ACCV 2020

clearspace
. today

EPFL

Towards Robust Fine-grained Recognition by Maximal Separation of Discriminative Features

Asian Conference of Computer Vision 2020

Krishna Kanth Nakka and Mathieu Salzmann

Contributions

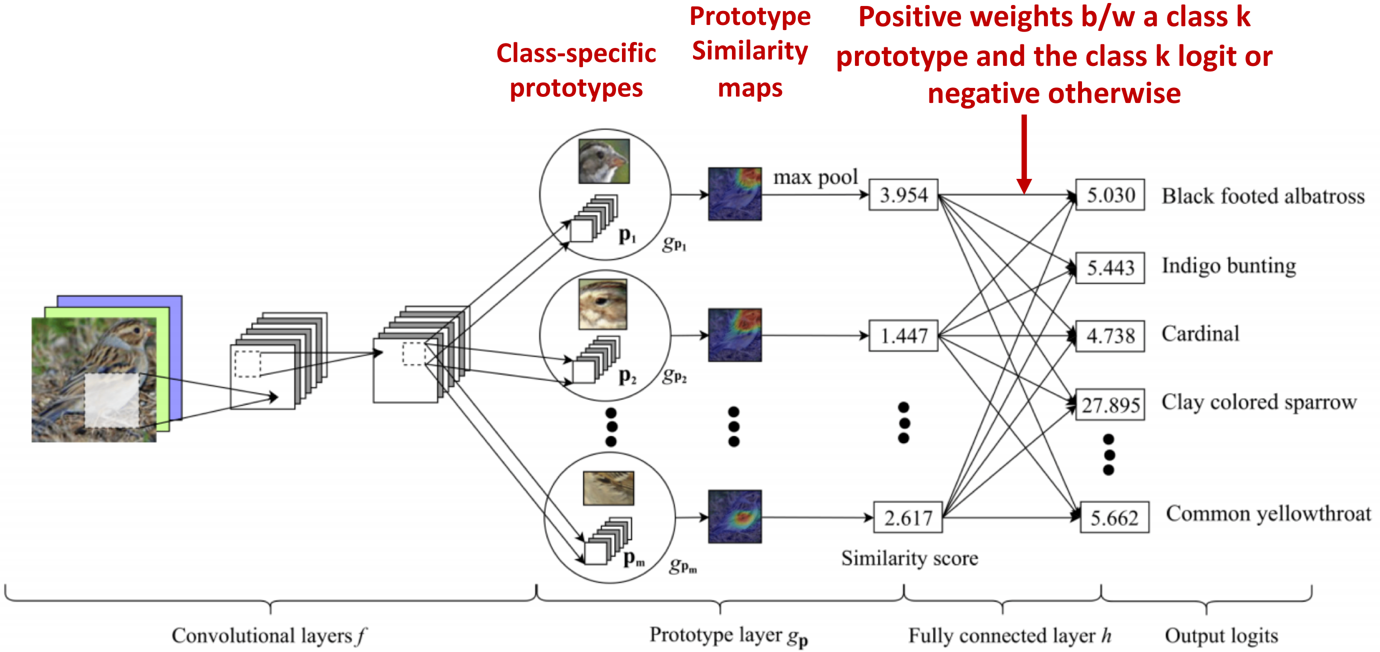
1. Interpreting Adversarial Attacks:

We analyze and explain the decisions of fine-grained recognition networks by studying the image regions responsible for classification for both clean and adversarial examples.

2. Adversarial Defense:

We design an interpretable, attention-aware network for robust fine-grained recognition by constraining the latent space of discriminative regions.

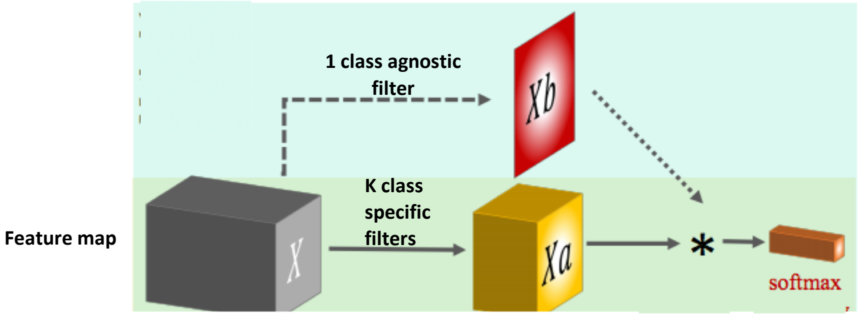
Modules: Interpretable Fine-grained Network



ProtoPNet Architecture

Chaofan Chen et al. This Looks Like That: Deep Learning for Interpretable Image Recognition, NIPS 2019

Modules: Interpretable Fine-grained Network



Attention Pooling Architecture

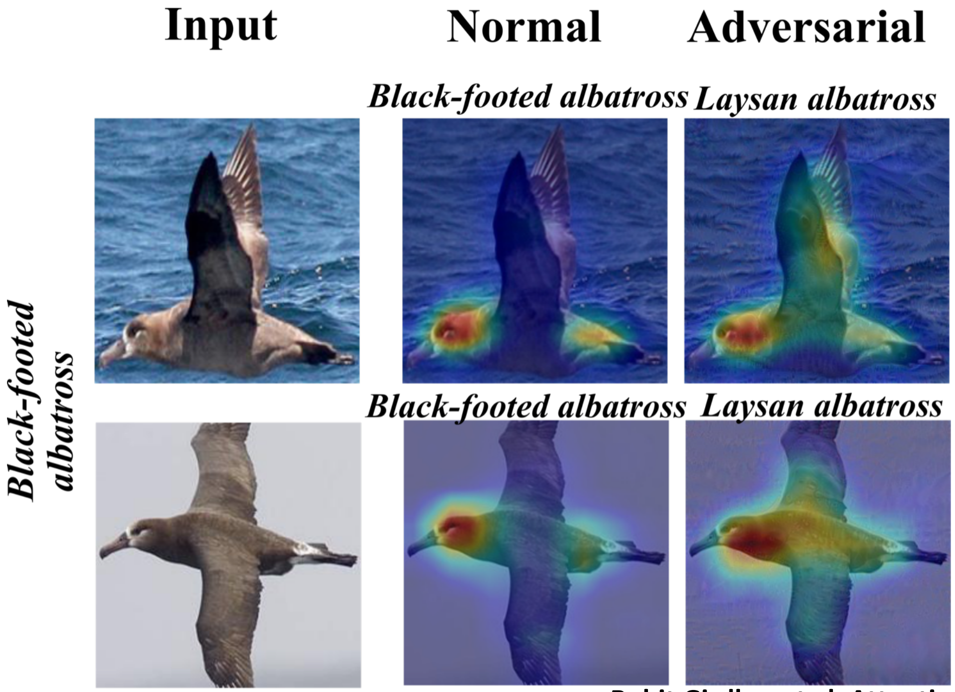
For K- classes,

- K class specific filters
- 1 class agnostic filter
- Both attention maps are multiplied and spatially averaged to yield logits

Rohit Girdhar et al. Attentional Pooling for Action Recognition, NIPS 2017

Key Factors for Success of Adversarial Attacks

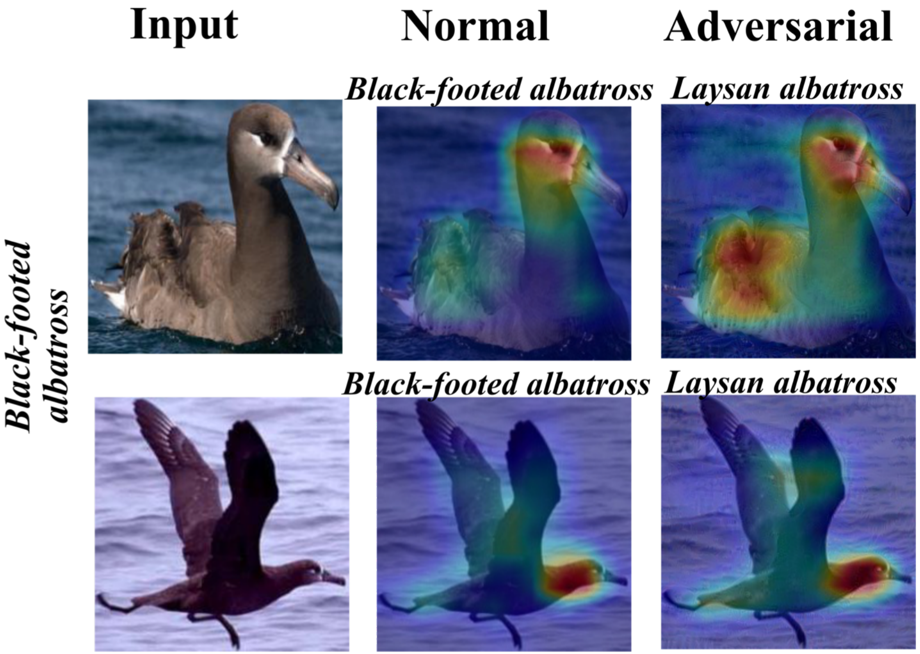
- Discriminative regions of two different classes being too close in feature space



Rohit Girdhar et al. Attentional Pooling for Action Recognition, NIPS 2017

Key Factors for Success of Adversarial Attacks

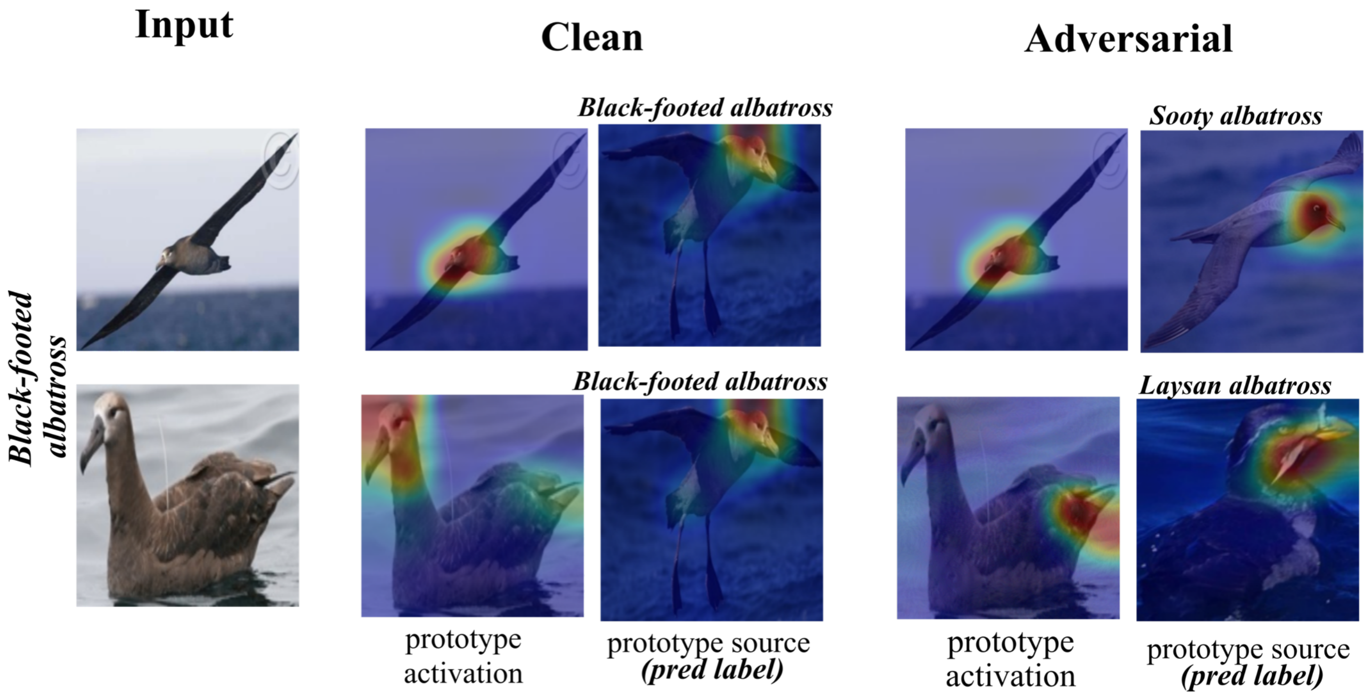
- Discriminative regions of two different classes being too close in feature space



Rohit Girdhar et al. Attentional Pooling for Action Recognition, NIPS 2017

Key Factors for Success of Adversarial Attacks

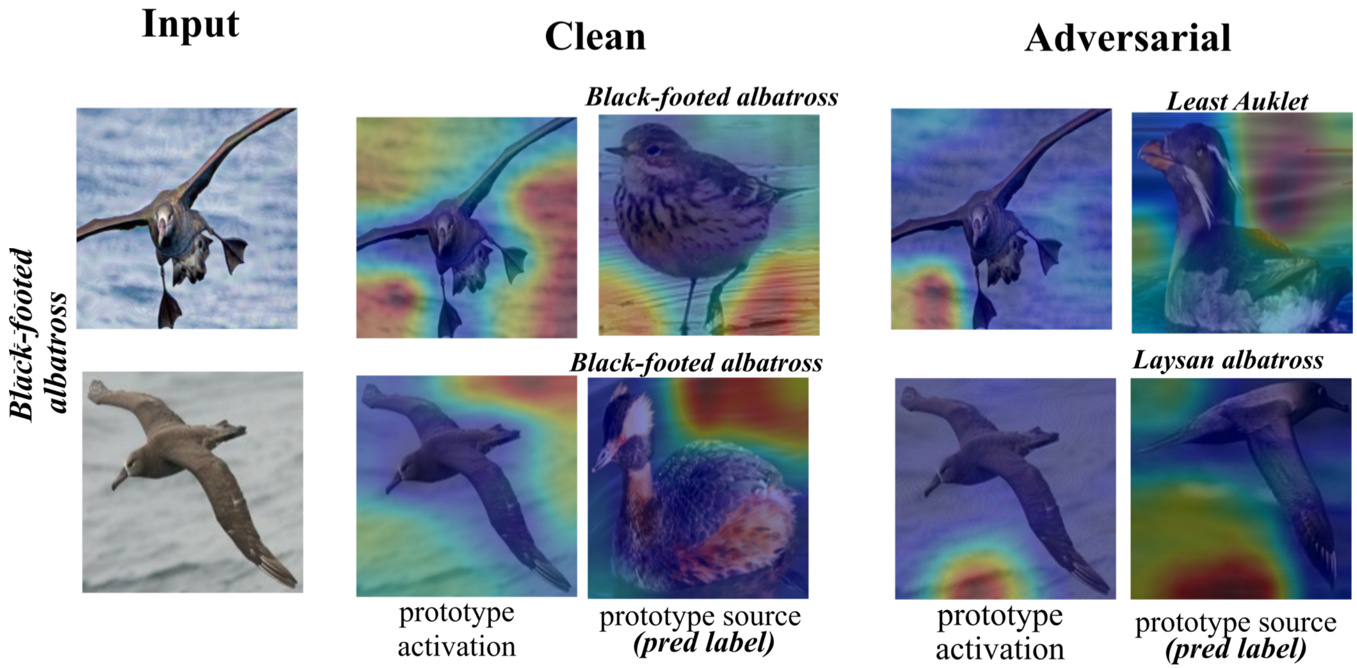
- Discriminative regions of two different classes being too close in feature space



Chaofan Chen et al. This Looks Like That: Deep Learning for Interpretable Image Recognition, NIPS 2019

Key Factors for Success of Adversarial Attacks

- Use of non-discriminative regions for classification



Chaofan Chen et al. This Looks Like That: Deep Learning for Interpretable Image Recognition, NIPS 2019

Framework

We introduce an attention-based regularization mechanism

- Maximally separate the latent features of discriminative regions of different classes
- Minimize the contribution of the non-discriminative regions

Two regularization losses:

Attentional-cluster cost - Prototypes lie close to high attention regions of its own class

Attentional-separation cost - Prototypes lie away from high attention regions of other classes

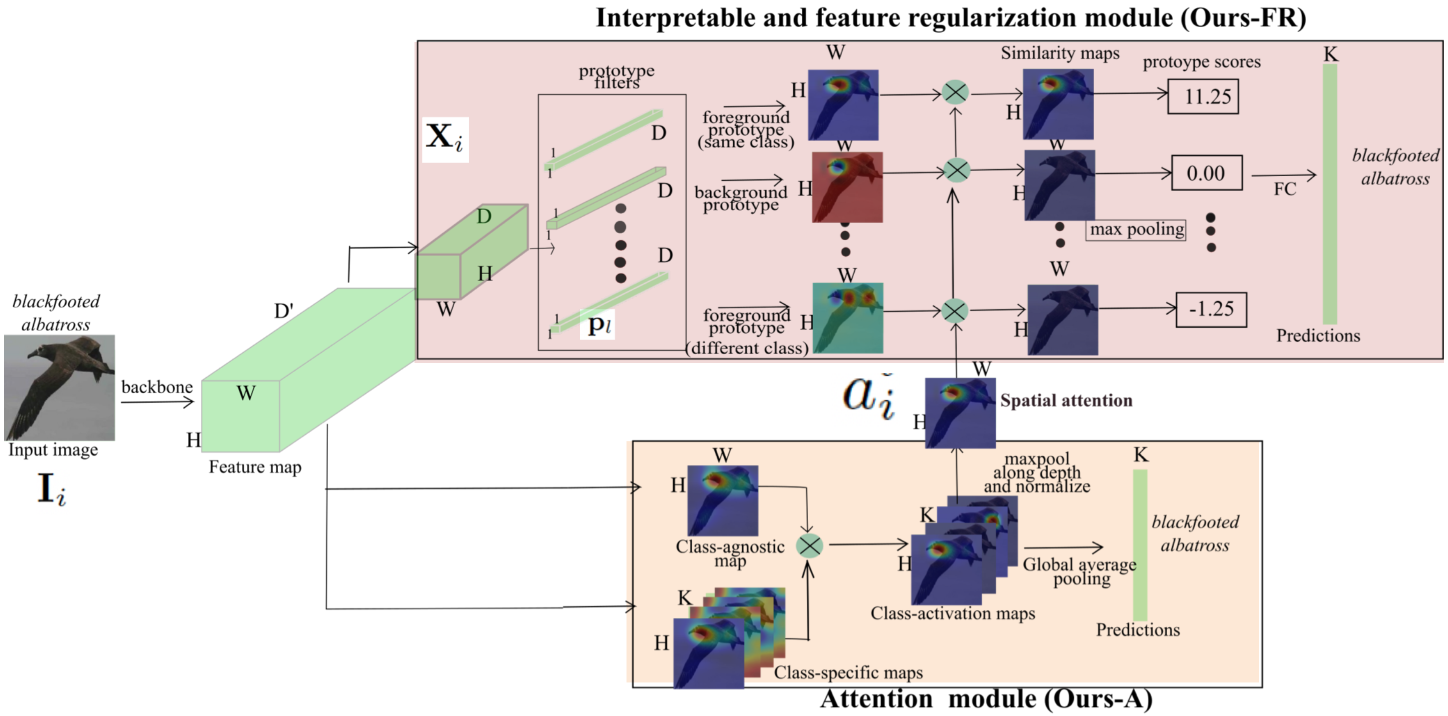
Architecture

Two prediction heads.

1. Spatial attention branch to obtain discriminative regions - Ours-A
2. Feature regularization branch to maximally separate discriminative regions - Ours-FR

Prototype Similarity maps are modulated with spatial attention branch to learn prototypes close to high attention regions.

Proposed Architecture



Discriminative Feature Separation

Attentional Cluster Loss:

The attentional-clustering loss pulls the high-attention regions in a sample close to the nearest prototype of its own class.

$$\mathrm{L}_{clst}^{att}(\mathbf{I}_i) = \sum_{t=1}^{N} a_i^t \min_{l:\mathbf{p}_l \in \mathbf{P}_{y_i}} \|\mathbf{x}_i^t - \mathbf{p}_l\|_2^2$$

a_i^t - Attention weight at location t for image $\mathbf{I}_i$

$\mathbf{x}_i^t$ - feature vector at location t for image $\mathbf{I}_i$

$\mathbf{P}_{y_i}$ - Set of prototypes belonging to class y_i

Discriminative Feature Separation

Attentional Separation Loss:

The attentional separation loss pushes the high-attention regions away from the nearest prototype of any other class.

$$\mathrm{L}_{sep}^{att}(\mathbf{I}_i) = -\sum_{t=1}^{N} a_i^t \min_{l:\mathbf{p}_l \notin \mathbf{P}_{y_i}} \|\mathbf{x}_i^t - \mathbf{p}_l\|_2^2$$

a_i^t - Attention weight at location t for image $\mathbf{I}_i$

$\mathbf{x}_i^t$ - feature vector at location t for image $\mathbf{I}_i$

$\mathbf{P}_{y_i}$ - Prototypes belonging to class y_i

Discriminative Feature Separation

Combined regularization Loss:

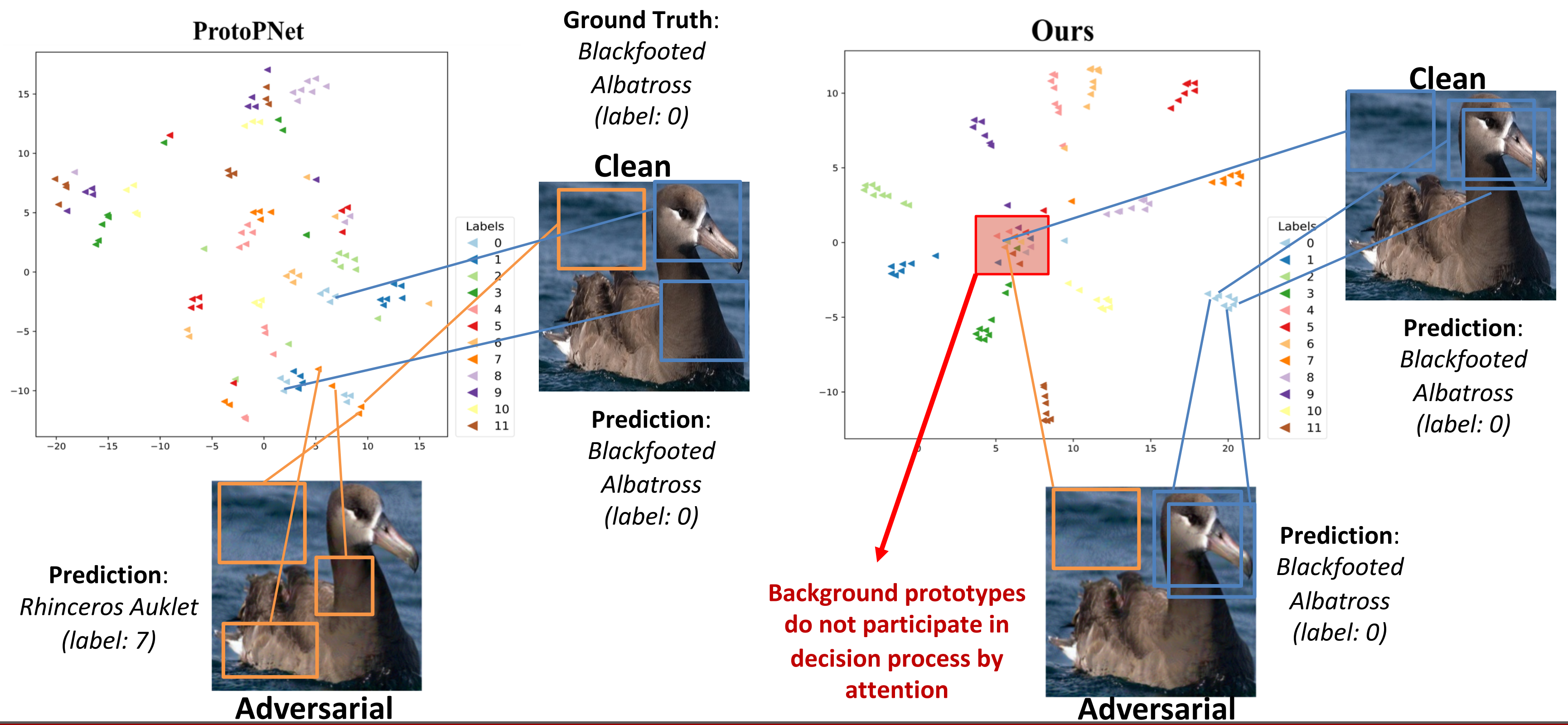
We further push the non-discriminative regions away from informative prototypes by using attention from other images of Batch

$$\mathrm{L}_{reg}(\mathbf{I}_i) = \sum_{j=1}^{B} \sum_{t=1}^{N} \lambda_1 a_j^t \min_{l:\mathbf{p}_l \in \mathbf{P}_{y_i}} \|\mathbf{x}_i^t - \mathbf{p}_l\|_2^2 - \lambda_2 a_j^t \min_{l:\mathbf{p}_l \notin \mathbf{P}_{y_i}} \|\mathbf{x}_i^t - \mathbf{p}_l\|_2^2$$

Total Loss:

$$\mathrm{L}(\mathbf{I}_i) = \mathrm{CE}_{att}(\mathbf{I}_i) + \mathrm{CE}_{reg}(\mathbf{I}_i) + \mathrm{L}_{reg}(\mathbf{I}_i)$$

t-SNE Visualization of Learned Prototypes



Experiments

Datasets – CUB200, Cars196 cropped images

Attacks - FGSM, BIM, PGD and MIM

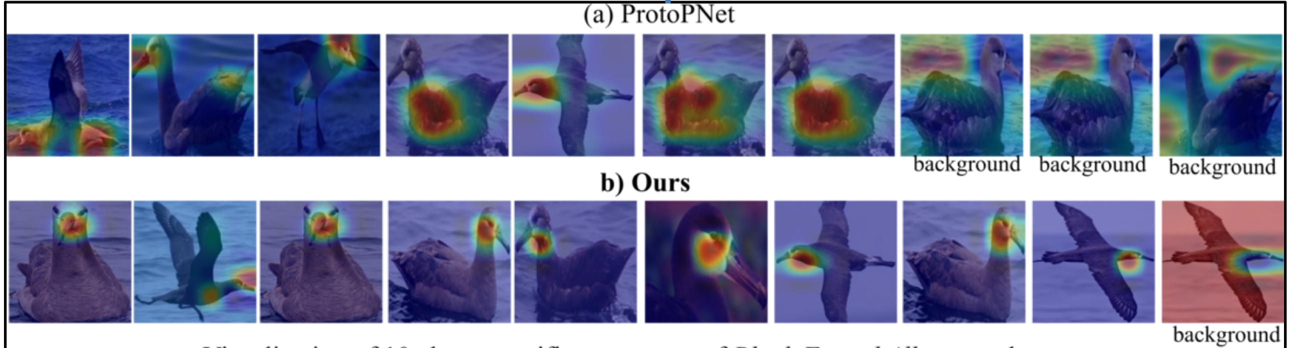
Black Box Transfer attacks - BB-V (VGG-16) and BB-D (DenseNet)

Defense - Adversarial training with single step FGSM with random initialization*

*** Eric Wong et al. Fast is better than free: Revisiting adversarial training, ICLR 2020**

Comparison of the Prototypes

ProtoPNet: multiple background prototypes and prototypes that focus on large regions

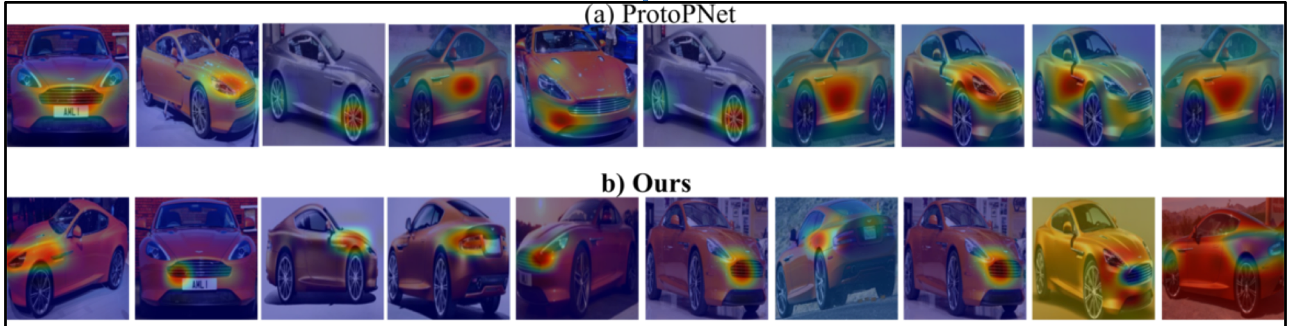


Visualization of 10 classs specific prototypes of *Black Footed Albatross* class

Ours: prototypes are fine-grained and entire non-discriminative regions is activated by single prototype

Comparison of the Prototypes

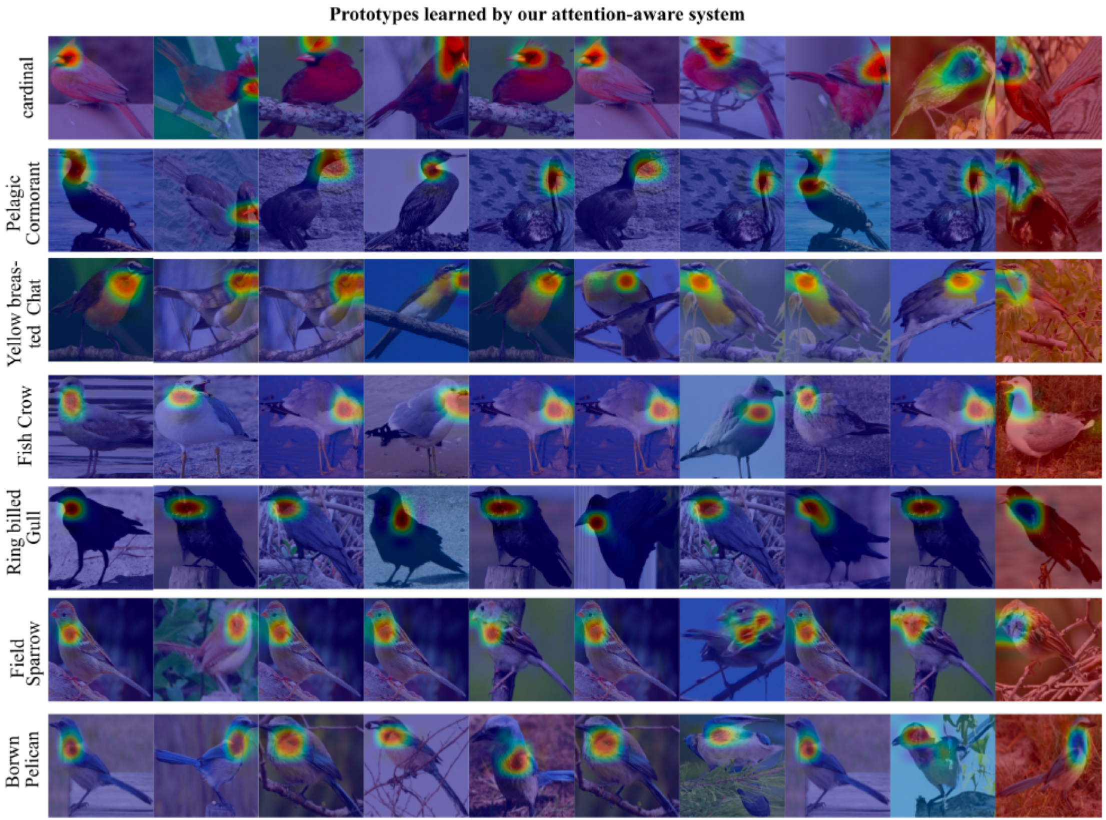
ProtoPNet: multiple background prototypes and prototypes that focus on large regions



Visualization of 10 classs specific prototypes of *Acura TL Sedan 2012* class

Ours: prototypes are fine-grained and entire non-discriminative regions is activated by single prototype

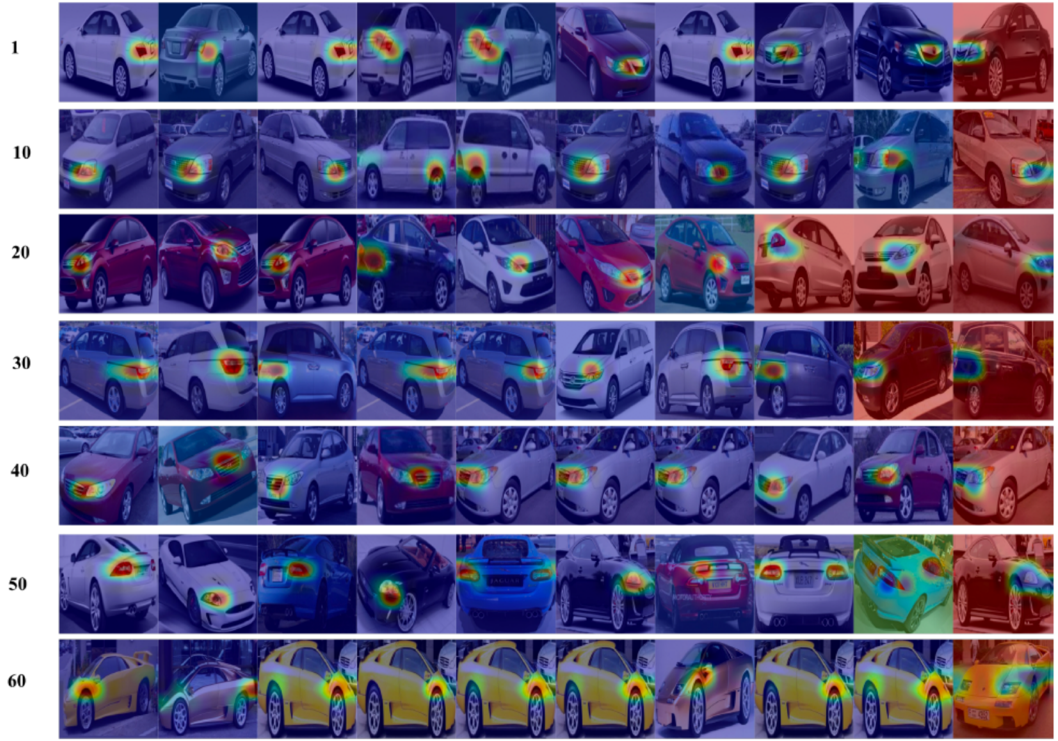
Learned Prototypes



Ours: prototypes are fine-grained and entire non-discriminative regions is activated by single prototype

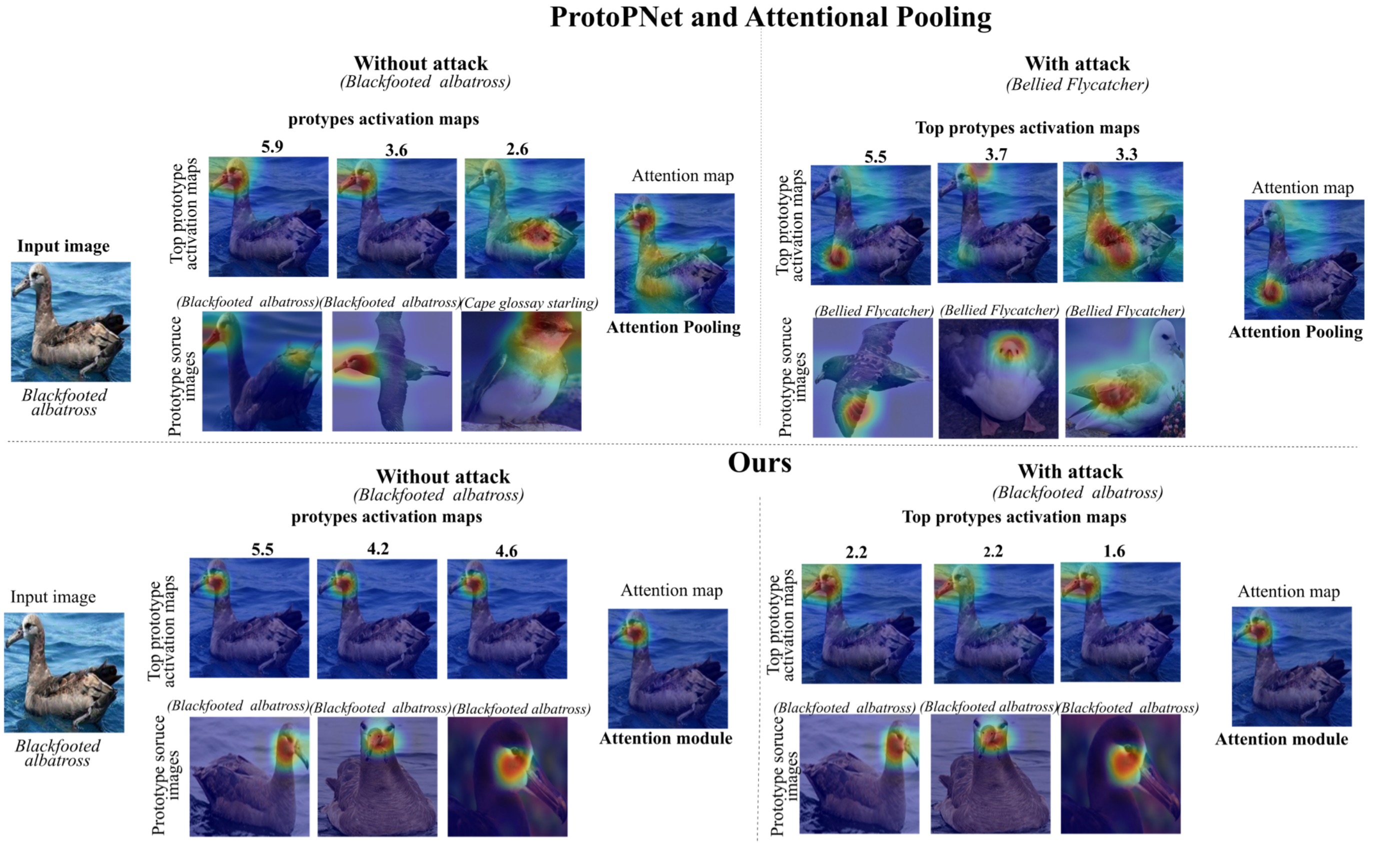
Learned Prototypes

Label | Prototypes learned by our attention-aware system



Ours: prototypes are fine-grained and entire non-discriminative regions is activated by single prototype

Comparison of the activated image regions



Results on CUB200 on Undefended Models

Base Network	Attacks (Steps,ϵ)	Clean (0,0)	FGSM (1,2)	FGSM (1,8)	BIM (10,2)	BIM (10,8)	PGD (10,2)	PGD (10,8)	MIM (10,2)	MIM (10,8)	BB-V (10,2)	BB-D (10,8)
VGG-16	AP [14]	78.0%	36.5%	31.0%	27.7%	14.6%	23.5%	11.7%	30.2%	16.7%	9.6%	60.4%
	AP+ Triplet [57]	**81.0%**	**49.5%**	36.6%	33.5%	11.2%	26.5%	8.50%	37.7%	14.3%	8.54%	63.4%
	AP+ PCL [35]	80.0%	41.0%	33.1%	32.9%	13.6%	23.5%	9.6%	35.3%	17.1%	10.6%	65.8%
	Ours-A	80.4%	47.2%	**40.2%**	**40.0%**	**23.2%**	**35.3%**	**21.8%**	**42.2%**	**26.4%**	**12.9%**	**66.9%**
	ProtoPNet [15]	69.0%	19.9%	8.10%	3.80%	0.00%	2.20%	0.00%	5.00%	0.10%	**22.9%**	58.5%
	ProtoPNet+Ours	**73.2%**	**49.9%**	**42.2%**	**42.5%**	**35.3%**	**38.4%**	**30.1%**	**42.9%**	**37.5%**	15.4%	**59.7%**
VGG-19	AP [14]	75.7%	20.4%	14.5%	13.4%	6.9%	10.5%	5.7%	14.8%	6.9%	21.1%	61.3%
	AP+ Triplet [57]	**82.0%**	**53.9%**	38.2%	35.0%	12.4%	27.7%	9.40%	39.4%	15.3%	17.40%	64.9%
	AP+ PCL [35]	76.9%	20.3%	14.8%	12.1%	5.7%	8.8%	4.2%	13.9%	6.8%	19.8%	60.2%
	Ours-A	79.7%	51.4%	**44.6%**	**42.3%**	**26.5%**	**36.8%**	**26.3%**	**45.0%**	**42.6%**	**29.8%**	**68.2%**
	ProtoPNet [15]	73.8%	22.9%	11.1%	3.2%	0.0%	1.2%	0.0%	3.6%	0.0%	21.0%	58.0%
	Ours-FR	75.4%	**52.2%**	**46.3%**	**46.6%**	**41.3%**	**42.4%**	**31.0%**	**44.4%**	**37.6%**	**30.4%**	**63.7%**
ResNet-34	AP [14]	**79.9%**	30.4%	26.3%	18.0%	7.20%	13.2%	5.8%	22.3%	8.6%	43.0%	59.4%
	AP+ Triplet [57]	78.6%	25.6%	18.7%	11.4%	2.9%	7.1%	1.8%	14.7%	3.8%	42.11%	58.4%
	AP+ PCL [35]	77.9%	30.1%	24.5%	21.4%	13.3%	17.6%	11.6%	23.9%	15.3%	45.7%	61.4%
	Ours-A	79.0%	**32.3%**	**27.0%**	**24.8%**	**20.5%**	**22.5%**	**19.8%**	**26.2%**	**22.0%**	**48.6%**	**63.2%**
	ProtoPNet [15]	75.1%	23.2%	12.8%	7.80%	1.80%	4.10%	1.00%	8.90%	2.20%	39.1%	53.0%
	Ours-FR	**76.3%**	**30.7%**	**22.0%**	**19.3%**	**13.6%**	**14.2%**	**13.0%**	**19.1%**	**13.8%**	**46.0%**	**60.0%**

Table: Classification accuracy of different undefended networks with L_{inf} based attacks on CUB200

Results on Cars196 on Undefended Models

Base Nettwork	Attacks (Steps,ϵ)	Clean (0,0)	FGSM (1,2)	FGSM (1,8)	BIM (10,2)	BIM (10,8)	PGD (10,2)	PGD (10,8)	MIM (10,2)	MIM (10,8)	BB-V (10,2)	BB-D (10,8)
VGG-16	AP [14]	**91.2%**	52.6%	40.2%	37.4%	10.5%	28.8%	6.93%	41.7%	12.9%	12.5%	82.5%
	AP+Triplet [57]	91.1%	54.3%	**43.5%**	42.4%	14.9%	34.1%	9.54%	45.5%	19.2%	15.6%	**84.7%**
	AP+PCL [35]	90.2%	51.7%	40.5%	39.3%	14.1%	31.8%	9.44%	42.5%	17.5%	16.7%	83.9%
	Ours-A	88.5%	**58.7%**	40.2%	**48.0%**	**28.6%**	**46.5%**	**21.7%**	**53.2%**	**33.2%**	**19.9%**	82.2%
	ProtoPNet [15]	**84.5%**	31.2%	9.85%	4.78%	0.01%	2.23%	0.00%	6.5%	0.01%	**27.8%**	**75.5%**
	Ours-FR	83.8%	**60.1%**	**52.0%**	**51.3%**	**41.0%**	**47.8%**	**32.9%**	**51.8%**	**43.9%**	23.4%	75.1%
VGG-19	AP	**91.5%**	50.1%	37.8%	33.4%	10.3%	23.83%	6.93%	37.9%	12.7%	20.7%	82.8%
	AP+Triplet [57]	91.0%	56.2%	45.1%	40.5%	13.0%	30.3%	8.70%	45.3%	16.7%	29.0%	85.0%
	AP+PCL [35]	91.3%	61.3%	49.9%	49.0%	19.7%	40.2%	14.1%	52.4%	23.4%	30.6%	**85.7%**
	Ours-A	88.7%	**64.4%**	**54.8%**	**56.4%**	**36.7%**	**51.7%**	**33.4%**	**58.1%**	**41.0%**	**35.9%**	82.5%
	ProtoPNet [15]	**85.6%**	34.1%	20.8%	11.3%	1.11%	4.40%	0.5%	14.2%	1.39%	26.5%	75.5%
	Ours-FR	85.0%	**62.4%**	**54.7%**	**54.5%**	**45.7%**	**51.2%**	**38.5%**	**54.3%**	**47.6%**	**36.1%**	**76.8%**

Table: Classification accuracy of different undefended networks with L_{inf} based attacks on Cars196.

Results on CUB200 on Robust Models

Base Network	Attacks (Steps,ϵ)	Clean (0,0)	FGSM (1,2)	FGSM (1,8)	BIM (10,2)	BIM (10,8)	PGD (10,2)	PGD (10,8)	MIM (10,2)	MIM (10,8)	BB-V (10,2)	BB-D (10,8)
VGG-16	AP* [14]	54.9%	44.9%	24.2%	41.9%	18.2%	41.2%	16.9%	41.9%	18.7%	54.6%	54.0%
	AP+PCL* [35]	60.7%	50.5%	28.5%	47.1%	22.8%	46.7%	21.6%	47.2%	23.5%	59.5%	59.9%
	Ours-A*	**63.1%**	**56.1%**	**34.8%**	**51.7%**	**29.6%**	**50.8%**	**28.0%**	**52.0%**	**32.5%**	**66.3%**	**68.0%**
	ProtoPNet* [15]	60.1%	44.5%	26.9%	**57.1%**	10.9%	35.9%	10.3%	37.6%	13.5%	58.4%	59.1%
	Ours-FR*	**63.0%**	**53.3%**	**37.3%**	49.4%	**30.4%**	**48.1%**	**28.6%**	**49.7%**	**31.1%**	**61.1%**	**62.0%**
VGG-19	AP* [14]	58.0%	47.5%	29.1%	44.3%	25.6%	44.0%	24.34%	44.4%	26.2%	57.0%	57.3%
	AP+PCL* [35]	61.8%	52.1%	30.9%	48.9%	24.7%	48.6%	23.3%	49.1%	25.4%	60.5%	60.9%
	Ours-A*	**68.2%**	**57.1%**	**36.5%**	**53.2%**	**30.4%**	**52.6%**	**29.2%**	**53.5%**	**31.2%**	**66.2%**	**66.9%**
	ProtoPNet* [15]	55.1%	40.0%	28.9%	26.5%	11.3%	29.7%	9.60%	25.6%	10.2%	53.6%	53.9%
	Ours-FR*	**64.4%**	**55.5%**	**37.4%**	**51.2%**	**30.6%**	**50.4%**	**28.7%**	**52.1%**	**32.3%**	**62.5%**	**63.2%**
ResNet-34	AP* [14]	55.6%	47.8%	29.2%	44.80%	21.0%	44.5%	19.4%	44.9%	21.9%	55.3%	55.2%
	AP+PCL* [35]	54.5%	45.4%	26.9%	42.3%	18.2%	41.9%	16.4%	42.4%	19.1%	54.0%	54.0%
	Ours-A*	**62.2%**	**54.2%**	**35.7%**	**51.5%**	**25.5%**	**51.0%**	**23.1%**	**51.6%**	**26.6%**	**61.5%**	**61.9%**
	ProtoPNet* [15]	**57.9%**	46.5%	30.3%	41.1%	21.1%	40.3%	18.4%	41.5%	20.9%	56.9%	57.0%
	Ours-FR*	57.6%	**49.5%**	**32.3%**	**45.8%**	**23.2%**	**44.9%**	**19.9%**	**46.1%**	**24.6%**	**57.1%**	57.0%

Table: Classification accuracy of different robust networks with L_{inf} based attacks on CUB200.

Results on Cars196 on Robust Models

Base Network	Attacks Steps,ϵ)	Clean	FGSM (1,2)	FGSM (1,8)	BIM (10,2)	BIM (10,8)	PGD (10,2)	PGD (10,8)	MIM (10,2)	MIM (10,8)	BB-V (10,8)	BB-D (10,8)
VGG-16	AP* [6]	86.2%	**81.1%**	**63.6%**	**78.9%**	53.8%	**78.7%**	50.8%	**78.7%**	55.1%	85.1%	85.9%
	AP+PCL* [7]	**87.4%**	80.5%	59.4%	77.6%	48.5%	77.2%	44.9%	77.9%	50.2%	**86.0%**	**87.1%**
	Ours-A*	84.8%	79.8%	63.3%	77.0%	**54.6%**	76.6%	**51.1%**	77.1%	**55.8%**	84.5%	85.6%
	ProtoPNet* [3]	64.4%	53.7%	31.9%	48.9%	16.5%	48.2%	13.4%	49.2%	18.2%	63.8%	64.2%
	Ours-FR*	**83.7%**	**76.37%**	**62.8%**	**73.5%**	**55.0%**	**72.6%**	**51.9%**	73.8%	**55.4%**	**80.8%**	**82.0%**
VGG-19	AP* [6]	**88.2%**	82.4%	63.4%	79.9%	54.2%	79..6%	50.7%	80.0%	55.7%	86.9%	88.0%
	AP+PCL* [7]	**88.2%**	**82.7%**	64.6%	**80.2%**	57.4%	**79.6%**	54.3%	**80.3%**	58.5%	**87.2%**	**88.1%**
	Ours-A*	87.3%	80.29%	**67.1%**	78.4%	**60.15%**	78.2%	**58.2%**	78.6%	**61.3%**	86.5%	87.3%
	ProtoPNet* [3]	30.0%	19.9%	15.7%	15.0%	16.3%	9.1%	3.00%	3.32%	2.28%	29.4%	29.7%
	Ours-FR*	**84.6%**	**79.6%**	**66.9%**	**77.7%**	**58.6%**	**76.5%**	**55.6%**	**77.8%**	**59.1%**	**83.7%**	**84.5%**

Table: Classification accuracy of different robust networks with L_{inf} based attacks on Cars196.

Ablation Study

Network	Att-clustering loss	Att-separation loss	Clean (0,0)	PGD (10,8)
AP [14]	-	-	78.0%	11.7%
Ours-A	-	-	78.7%	14.07%
	-	✓	79.6%	0.0%
	✓	-	80.0%	19.3%
	✓	✓	80.4%	21.8%

Network	Att-clustering loss	Att-separation loss	Clean (0,0)	PGD (10,8)
ProtoPNet [15]	-	-	69.0%	0.0%
Ours-FR	-	-	75.7%	13.76%
	-	✓	69.8%	0.0%
	✓	-	73.7%	18.7%
	✓	✓	73.2%	30.1%

Table: Contribution of each proposed feature regularization module in classification accuracy of undefended VGG-16 network

Gradient Obfuscation Study

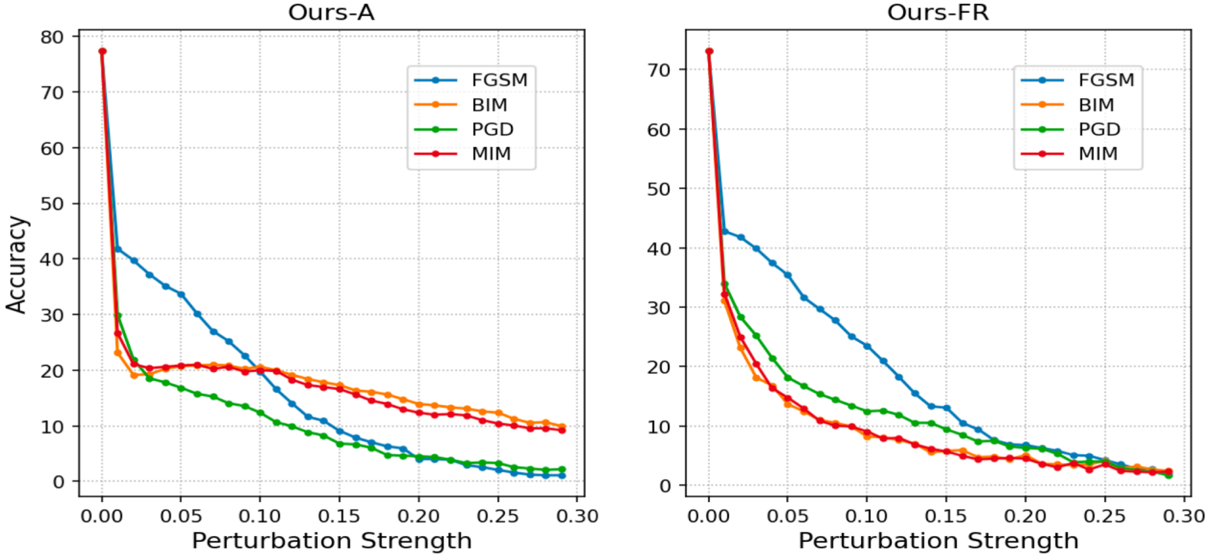


Table: Performance of VGG-16 with our proposed approach under different perturbation strengths.

Adversarial Detection

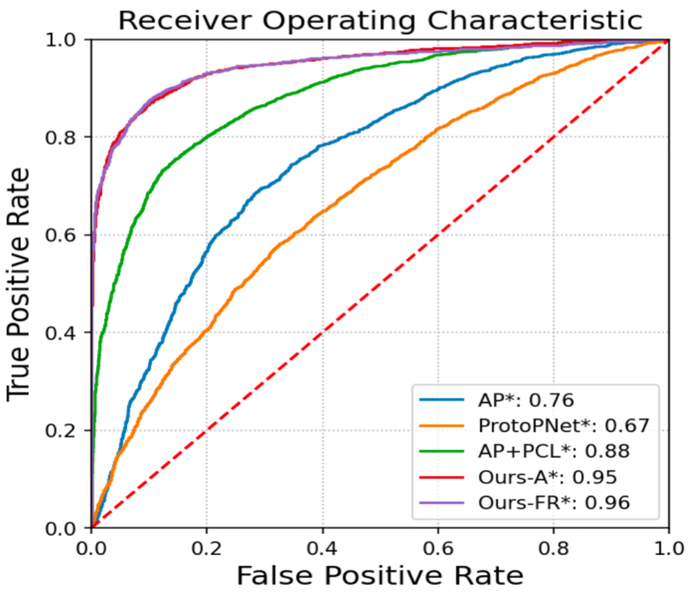


Table: ROC curves for adversarial sample detection on robust VGG-16 with PGD attack

Ours-A* and Ours-FR* performs better than baselines

- Compute minimum Mahalanobis distance from pretrained class conditional distributions at each layer
- Train a logistic detector on 20% samples and evaluated on rest 80% of adversarial successful cum correctly classified test data

Conclusion

- We have performed the first study of adversarial attacks for fine-grained recognition.
- Our analysis has highlighted the key factor for the success of adversarial attacks in this context.
- Designed an attention and prototype-based framework that explicitly encourages the prototypes to focus on the discriminative image regions

Thank you!