18 [6]	51.85	11.21
AstroNet [5]	**52.56**	**7.62**
Ours (ResNet18)	**54.02**	**7.28**

widely used in the field of few-shot learning, which contains 100 classes with 600 images per class, in which the first 500 images are selected from each category for training and the remaining 100 images for testing in our experiments.

The experimental results of our method on Mini-ImageNet are shown in Tab. 12. Our GliaNet using ResNet18 as NM achieves the highest accuracy on Mini-ImageNet and outperforms AstroNet by 1.46% in accuracy. Our method reduces the capacity of ResNet18 by 35.06%. This proves that Glia modulates the NM communication, which allows our GliaNet to have a good generalization performance.

3.6. More Visualization Results

We show more visualization results compared with AstroNet [5] on image classification on CIFAR10 and image segmentation on COCO to demonstrate the effectiveness of our method.

Fig. 6 shows more details of our visualization results on CIFAR10. Compared with the AstroNet, which is the artificial Astrocyte-Neuron model-based network. Our Glia-Neuron model based-GliaNet allows NM to better focus on target regions on input images, and to avoid distractions from the surrounding environment. For example, the image labeled 'Deer' in the 5th row of Fig. 6 contains antlers, our method notices this detail. The image labeled 'Horse' in the $6^{th} - 7^{th}$ rows contains riders and horses, our method pays better attention to the target region, *i.e.*, the horse.

Fig. 7 shows our qualitative segmentation comparison against AstroNet backbones, confirming our benefit. From
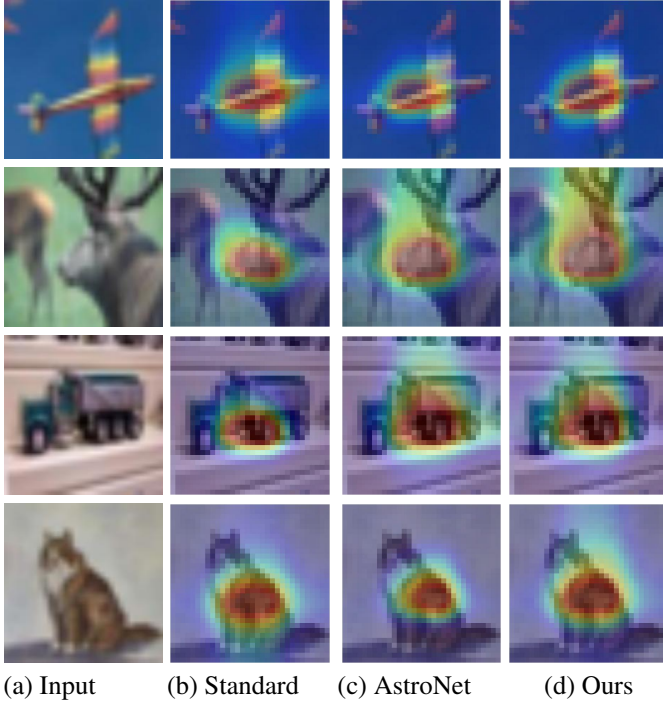
Figure 6. *We illustrate the NM's (ResNet18 on CIFAR10) attention on the input image with ResNet18 without Glia, the AstroNet and our method. From the $1^{st} - 4^{th}$ rows, the labels are 'Airplane', 'Deer', 'Truck', and 'Cat'.*

Figure 7. *We display the segmentation results on images (a) using different backbones, which are obtained by using AstroNet (b) and our method (c), respectively. Our segmentation boundary is more accurate. From the $1^{st} - 4^{th}$ rows, the labels are 'Man and Surfboard', 'Airplane', 'Cow', and 'Bear'.*

the $1^{st} - 6^{th}$ rows, we need to segment out 'Man and Surfboard', 'Cat', 'Sheep', 'Airplane', 'Cow', and 'Bear', respectively. Compared with the AstroNet, we predict boundaries more accurately. For example, for the segmentation of 'Cat', we segment the contours of the cat's ears and the chair more finely, and for the segmentation of 'Man and Surfboard', we more accurately segment the contour of the man and Surfboard.

References

[1] Mokhtar S Bazaraa, Hanif D Sherali, and Chitharanjan M Shetty. *Nonlinear programming: theory and algorithms.* John wiley & sons, 2013. 3

[2] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. Diffusiondet: Diffusion model for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19830–19843, 2023. 5

[3] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022. 4, 5

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 2009. 1
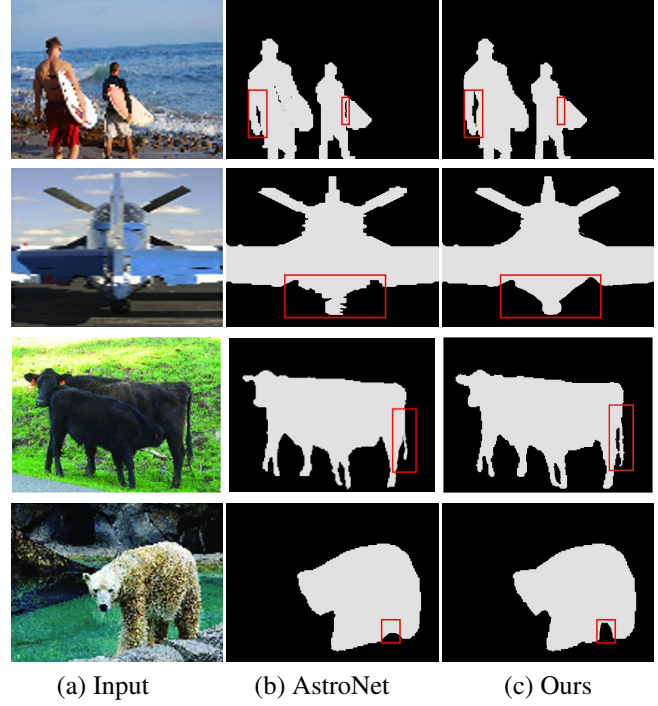
[5] Mengqiao Han, Liyuan Pan, and Xiabi Liu. Astronet: When astrocyte meets artificial neural network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 3, 4, 5

[6] 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*, 2016. 2, 4, 5

[7] Xin He, Jiangchao Yao, Yuxin Wang, Zhenheng Tang, Ka Chun Cheung, Simon See, Bo Han, and Xiaowen Chu. Nas-lid: Efficient neural architecture search with local intrinsic dimension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023. 3

[8] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017. 3

[9] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 3

[10] Hongmin Li, Hanchao Liu, Xiangyang Ji, Guoqi Li, and Luping Shi. Cifar10-dvs: an event-stream dataset for object classification. *Frontiers in neuroscience*, 11:309, 2017. 5

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence

Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014. 4

[12] Zhuang Liu, Zhiqiu Xu, Joseph Jin, Zhiqiang Shen, and Trevor Darrell. Dropout reduces underfitting. In *International Conference on Machine Learning*, pages 22233–22248. PMLR, 2023. 4

[13] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, 2017. 2

[14] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 2016. 5

[15] Shuai Wang, Yao Teng, and Limin Wang. Deep equilibrium object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6296–6306, 2023. 5

[16] 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. 4

[17] Jiechao Yang and Yong Liu. Hotnas: Hierarchical optimal transport for neural architecture search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 3

[18] Hao Zhang, Feng Li, Huaizhe Xu, Shijia Huang, Shilong Liu, Lionel M Ni, and Lei Zhang. Mp-former: Mask-piloted transformer for image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18074–18083, 2023. 4, 5

[19] Linfeng Zhang, Xin Chen, Junbo Zhang, and Runpei Dong. Contrastive deep supervision. In *European Conference on Computer Vision*, 2022. 3