MetaIRNet++:

Augmented One Shot Fine Grained Image Recognition Algorithm Based On Differentiable Data Augmentation

Talker: Kai Chen
Supervisor: Prof. Yanwei Fu
School of Computer Science, Fudan University
2020.06.16

CONTENT

01

Key Words

Three Key Words

- Fine grained Image Recognition
- One Shot Learning
- MetaIRNet++
  - Augmented MetaIRNet [1]

[1] Satoshi, Fu, Crandall. Meta-Reinforced Synthetic Data for One-Shot Fine-Grained Visual Recognition. In NIPS, 2019

Quiz: Hawk or Falcon?

source: https://birdeden.com/distinguishing-between-hawks-falcons

Hawk

Falcon

source: https://birdeden.com/distinguishing-between-hawks-falcons

Fine-grained Image Recognition

- Much harder than normal classification.
  - Higher **intra-class** variance.
  - Lower **inter-class** variance.
- Difficult to collect data.
  - Can’t use crowdsourcing.
  - Need expert annotator.
- Need **one-shot** learning.

One Shot Learning

- Solution: **Meta Learning**
  - Deep Learning = Representation Learning
  - Meta Learning = Representation + Transfer Learning
- Goal: learn prior knowledge and generalize quickly to novel tasks
  - Embedding transfer learning into training process **explicitly**.
  - Always training on support set and evaluate on query set
- In this work we focus on finding a better prior model and releasing the difficulty of generalization, which can be thought as an **optimization** problem.

Three Key Words

- Fine grained Image Recognition
- One Shot Learning
- MetaIRNet++
  - Augmented MetaIRNet [1]

[1] Satoshi, Fu, Crandall. Meta-Reinforced Synthetic Data for One-Shot Fine-Grained Visual Recognition. In NIPS, 2019

02

MetalRNet

Can we generate more data?

- How about state-of-the-art GANs?

- Challenge: GAN training itself need a lot of data.

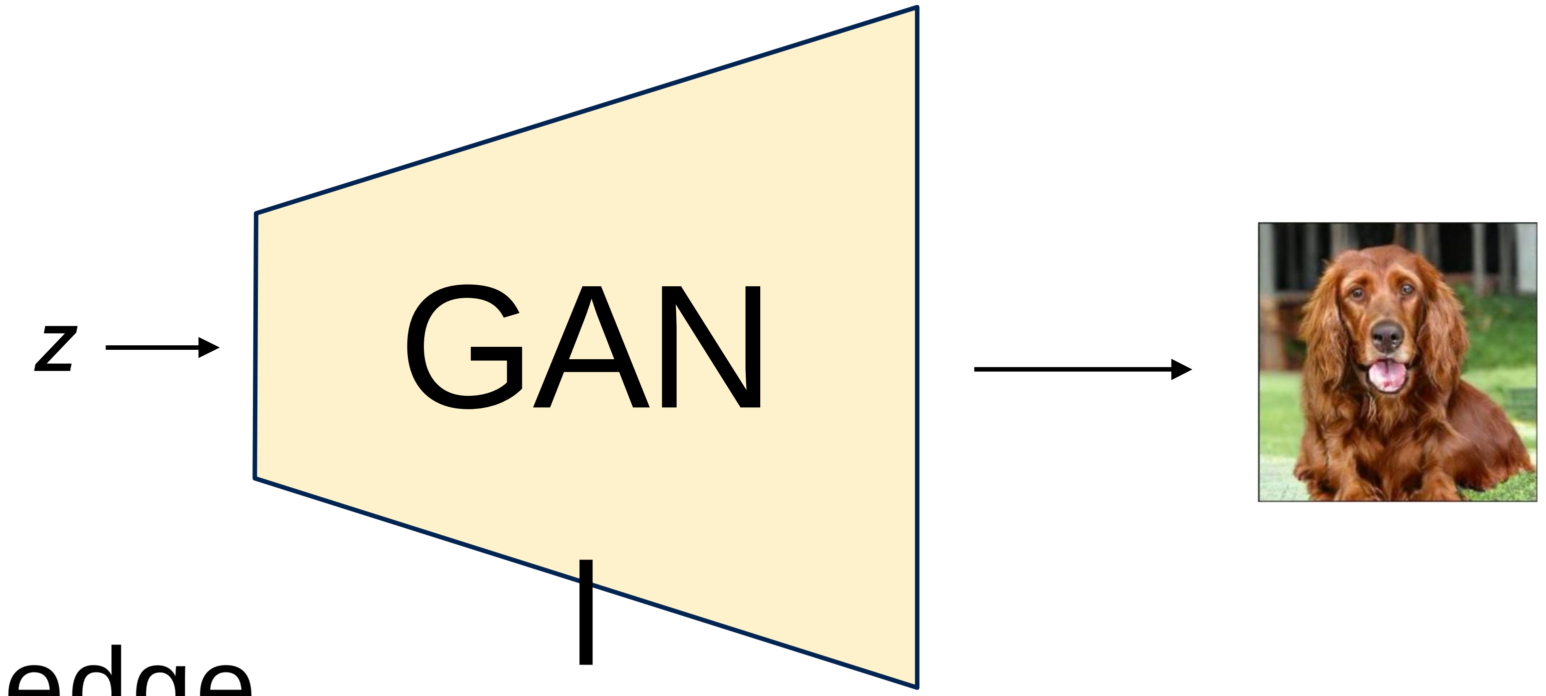
Fine-tune GANs trained on ImageNet.

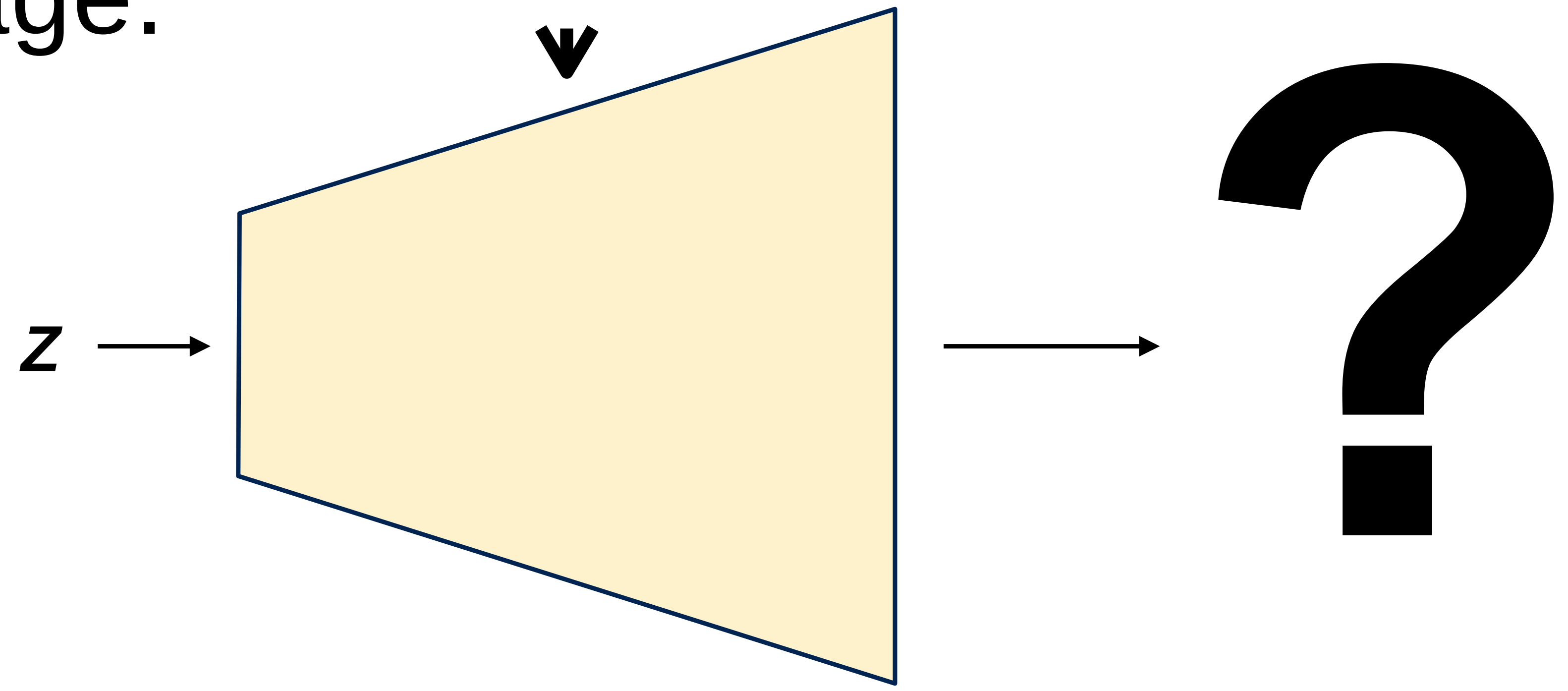
One Million General Images

IMAGENET

A Domain Specific Image

Transfer generative knowledge
from one million general images
to a domain specific image.





Fine-tune BigGAN with a single image

$$L_G(G, \mathbf{I}_z, z) = L_1(G(z), \mathbf{I}_z) + \lambda_p * L_{perc}(G(z), \mathbf{I}_z) + \lambda_z * L_{EM}(z, r)$$

- Terminology:
  - G: GAN generator;
  - $\mathbf{I}_z$: Real Image;
  - z: random noise
  - $r \sim N(0, 1)$
- Loss Function:
  - L_1 : Per-pixel L1 Loss;
  - L_{prec}: Perceptual Loss;
  - L_{EM} : Earth Moving Distance

Fine-tune Batch Norm Only

$$\hat{x} = \frac{x - \mathbb{E}(x)}{\sqrt{\mathrm{Var}(x) + \epsilon}} \qquad h = \gamma\hat{x} + \beta$$

Original

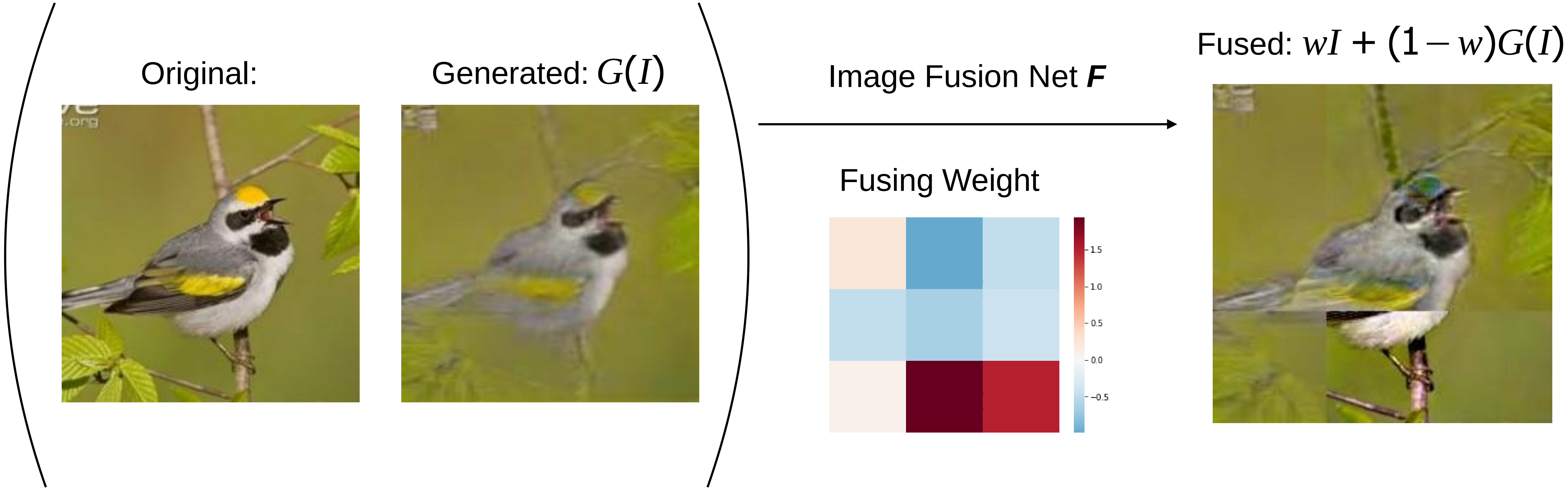
Fine-Tune All

Fine-Tune BatchNorm

Are generated image helpful?

- No! Accuracy drops :(
- GAN images usually do not help classification.
- Problem: Mode Collapse
- Challenge: How to utilize the generated images?

Learn to reinforce with original



Use meta-learning to learn the best mixing strategy to help one-shot classifiers.

03

MetalIRNet++

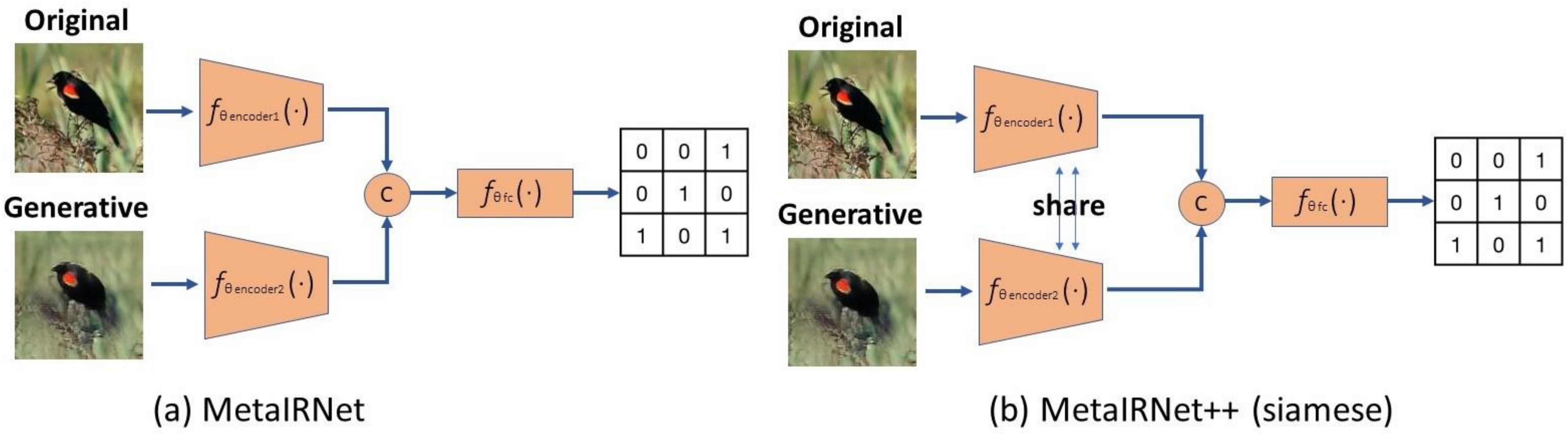
Analysis: Advantage

- **Intuition**: Images following real data distribution may not need to seem real
- Image Fusion Network:
  - For generated images, complement information loss
  - For real images, increase data complexity and information entropy
  - Make the indifferentiable fusion operator differentiable

Analysis: Problem

- **Problem 1**: Too many parameters
  - Under default setting, 2/3 parameters are used for image fusion network while only 1/3 parameters work for classification
- **Problem 2**: Robustness of global-inconsistent synthetic images
  - Synthetic images: local consistent and global inconsistent
  - Real & generated: local and global consistent
  - Optimization on two data domains: sub-optimal

MetaIRNet++ (Siamese)



- Use Siamese Network [2] in Image Fusion Network
  - Guarantee the same embedding space
  - Saving at least 1/3 parameters while keeping comparable accuracy
- Half learning rate for Image Fusion Network

[2] Spoel et al. Siamese Neural Networks for One-Shot Image Recognition. In ICML Workshop, 2015.

Implicit Global Inconsistent: Pilot Study

表 3.2: 5-way-1-shot conv4 from scratch accuracy (%) on CUB val set using different implicit augmented methods. **1 stage** means we don't use pretrained ProtoNet.

Method	Backbone	ProtoNet Acc	MetaIRNet Acc
1 stage	Conv4	-	60.31±2.46
ProtoNet	Conv4	61.49±2.17	62.80±2.08
Global softmax	Conv4	61.49±2.17	60.35±1.98
Mixup	Conv4	64.10±2.41	**63.04±2.14**
Random Fusion (Beta)	Conv4	63.58±2.46	62.15±2.10
Random Fusion (Uniform)	Conv4	61.43±2.06	61.13±2.12

- **Conclusion:**
  - Implicit methods can't bring significant improvement
  - Especially after long training process.

Explicit Global Inconsistent: MetaIRNet++ (Block Dropout)

- Based on the 3 x 3 grid structure in image fusion network, during every forward propagation, we random drop some of the image patches to introduce inconsistence to all data points.

Original

Fine-Tune

Fusion

Block Dropout

MetalRNet++ (Block Dropout)

- **Block Dropout**: info dropping data augmentation [3]
  - We don't want to optimize on two separate data domains.
  - We can't make generated images closer to original ones
  - So can we introduce inconsistence to original images in reverse?
- **Dropout Probability**: 0.5 by default
  - Same with modern data augmentation in vision
  - Introduce no prior in this random process

[3] Chen et al. GridMask Data Augmentation, Arxiv.

MetalRNet++ (Block Dropout)

- **Keep ratio:** 0.5 – 1.0 by default
- **Centerness**: always keep center patch
  - Balance between foreground information and background information [3]
- **Variance**: fix data variance in training time

$$r_{keep} = \frac{sum(mask == 1)}{sum(mask)}$$

$$\mathbf{I}_{augment} = \frac{1}{r_{keep}} * \mathbf{I} * mask$$

[3] Chen et al. GridMask Data Augmentation, Arxiv.

MetaIRNet++ (Block Dropout)

Algorithm 1 Block Dropout procedure $f_{block\_dropout}$ of our MetaIRNet++

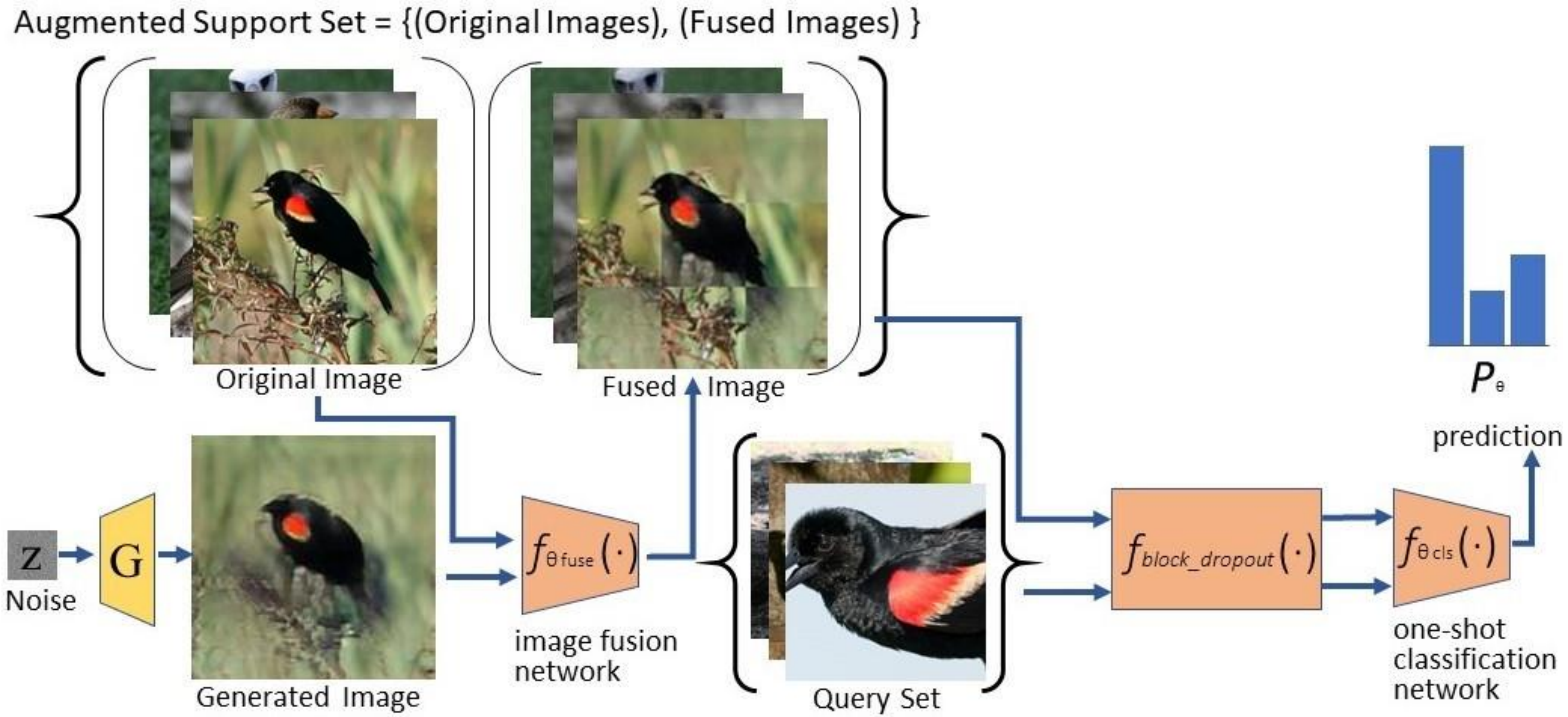
I $\leftarrow$ input image batch
prob $\leftarrow$ dropout prob
ratio $\leftarrow$ keep ratio range, a list of length 2
block_size $\leftarrow$ block augmentation weight size (3 by default)

if random.rand() $>$ prob **then**
  return I
end if

$r_{keep} \leftarrow$ random.uniform(ratio[0], ratio[1])
$mask \leftarrow$ mask_generation(r_{keep})
$mask$[center] $\leftarrow 1$
$mask \leftarrow mask$.upsample(I.size())

because unsampling might change r_{keep} slightly
$r_{keep} \leftarrow$ sum($mask == 1$) / sum($mask$)
$mask \leftarrow 1/r_{keep} * mask$
return I * $mask$

MetaIRNet++ Framework



04

Experiments

Implementation Details

- Dataset: Caltech-UCSD Bird Dataset [4]
- Two experiments setting:
  - Pre-trained: ImageNet pretrained ResNet-18
  - Scratch: Conv4 + two stage training
- Keep all hyperparameters and learning schedule same with MetaIRNet

[4] Wan et al. The caltech-ucsd birds-200-2011 dataset, California Institute of Technology, 2011.

Ablation Study: MetaIRNet++ (Siamese)

表 4.1: MetaIRNet++(siamese) with pretrained ResNet-18's 5-way-1-shot accuracy (%) on CUB val set.

Model	LR	Params(M)	MACs(G)	Acc
MetaIRNet	0.001	33.54	181.86	**84.01±1.70**
MetaIRNet++(siamese)	0.001	22.36	181.86	83.79±1.83
MetaIRNet++(siamese)	0.0005	22.36	181.86	87.98±1.54
MetaIRNet++(siamese)	0.0005 fusion	22.36	181.86	**83.40±1.73**
Δ		**33.33%**	0	**0.61**

表 4.2: MetaIRNet++(siamese) with conv4 from scratch's 5-way-1-shot accuracy (%) on CUB test set.

Model	LR	Params(M)	MACs(G)	Acc
MetaIRNet	0.001	22.48	81.76	**62.80±2.08**
MetaIRNet++(siamese)	0.001	11.30	81.76	61.78±2.05
MetaIRNet++(siamese)	0.0005	11.30	81.76	60.93±2.10
MetaIRNet++(siamese)	0.0005 fusion	11.30	81.76	**62.51±2.28**
Δ		**49.73%**	0	**0.29**

Ablation Study: MetaIRNet++ (Siamese)

- **Params, MACs and Accuracy**:
  - Same MACs but saving at least 33.3% parameters
  - Still Comparable accuracy
  - Make end-to-end training “Generator + MetaIRNet++” possible
- **Half learning rate for image fusion network**:
  - Narrow the gap between MetaIRNet and MetaIRNet++ (Siamese)

Ablation Study: MetaIRNet++ (Siamese)

- **Siamese or not**:
  - MetaIRNet++(Siamese) always worse than MetaIRNet
  - Reason:
    - Real images and generated images: two domains
    - Two branches in fusion network is unbalanced: our goal is to reinforce real images
- If data is enough, separate branches achieves best
- Or using Siamese can still get comparable accuracy

Ablation Study: MetaIRNet++ (Block Dropout)

表 4.3: MetaIRNet++(Block Dropout) with pretrained ResNet-18's 5-way-1-shot accuracy (%) on CUB val set. Variables are Dropout Prob and Centerness.

Model	Prob	Centerness	Acc
MetaIRNet	-	-	84.01±1.70
MetaIRNet++(BD)	0.5	0	82.66±1.90
MetaIRNet++(BD)	0.8	0	82.38±1.77
MetaIRNet++(BD)	1.0	0	77.35±2.18
MetaIRNet++(BD)	0.5	1	**85.06±1.74**
MetaIRNet++(BD)	0.8	1	82.38±2.04
MetaIRNet++(BD)	1.0	1	84.14±1.61

表 4.4: MetaIRNet++(Block Dropout) with pretrained ResNet-18's 5-way-1-shot accuracy (%) on CUB val set. Variables are Keep Ratio.

Model	Keep Ratio	Acc
MetaIRNet	-	84.01±1.70
MetaIRNet++(BD)	0.8	84.69±1.77
MetaIRNet++(BD)	(0.3, 0.5)	83.48±1.82
MetaIRNet++(BD)	(0.3, 0.8)	82.85±2.08
MetaIRNet++(BD)	(0.3, 1.0)	**85.15±1.79**
MetaIRNet++(BD)	(0.5, 0.8)	84.74±1.68
MetaIRNet++(BD)	(0.5, 1.0)	**84.93±1.78**
MetaIRNet++(BD)	(0.8, 1.0)	**85.10±1.80**

表 4.5: MetaIRNet++(Block Dropout) with pretrained ResNet-18's 5-way-1-shot accuracy (%) on CUB val set. Variables are Variance Fix Method.

Model	Variance	Acc
MetaIRNet	-	84.01±1.70
MetaIRNet++(BD)	Sample Keep Ratio	85.03±1.71
MetaIRNet++(BD)	Mean	**85.49±1.64**

表 4.6: MetaIRNet++(Block Dropout) with pretrained ResNet-18's 5-way-1-shot accuracy (%) keeps raising.

Model	Change	Acc	Δ
MetaIRNet		84.01±1.70	
	+ Prob 0.5 and Centerness	85.06±1.74	+ 1.05
	+ Keep ratio (0.5, 1.0)	84.93±1.78	+ 0.92
	+ / Mean	85.49±1.64	+ 1.48

MetaIRNet++ on CUB Test Set: Pretrained

表 4.7: MetaIRNet++ with pretrained ResNet-18's 5-way-1-shot accuracy (%) on CUB test set. Results marked * come from the original MetaIRNet[1] paper.

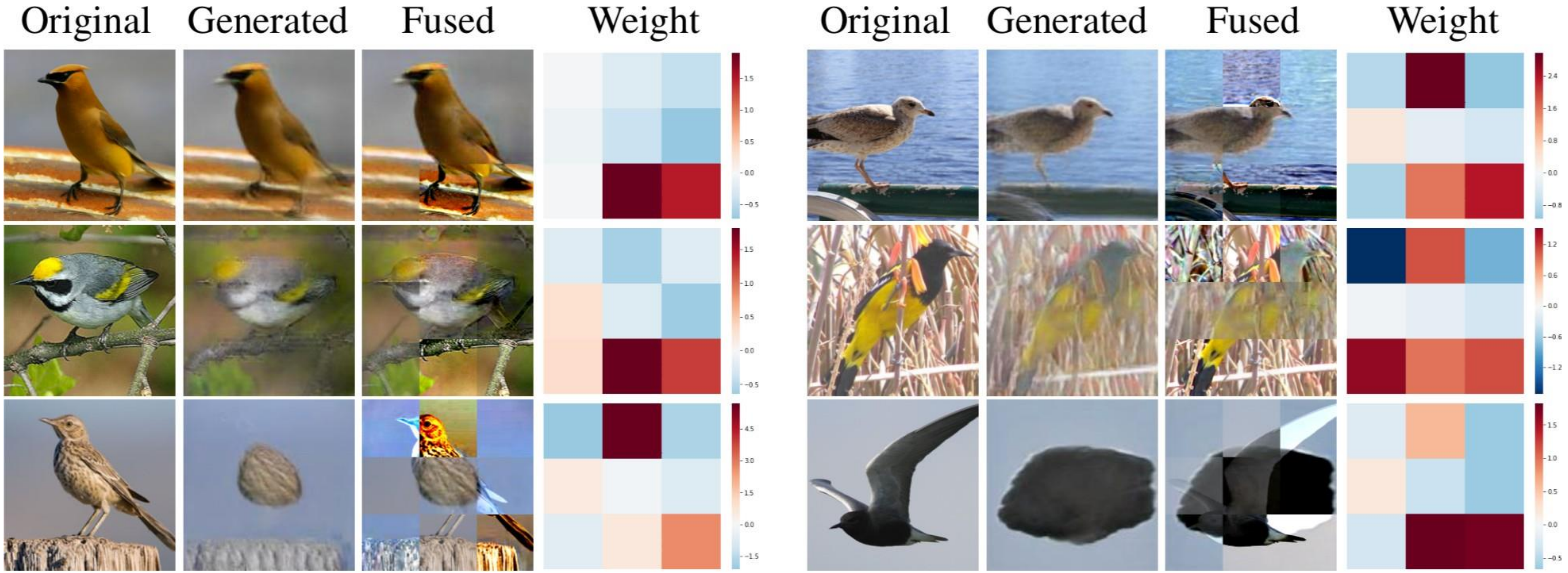
Model	Data Augmentation	Acc
Nearest Neighbor	-	79.00±0.62
Logistic Regression	-	81.17±0.60
*Softmax Regression	-	80.77±0.60
Softmax Regression	-	80.59±0.61
ProtoNet[15]	-	81.73±0.63
ProtoNet	FinetuneGAN	78.82±0.70
*ProtoNet	FinetuneGAN	79.40±0.69
ProtoNet	Flip	82.66±0.61
ProtoNet	Gaussian	81.75±0.63
ProtoNet	FinetuneGAN, Mixup	82.24±0.60
MetaIRNet[1]	FinetuneGAN	84.13±0.58
MetaIRNet	FinetuneGAN, Flip	84.80±0.56
MetaIRNet++	FinetuneGAN	**84.81±0.57**
MetaIRNet++(siamese)	FinetuneGAN	84.01±0.60
MetaIRNet++(BD)	FinetuneGAN	85.13±0.57
MetaIRNet++(BD)	FinetuneGAN, Flip	**85.53±0.56**
MetaIRNet++(BD)	FinetuneGAN, Dropout	84.58±0.57

MetaIRNet++ on CUB Test Set: Scratch

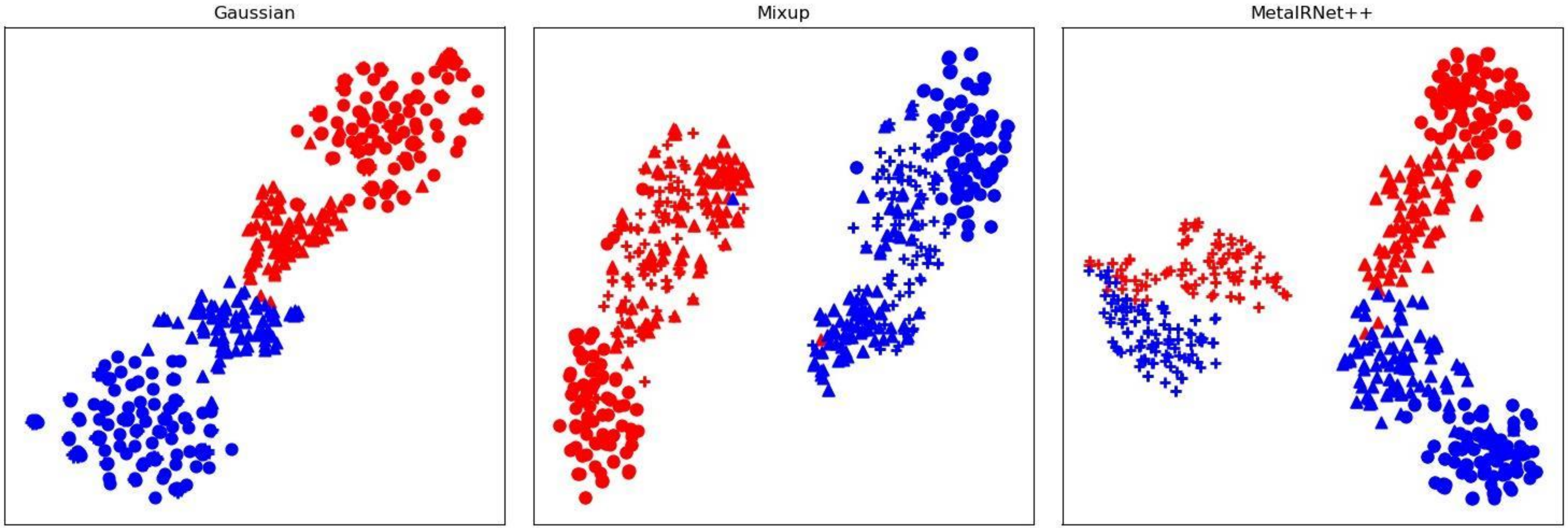
表 4.8: MetaIRNet++ with conv4 from scratch's 5-way-1-shot accuracy (%) on CUB test set. Results marked * come from the original MetaIRNet[1] paper.

Model	Backbone	Acc
MAML[14]	Conv4	55.92±0.95
MatchingNet[49]	Conv4	61.16±0.89
RelationNet[50]	Conv4	62.45±0.98
ProtoNet[15]	Conv4	63.50±0.70
*MetaIRNet[1]	Conv4	65.86±0.72
MetaIRNet	Conv4	66.66±0.70
MetaIRNet++	Conv4	65.76±0.73
MetaIRNet++(BD)	Conv4	67.59±0.73
MetaIRNet++(BD)	ResNet-18	**68.22±0.74**

Examples

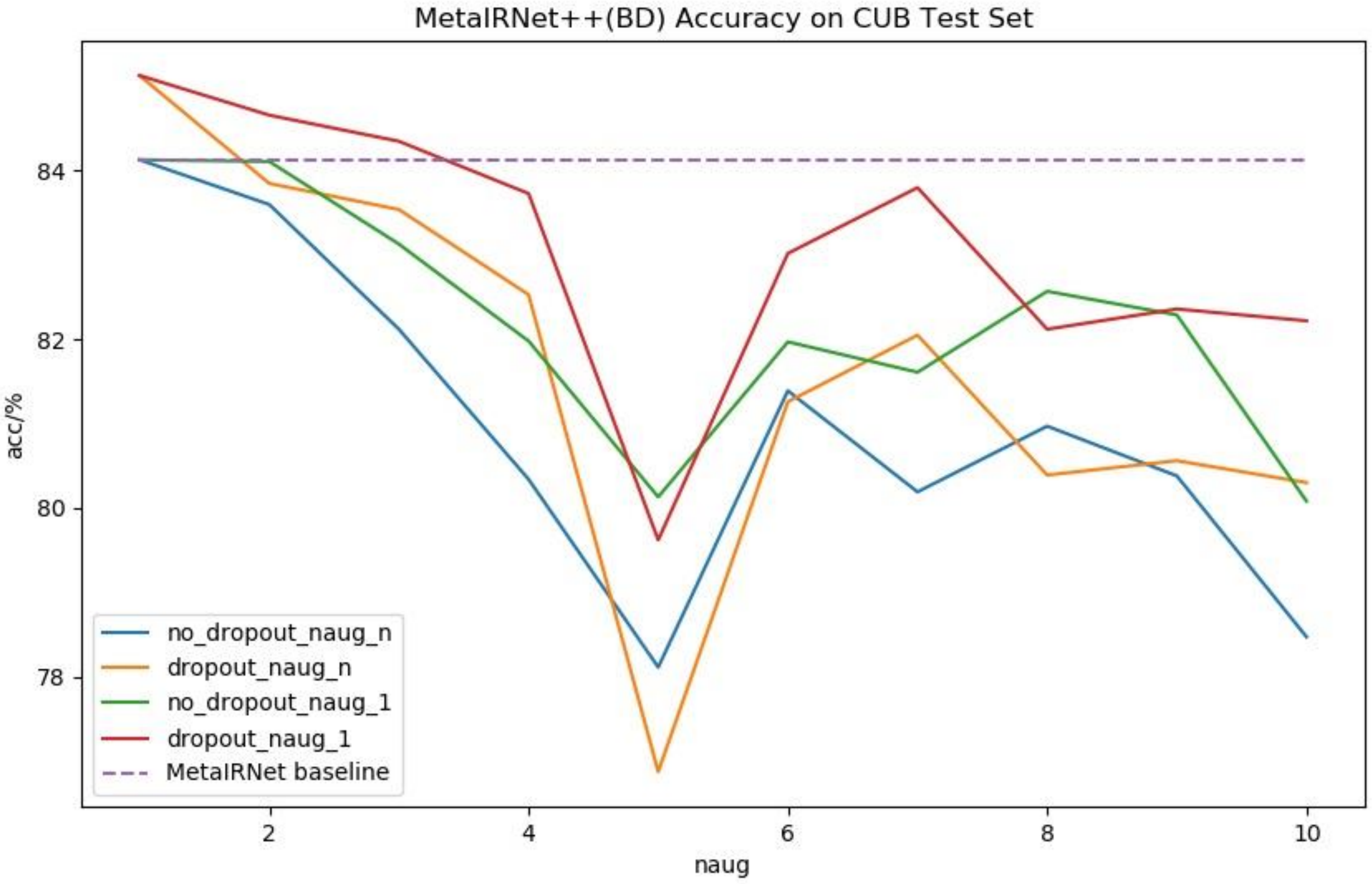

Original
Generated
Fused
Weight
Original
Generated
Fused
Weight

t-SNE of two random classes



- Visualization of three different types of data augmentation using samples from two random classes.
- MetaIRNet++ extends the decision boundary.

Robustness of n_{aug}



Future Work

- To end-to-end training “Generator + MetaIRNet++”, which is a safer way to fine tune a GAN-like generator
- To decrease the Block Dropout’s sensitivity of hyperparameters
- The minimum of n_{aug} is still $n_{aug} = 1$. We need to find a way to make full use of generated images.

Summary

- An augmented method for one-shot fine-grained recognition.
- We propose
    1. **Siamese Network** to save at least 1/3 parameters at the same time of keeping comparable accuracy.
    2. **Block Dropout** to increase one-shot fine-grained recognition accuracy with constant computational complexity.
- Empirically demonstrates the effectiveness.
- Encourage more work on image patches for one shot learning

Reference

[1] Satoshi, Fu, Crandall. Meta-Reinforced Synthetic Data for One-Shot Fine-Grained Visual Recognition. In NIPS, 2019

[2] Spoel et al. Siamese Neural Networks for One-Shot Image Recognition. In ICML Workshop, 2015.

[3] Chen et al. GridMask Data Augmentation, Arxiv.

[4] Wan et al. The caltech-ucsd birds-200-2011 dataset, California Institute of Technology, 2011.

[5] MetaIRNet Slide. https://drive.google.com/file/d/1YQtKEn4ySVqsMMU66dS1JvVhDWpO4Qqz/view?usp=sharing

谢谢

THANK YOU