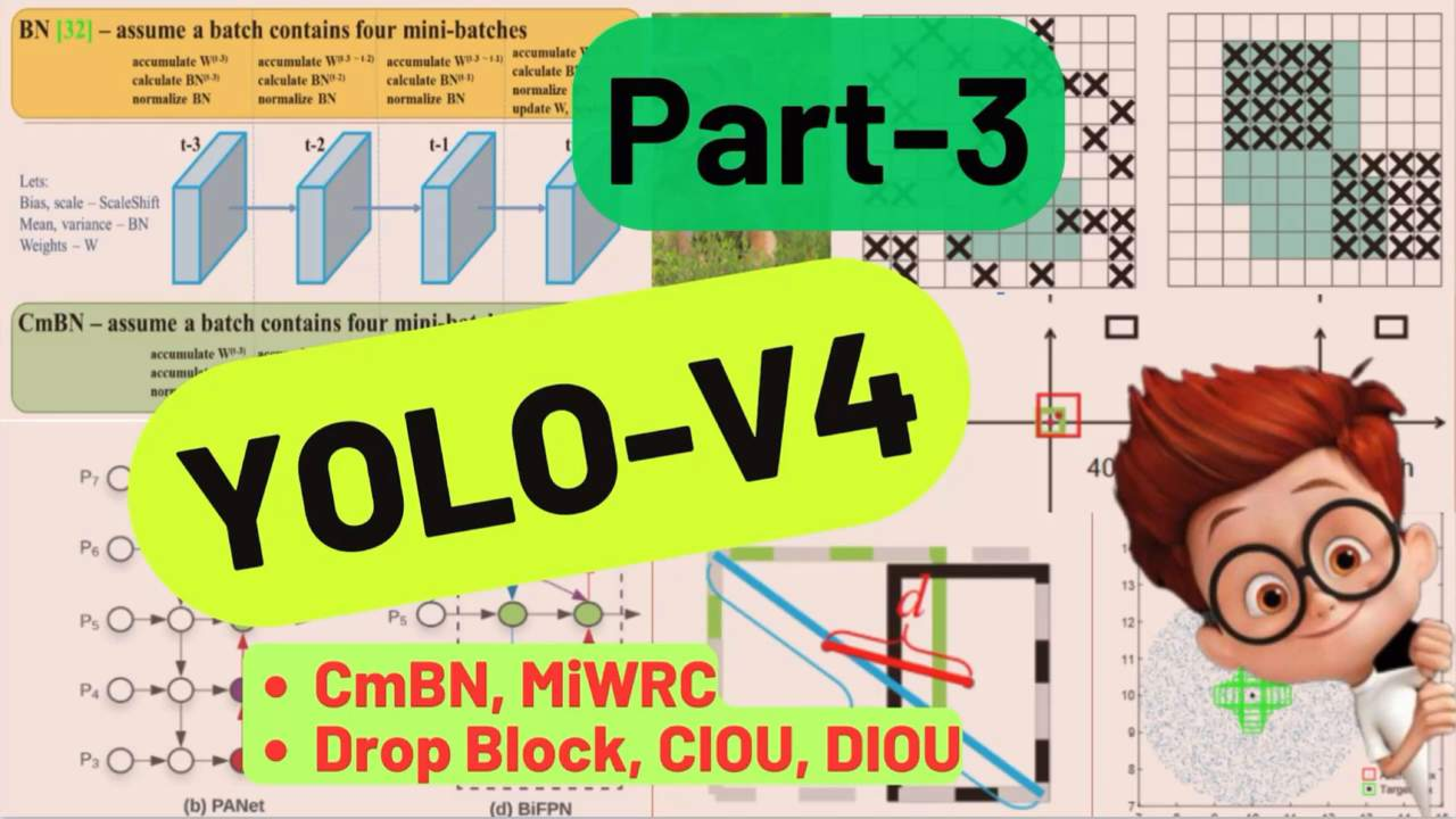
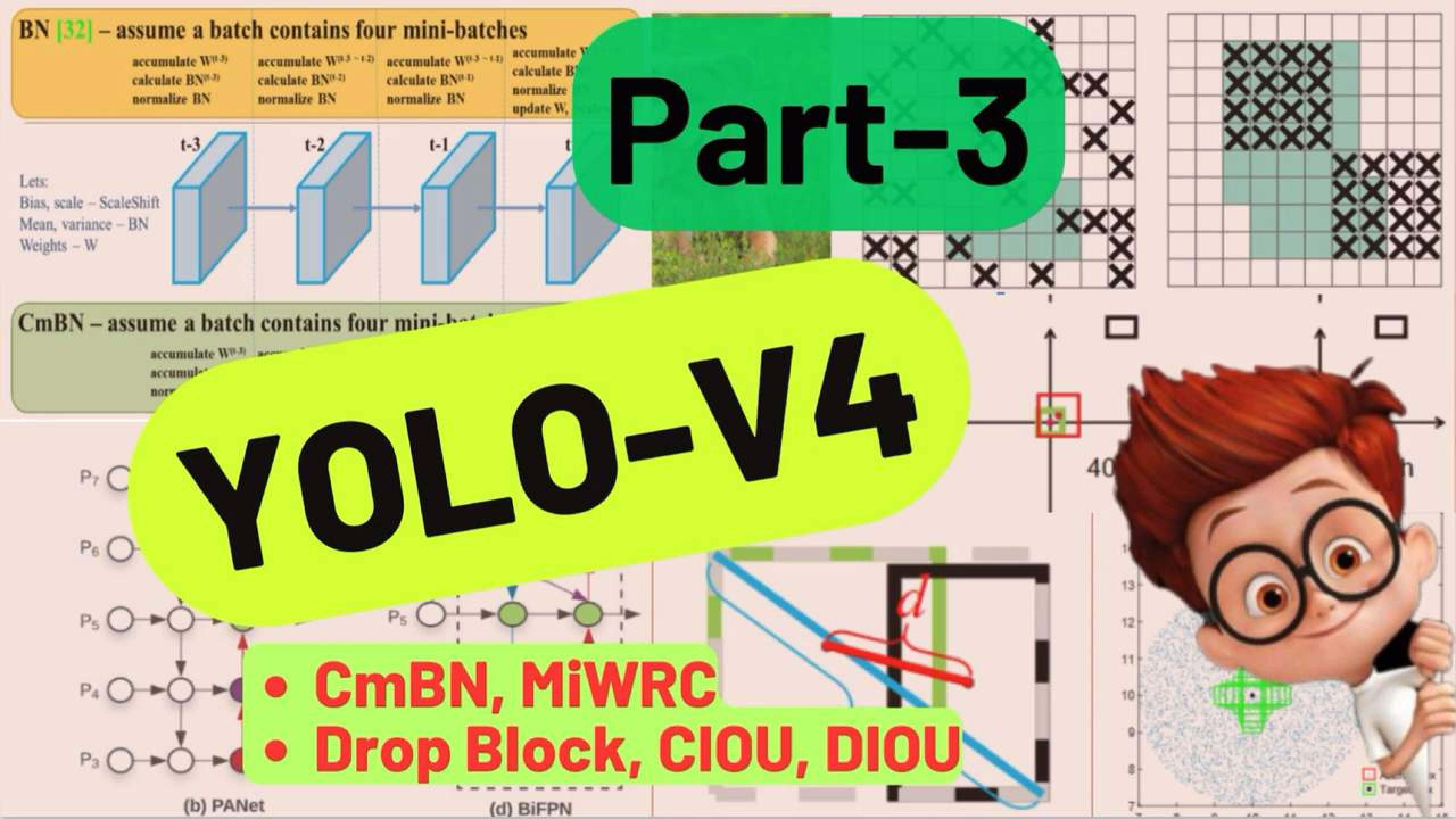
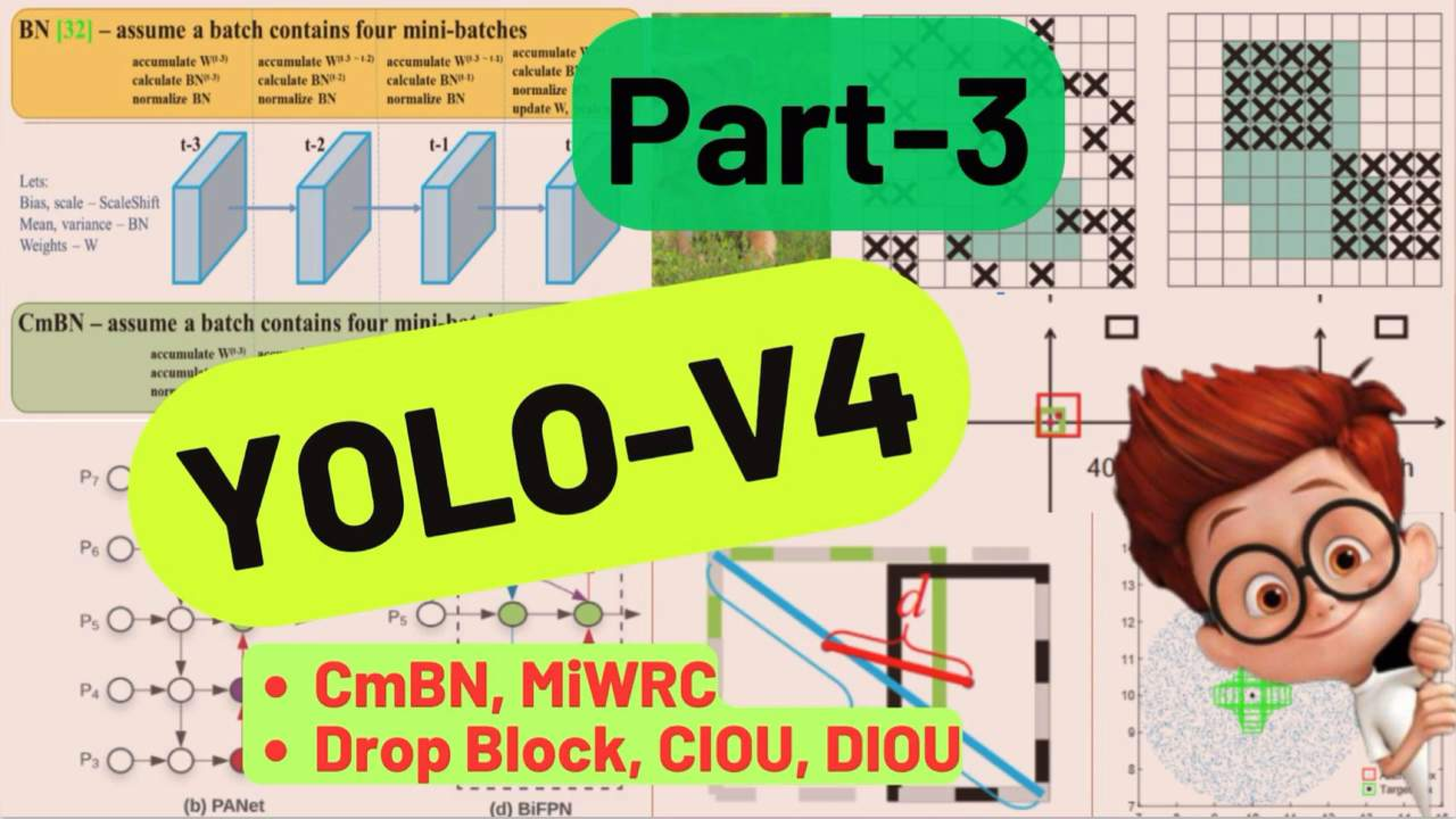


Part-3
YOLO-V4
• CmBN, MiWRC
• Drop Block, CIOU, DIOU
BN [32] – assume a batch contains four mini-batches
CmBN – assume a batch contains four mini-batches
Lets:
Bias, scale – ScaleShift
Mean, variance – BN
Weights – W
(b) PANet
(d) BiFPN


BN [32] – assume a batch contains four mini-batches
accumulate $W^{(t-3)}$
calculate $BN^{(t-3)}$
normalize BN
accumulate $W^{(t-3 \sim t-2)}$
calculate $BN^{(t-2)}$
normalize BN
accumulate $W^{(t-3 \sim t-1)}$
calculate $BN^{(t-1)}$
normalize BN
t-3
t-2
t-1
Lets:
Bias, scale – ScaleShift
Mean, variance – BN
Weights – W
CmBN – assume a batch contains four mini-bat
accumulate $W^{(t-3)}$
Part-3
YOLO-V4
• CmBN, MiWRC
• Drop Block, CIOU, DIOU
(b) PANet
(d) BiFPN
d


Part-3
YOLO-V4
• CmBN, MiWRC
• Drop Block, CIOU, DIOU
BN [32] – assume a batch contains four mini-batches
accumulate W
calculate BN
normalize BN
t-3
t-2
t-1
Lets:
Bias, scale – ScaleShift
Mean, variance – BN
Weights – W
(b) PANet
(d) BiFPN

In Previous video..

- **Backbone**
  - **DenseNet**
  - **CSPNet**
  - **CSPDarknet-53**
- **Neck**
  - **FPN**
  - **SPP**
  - **PAN**
- **Spatial Attention Module**

In Previous video..

- **Backbone**
  - **DenseNet**
  - **CSPNet**
  - **CSPDarknet-53**
- **Neck**
  - **FPN**
  - **SPP**
  - **PAN**
- **Spatial Attention Module**

In Previous video..

- **Backbone**
  - **DenseNet**
  - **CSPNet**
  - **CSPDarknet-53**
- **Neck**
  - **FPN**
  - **SPP**
  - **PAN**
- **Spatial Attention Module**

BoF & BoS - covered

	Backbone	Detector
Bag of Freebies (BoF)	• CutMix • Mosaic data augmentation • DropBlock • Class label smoothing	• CIoU-loss • Cross mini-Batch Normalization • DropBlock • Mosaic data augmentation • Self-Adversarial Training • Multiple anchors for a single ground truth • Cosine annealing scheduler • Optimal hyperparameters • Random training shapes
Bag of Specials (BoS)	• Mish activation • Cross-stage partial connections (CSP) • Multi-input weighted residual connections (MiWRC)	• Mish activation • SPP-block • SAM-block • PAN path-aggregation block • DIoU-NMS

BoF & BoS - covered

	Backbone	Detector
Bag of Freebies (BoF)	• CutMix • Mosaic data augmentation • DropBlock • Class label smoothing	• CIoU-loss • Cross mini-Batch Normalization • DropBlock • Mosaic data augmentation • Self-Adversarial Training • Multiple anchors for a single ground truth • Cosine annealing scheduler • Optimal hyperparameters • Random training shapes
Bag of Specials (BoS)	• Mish activation • Cross-stage partial connections (CSP) • Multi-input weighted residual connections (MiWRC)	• Mish activation • SPP-block • SAM-block • PAN path-aggregation block • DIoU-NMS

BoF & BoS - covered

	Backbone	Detector
Bag of Freebies (BoF)	• CutMix • Mosaic data augmentation • DropBlock • Class label smoothing	• CIoU-loss • Cross mini-Batch Normalization • DropBlock • Mosaic data augmentation • Self-Adversarial Training • Multiple anchors for a single ground truth • Cosine annealing scheduler • Optimal hyperparameters • Random training shapes
Bag of Specials (BoS)	• Mish activation • Cross-stage partial connections (CSP) • Multi-input weighted residual connections (MiWRC)	• Mish activation • SPP-block • SAM-block • PAN path-aggregation block • DIoU-NMS

In this Video

	Backbone	Detector
Bag of Freebies (BoF)	• CutMix • Mosaic data augmentation • DropBlock • Class label smoothing	• CIoU-loss • Cross mini-Batch Normalization • DropBlock • Mosaic data augmentation • Self-Adversarial Training • Multiple anchors for a single ground truth • Cosine annealing scheduler • Optimal hyperparameters • Random training shapes
Bag of Specials (BoS)	• Multi-input weighted residual connections (MiWRC)	• DIoU-NMS

In this Video

	Backbone	Detector
Bag of Freebies (BoF)	• CutMix • Mosaic data augmentation • DropBlock • Class label smoothing	• CIoU-loss • Cross mini-Batch Normalization • DropBlock • Mosaic data augmentation • Self-Adversarial Training • Multiple anchors for a single ground truth • Cosine annealing scheduler • Optimal hyperparameters • Random training shapes
Bag of Specials (BoS)	• Multi-input weighted residual connections (MiWRC)	• DIoU-NMS

In this Video

	Backbone	Detector
Bag of Freebies (BoF)	• CutMix • Mosaic data augmentation • DropBlock • Class label smoothing	• CIoU-loss • Cross mini-Batch Normalization • DropBlock • Mosaic data augmentation • Self-Adversarial Training • Multiple anchors for a single ground truth • Cosine annealing scheduler • Optimal hyperparameters • Random training shapes
Bag of Specials (BoS)	• Multi-input weighted residual connections (MiWRC)	• DIoU-NMS

Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

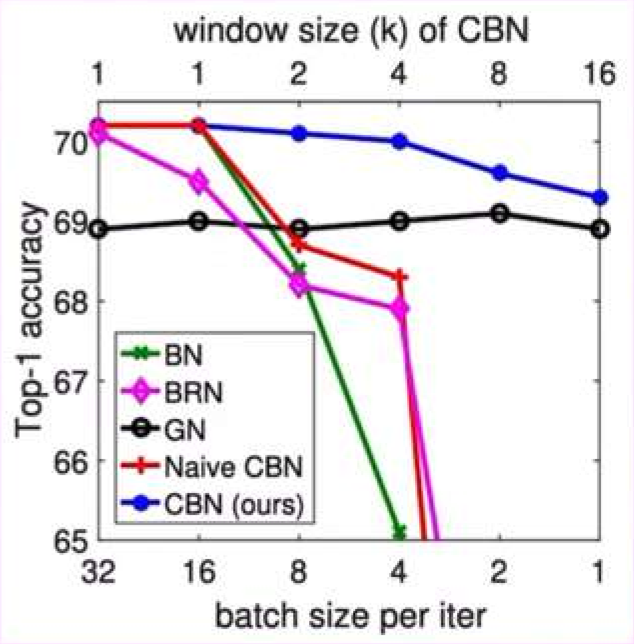
- Normalize:

$$\hat{x}_{ij} = \frac{x_{ij}-\mu_j}{\sqrt{\sigma_j^2+\epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

- Normalize:

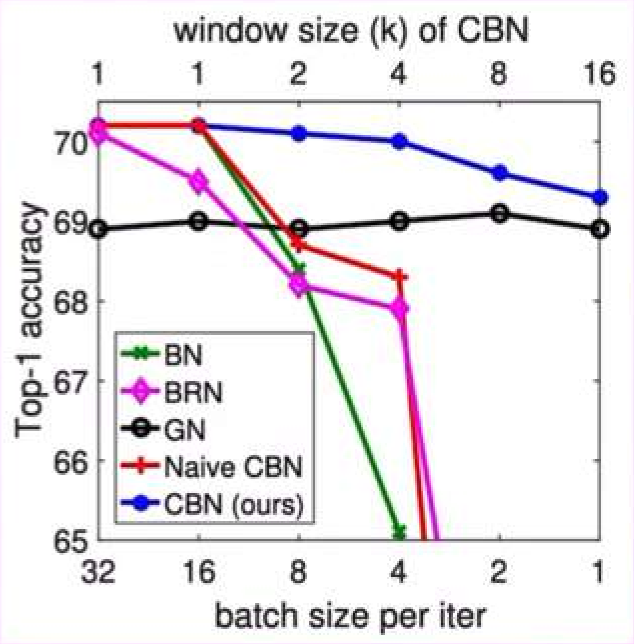
$$\hat{x}_{ij} = \frac{x_{ij}-\mu_j}{\sqrt{\sigma_j^2+\epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes

  (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

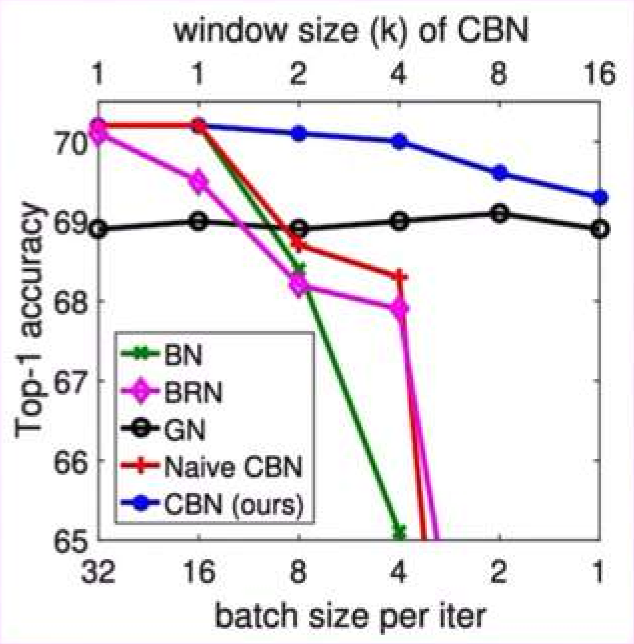
- Normalize:

$$\hat{x}_{ij} = \frac{x_{ij}-\mu_j}{\sqrt{\sigma_j^2+\epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

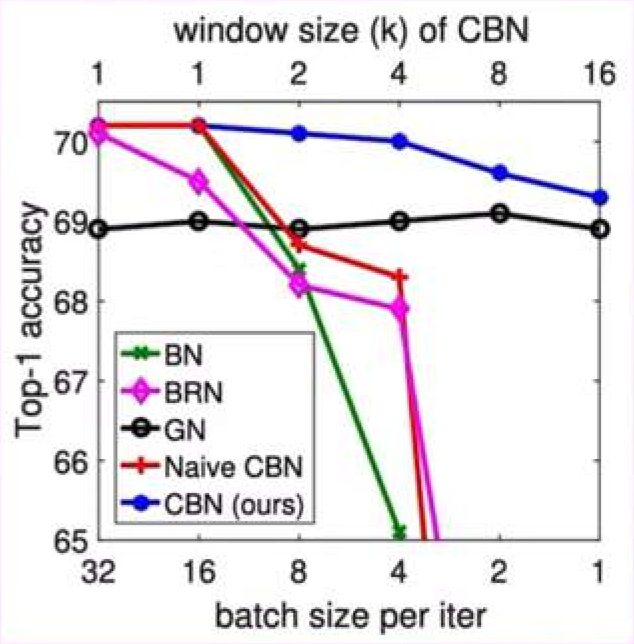
- Normalize:

$$\hat{x}_{ij} = \frac{x_{ij} - \mu_j}{\sqrt{\sigma_j^2 + \epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

- Normalize:

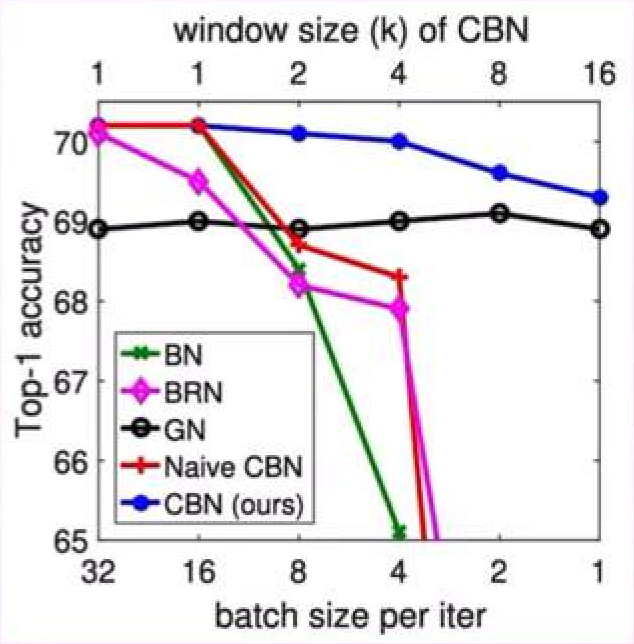
$$\hat{x}_{ij} = \frac{x_{ij}-\mu_j}{\sqrt{\sigma_j^2+\epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes

  (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$

- Variance (σ²):

$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$

- Normalize:

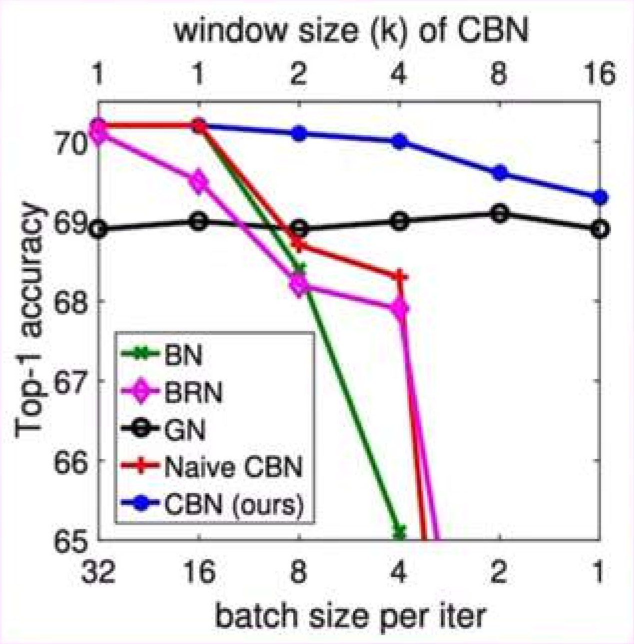
$$\hat{x}_{ij} = \frac{x_{ij}-\mu_j}{\sqrt{\sigma_j^2+\epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes

  (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

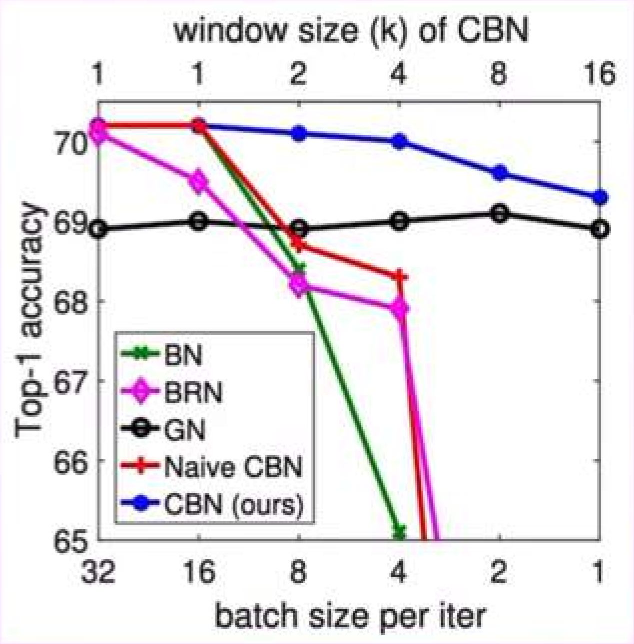
- Normalize:

$$\hat{x}_{ij} = \frac{x_{ij}-\mu_j}{\sqrt{\sigma_j^2+\epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

- Normalize:

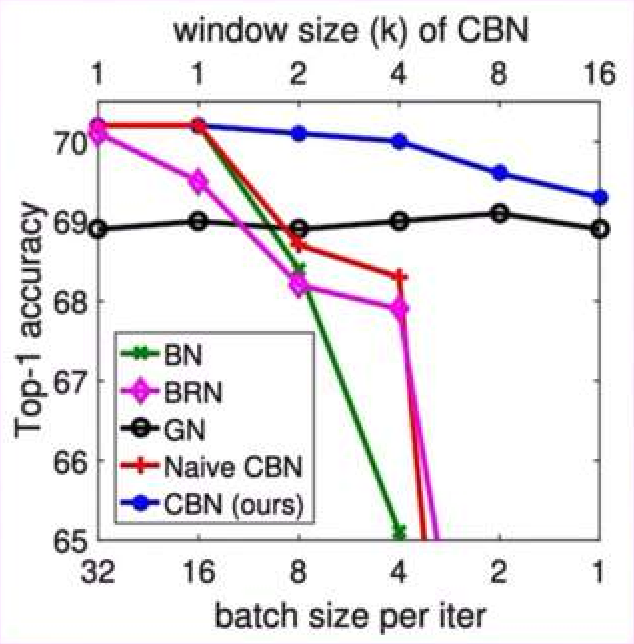
$$\hat{x}_{ij} = \frac{x_{ij}-\mu_j}{\sqrt{\sigma_j^2+\epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes

  (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

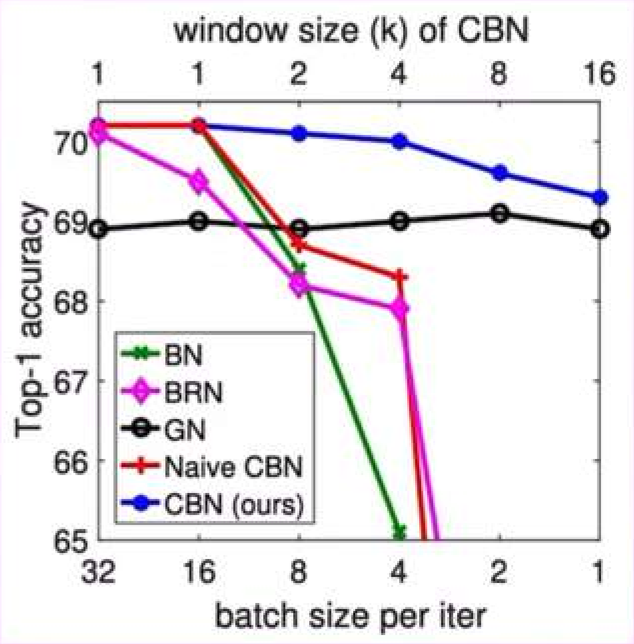
- Normalize:

$$\hat{x}_{ij} = \frac{x_{ij} - \mu_j}{\sqrt{\sigma_j^2 + \epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$\mu_j = \frac{1}{m} \sum_{i=1}^{m} x_{ij}$

- Variance (σ²):

$\sigma_j^2 = \frac{1}{m} \sum_{i=1}^{m} (x_{ij} - \mu_j)^2$

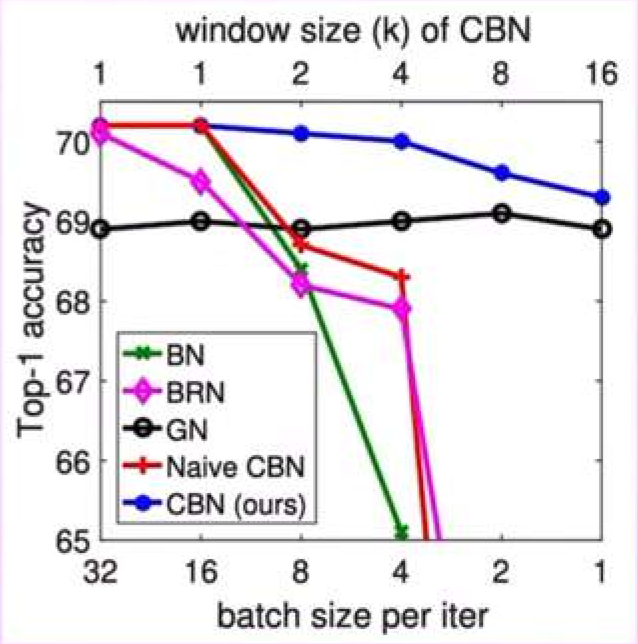
- Normalize:

$\hat{x}_{ij} = \frac{x_{ij} - \mu_j}{\sqrt{\sigma_j^2 + \epsilon}}$

$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$

Drawbacks:

- Fails in case of small batch sizes (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

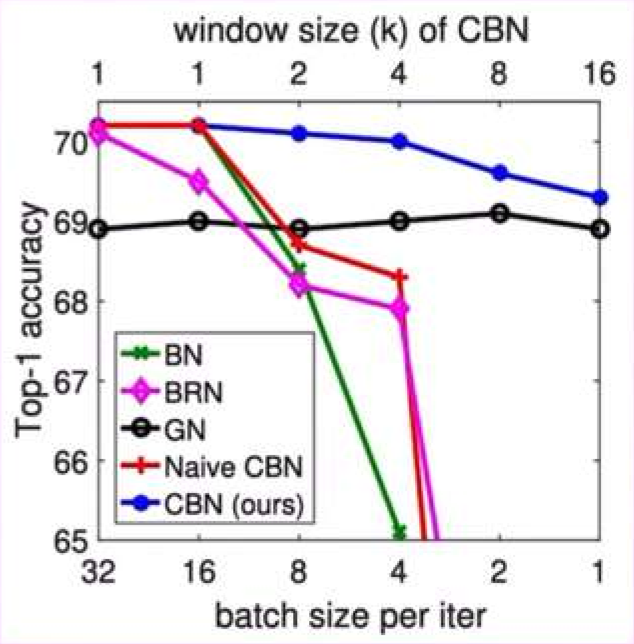
- Normalize:

$$\hat{x}_{ij} = \frac{x_{ij} - \mu_j}{\sqrt{\sigma_j^2 + \epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

- Normalize:

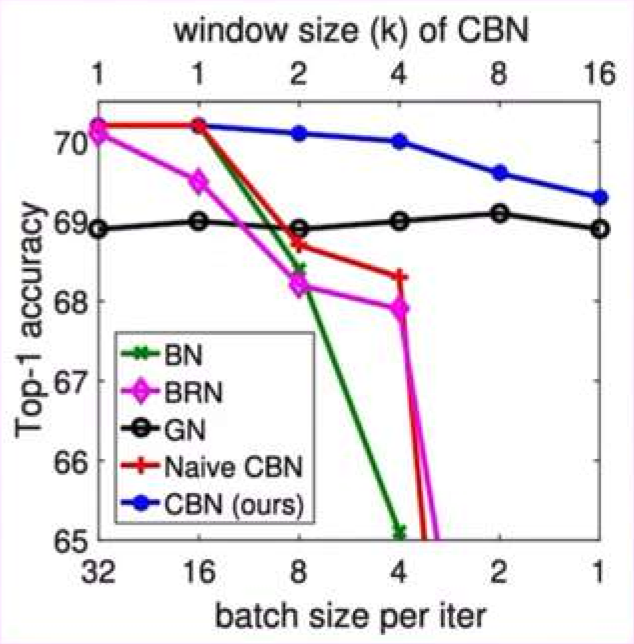
$$\hat{x}_{ij} = \frac{x_{ij}-\mu_j}{\sqrt{\sigma_j^2+\epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes

  (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

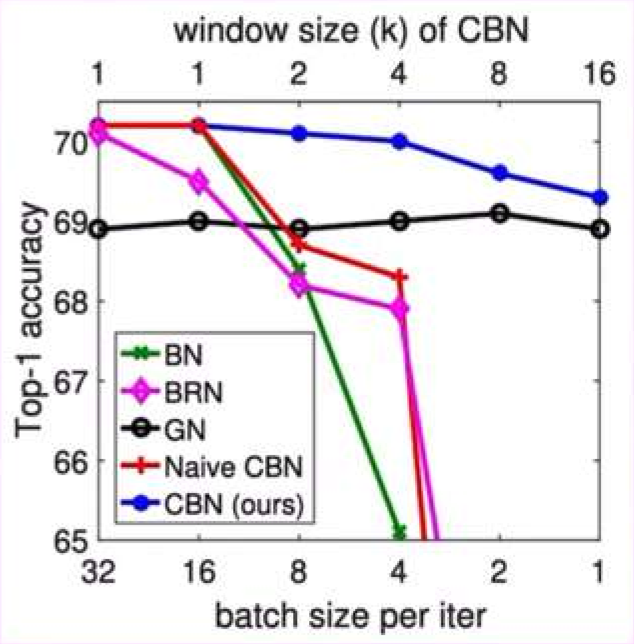
- Normalize:

$$\hat{x}_{ij} = \frac{x_{ij}-\mu_j}{\sqrt{\sigma_j^2+\epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes
  (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

- Normalize:

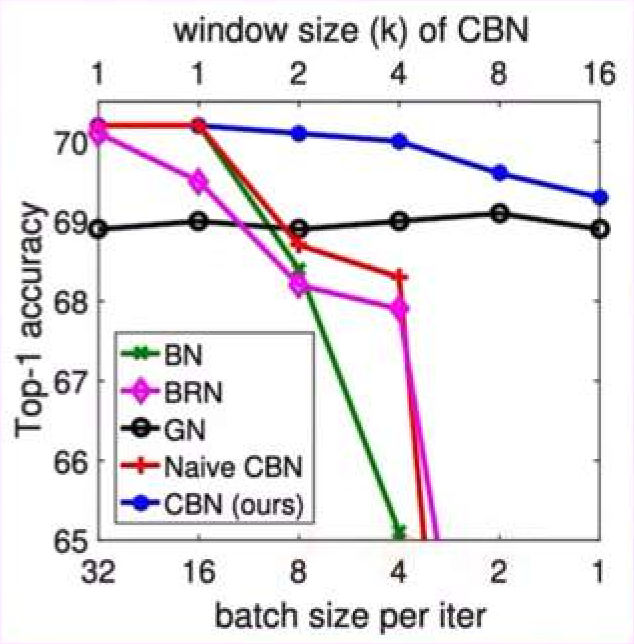
$$\hat{x}_{ij} = \frac{x_{ij} - \mu_j}{\sqrt{\sigma_j^2 + \epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes

  (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ^2):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

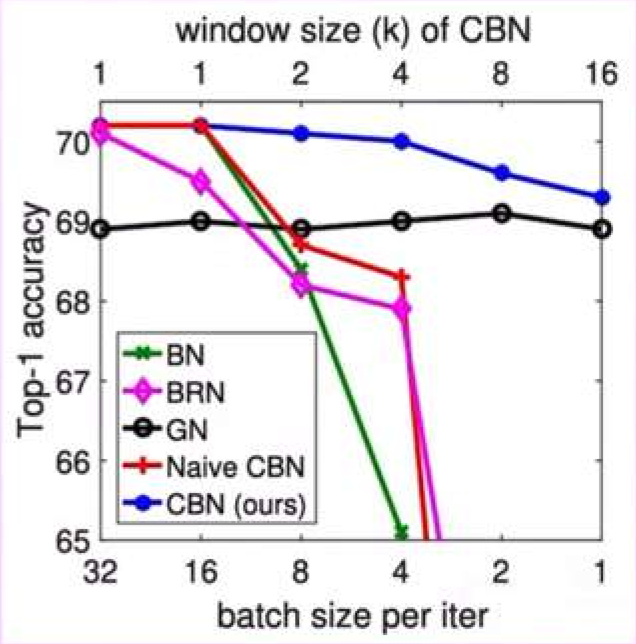
- Normalize:

$$\hat{x}_{ij} = \frac{x_{ij} - \mu_j}{\sqrt{\sigma_j^2 + \epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

- Normalize:

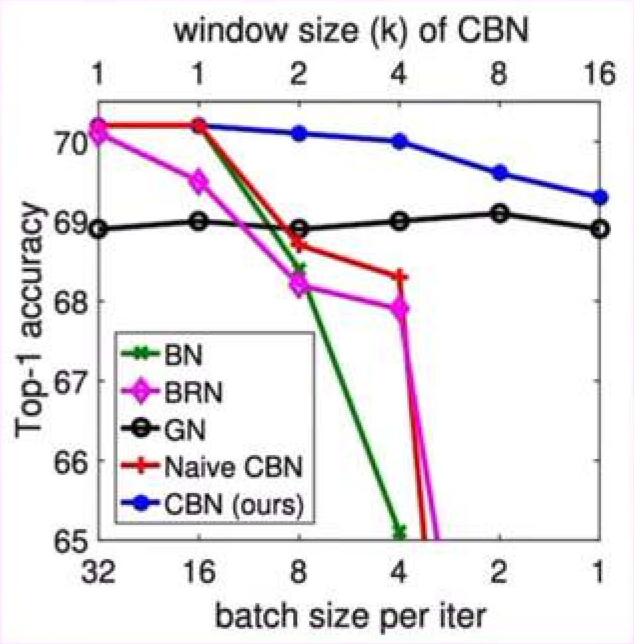
$$\hat{x}_{ij} = \frac{x_{ij}-\mu_j}{\sqrt{\sigma_j^2+\epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes

  (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$$

- Variance (σ²):

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$$

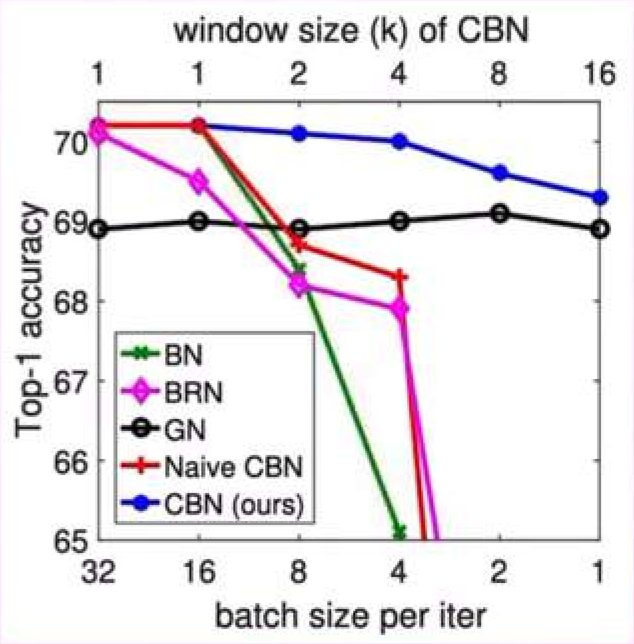
- Normalize:

$$\hat{x}_{ij} = \frac{x_{ij}-\mu_j}{\sqrt{\sigma_j^2+\epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$

- Variance (σ²):

$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$

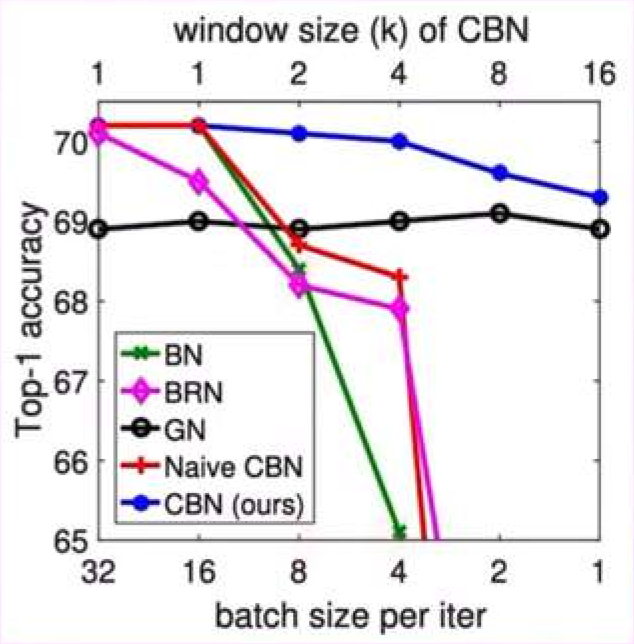
- Normalize:

$$\hat{x}_{ij} = \frac{x_{ij}-\mu_j}{\sqrt{\sigma_j^2+\epsilon}}$$

$$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$$

Drawbacks:

- Fails in case of small batch sizes (mean and variance are inaccurate)



Cross Mini Batch Normalization

Batch Normalization:

- Mean (μ):

  $\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$
- Variance (σ²):

  $\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2$

- Normalize:

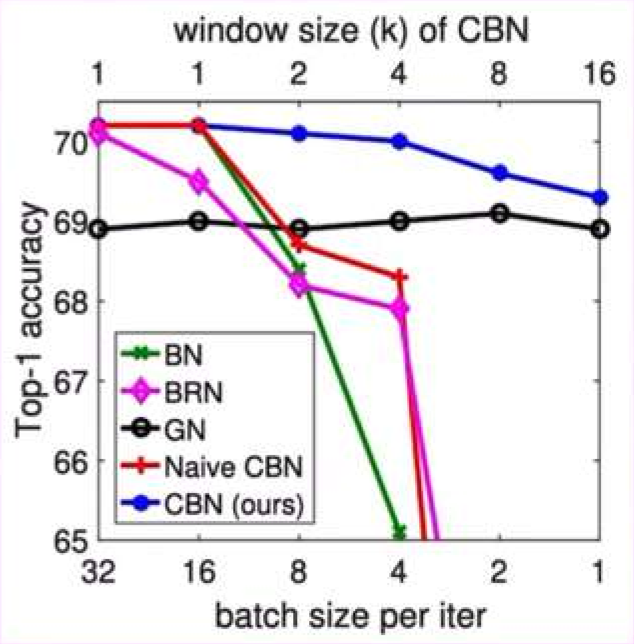
  $\hat{x}_{ij} = \frac{x_{ij} - \mu_j}{\sqrt{\sigma_j^2 + \epsilon}}$

$y_{ij} = \gamma_j \hat{x}_{ij} + \beta_j$

Drawbacks:

- Fails in case of small batch sizes

  (mean and variance are inaccurate)



Cross Mini Batch Normalization

Idea: What if we take previous batches also for mean and variance calculations

- Collect statistics over the entire batch instead of mini-batches

- CBN: CmBN - using Cross mini-Batch Normalization for collecting statistics inside the entire batch, instead of collecting statistics inside a single mini-batch

Cross Mini Batch Normalization

Idea: What if we take previous batches also for mean and variance calculations

- Collect statistics over the entire batch instead of mini-batches

- CBN: CmBN - using Cross mini-Batch Normalization for collecting statistics inside the entire batch, instead of collecting statistics inside a single mini-batch

Cross Mini Batch Normalization

Idea: What if we take previous batches also for mean and variance calculations

- Collect statistics over the entire batch instead of mini-batches

> - CBN: CmBN - using Cross mini-Batch Normalization for collecting statistics inside the entire batch, instead of collecting statistics inside a single mini-batch

Cross Mini Batch Normalization

Idea: What if we take previous batches also for mean and variance calculations

- Collect statistics over the entire batch instead of mini-batches

- CBN: CmBN - using Cross mini-Batch Normalization for collecting statistics inside the entire batch, instead of collecting statistics inside a single mini-batch

Cross Mini Batch Normalization

Idea: What if we take previous batches also for mean and variance calculations

- Collect statistics over the entire batch instead of mini-batches

- CBN: CmBN - using Cross mini-Batch Normalization for collecting statistics inside the entire batch, instead of collecting statistics inside a single mini-batch

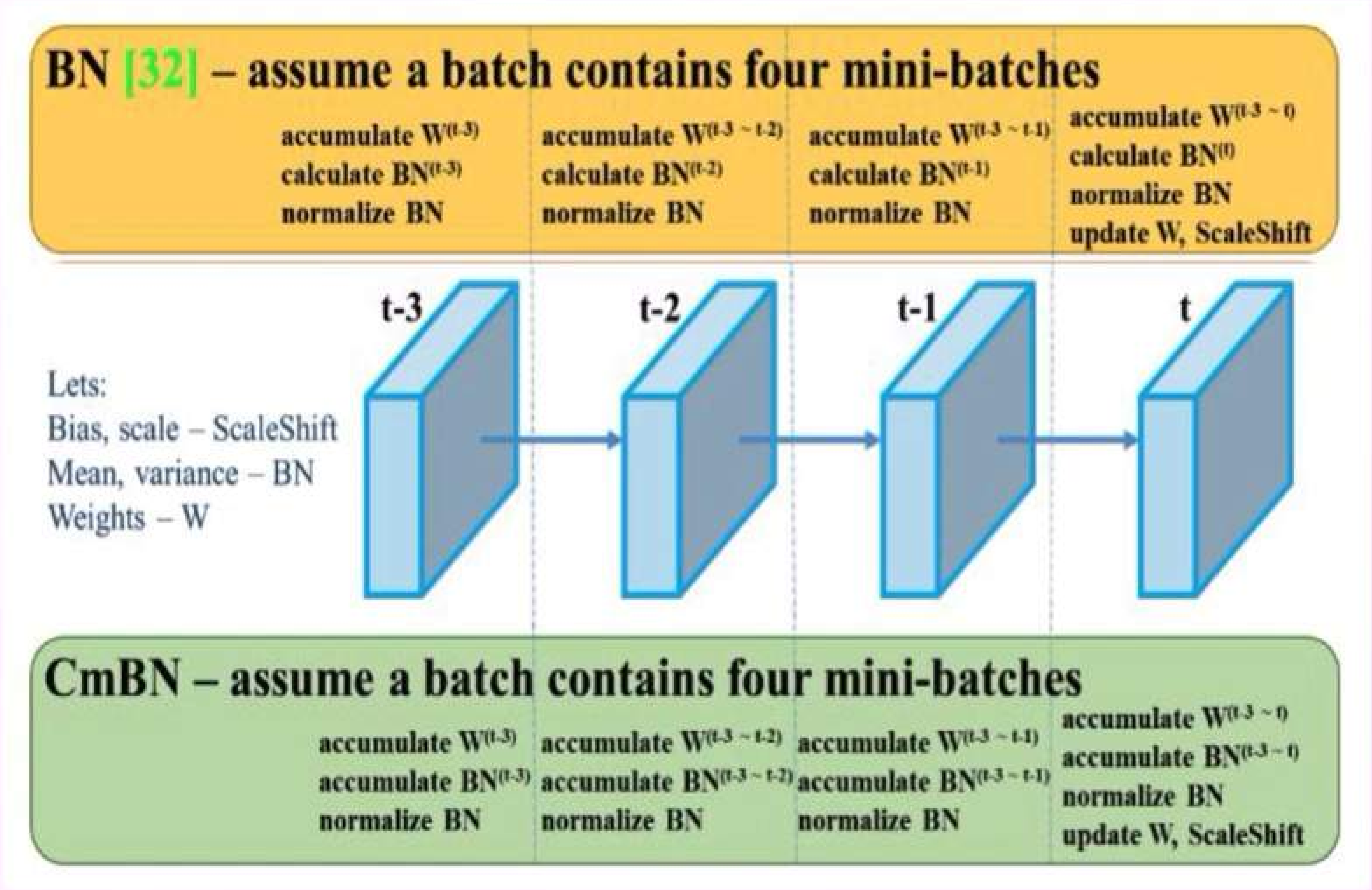
Cross Mini Batch Normalization

Idea: What if we take previous batches also for mean and variance calculations

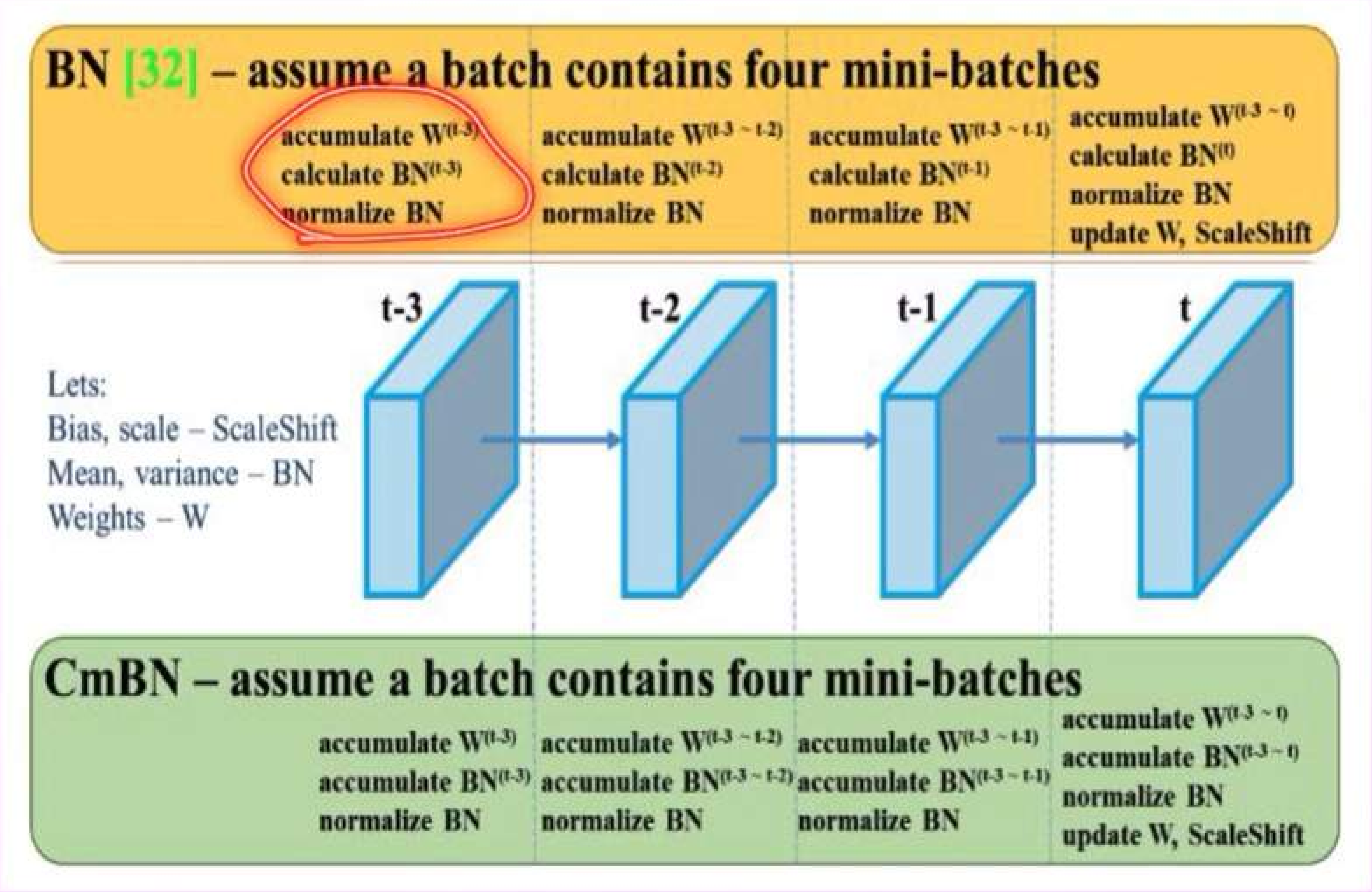
- Collect statistics over the entire batch instead of mini-batches

- CBN: CmBN - using Cross mini-Batch Normalization for collecting statistics inside the entire batch, instead of collecting statistics inside a single mini-batch

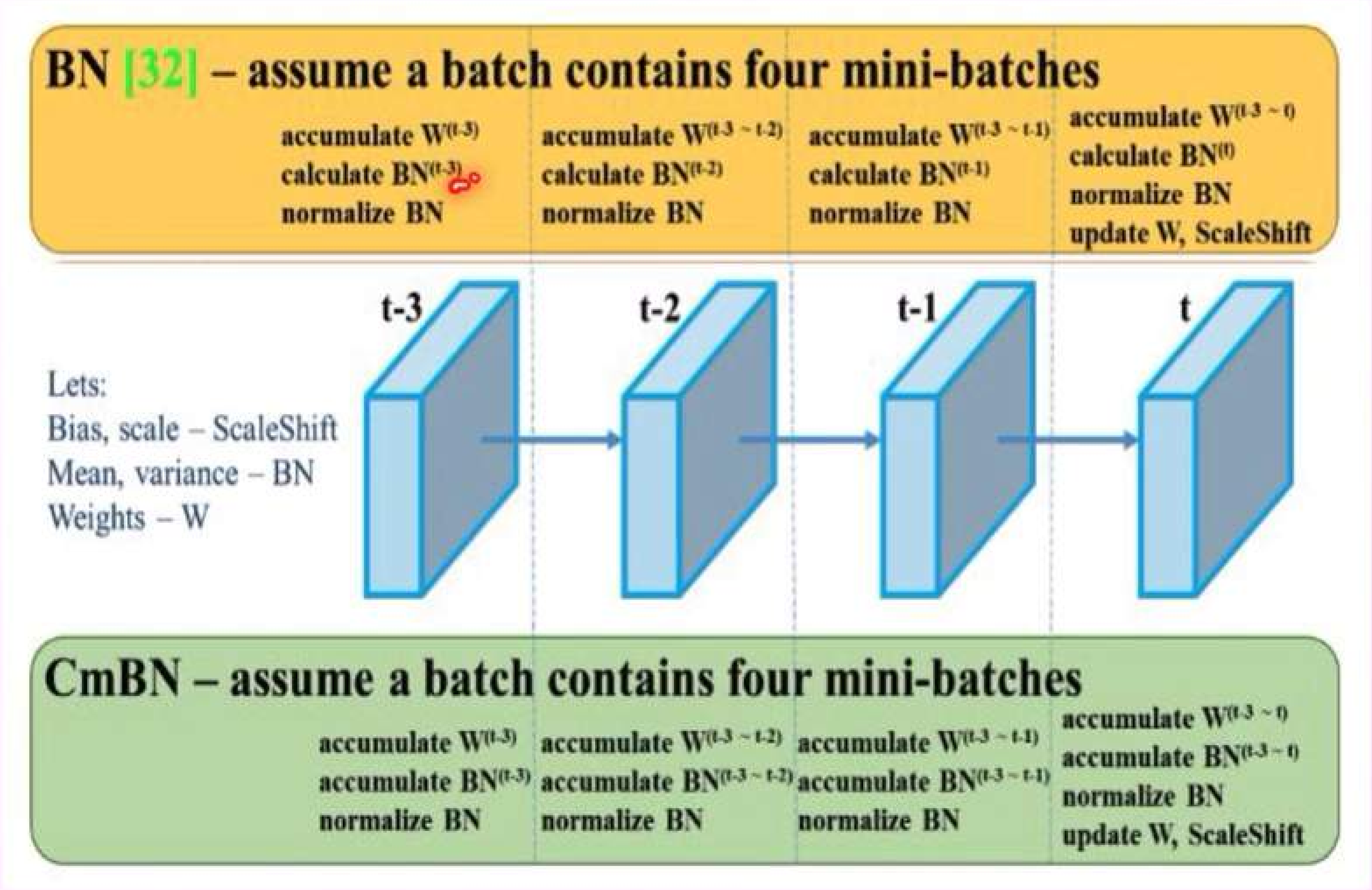
Cross Mini Batch Normalization



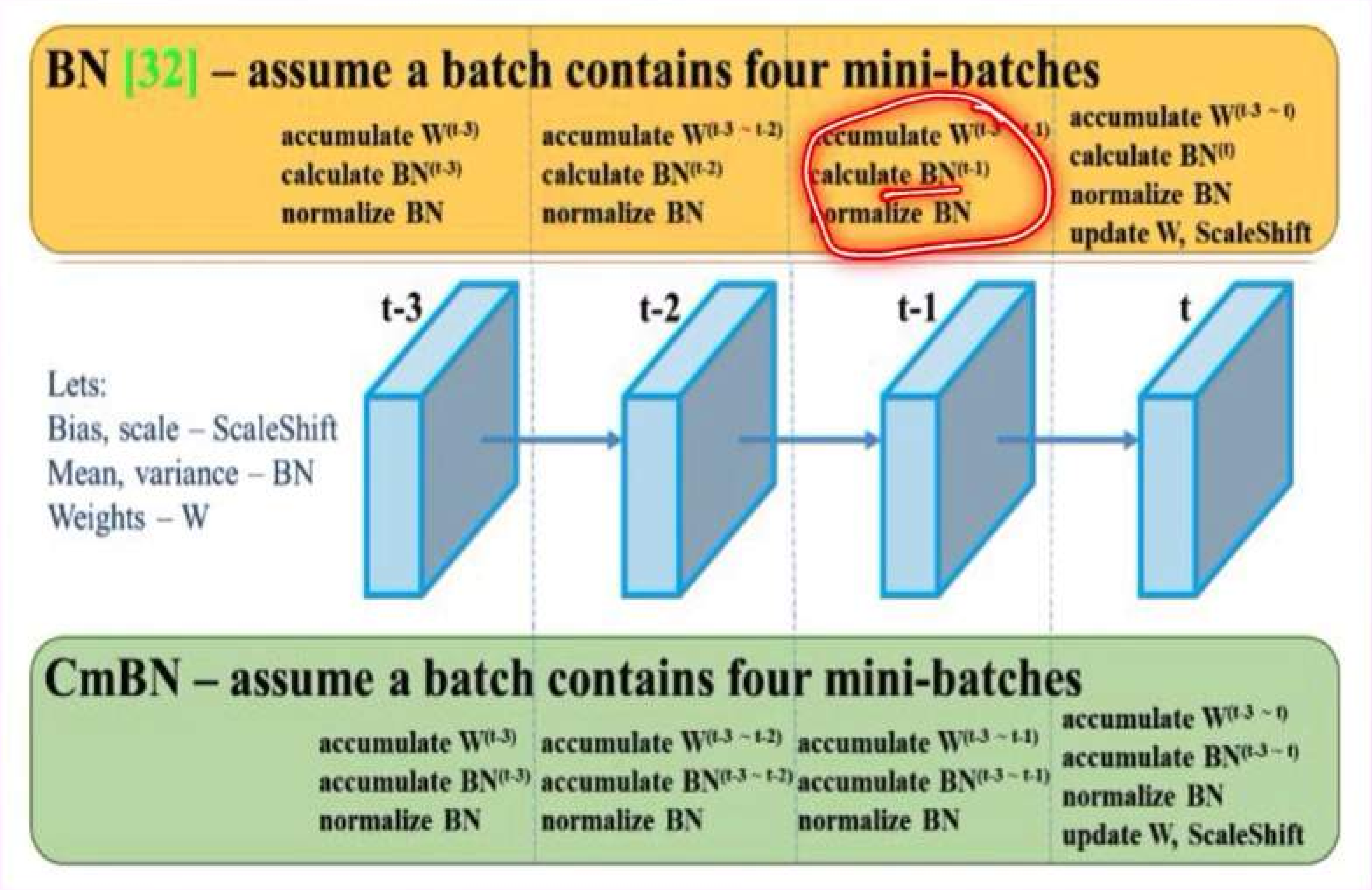
Cross Mini Batch Normalization



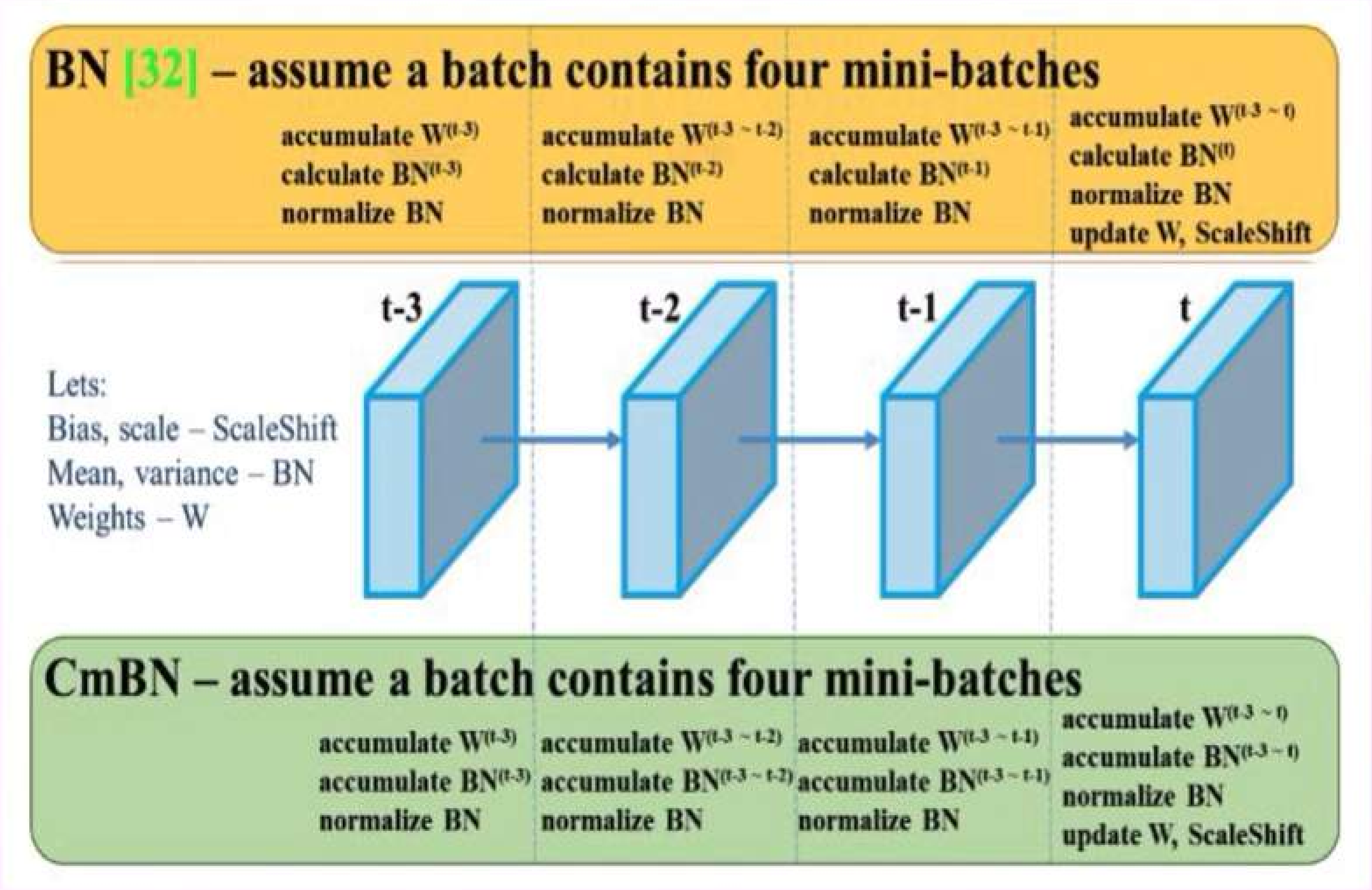
Cross Mini Batch Normalization



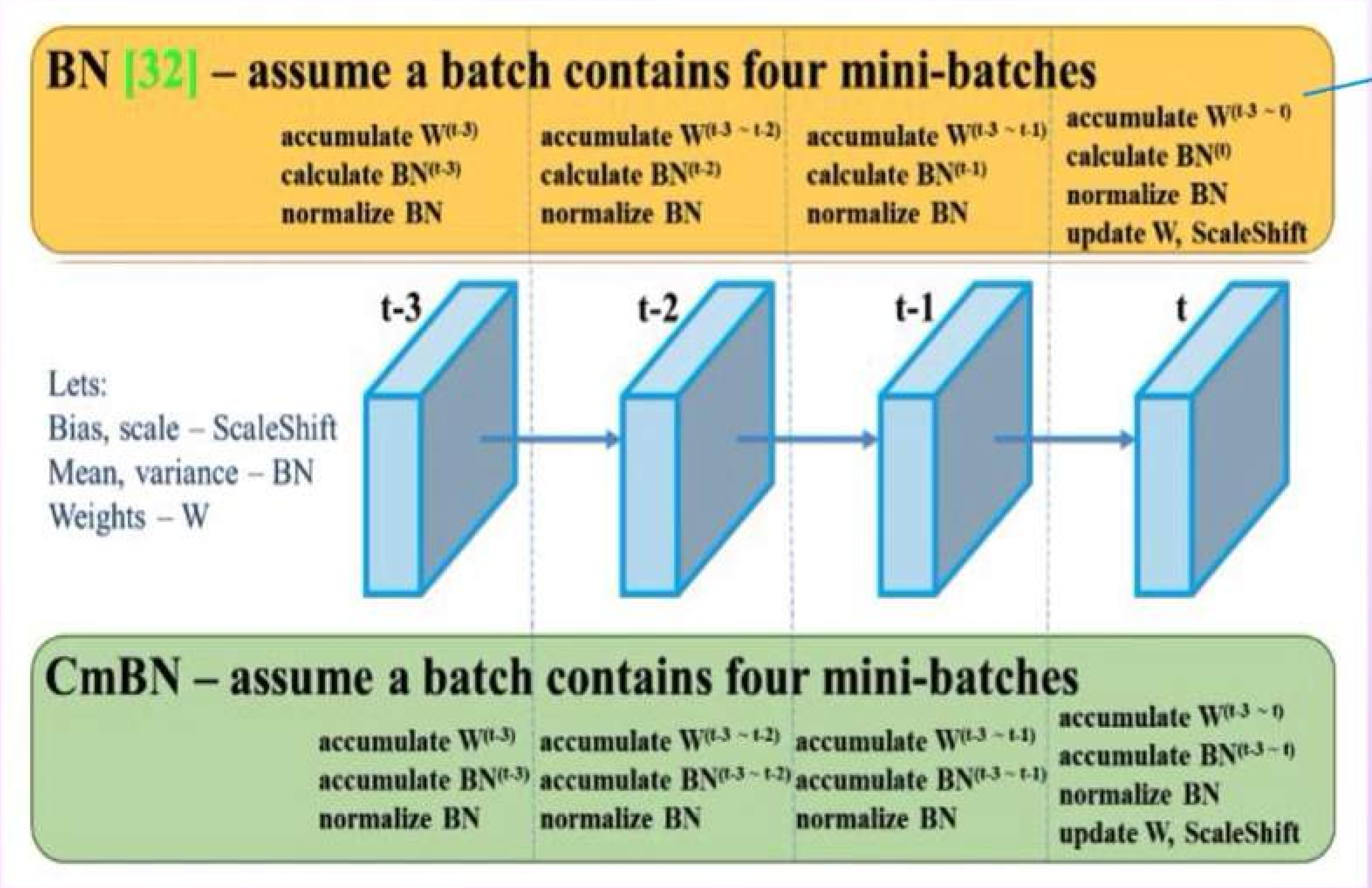
Cross Mini Batch Normalization



Cross Mini Batch Normalization

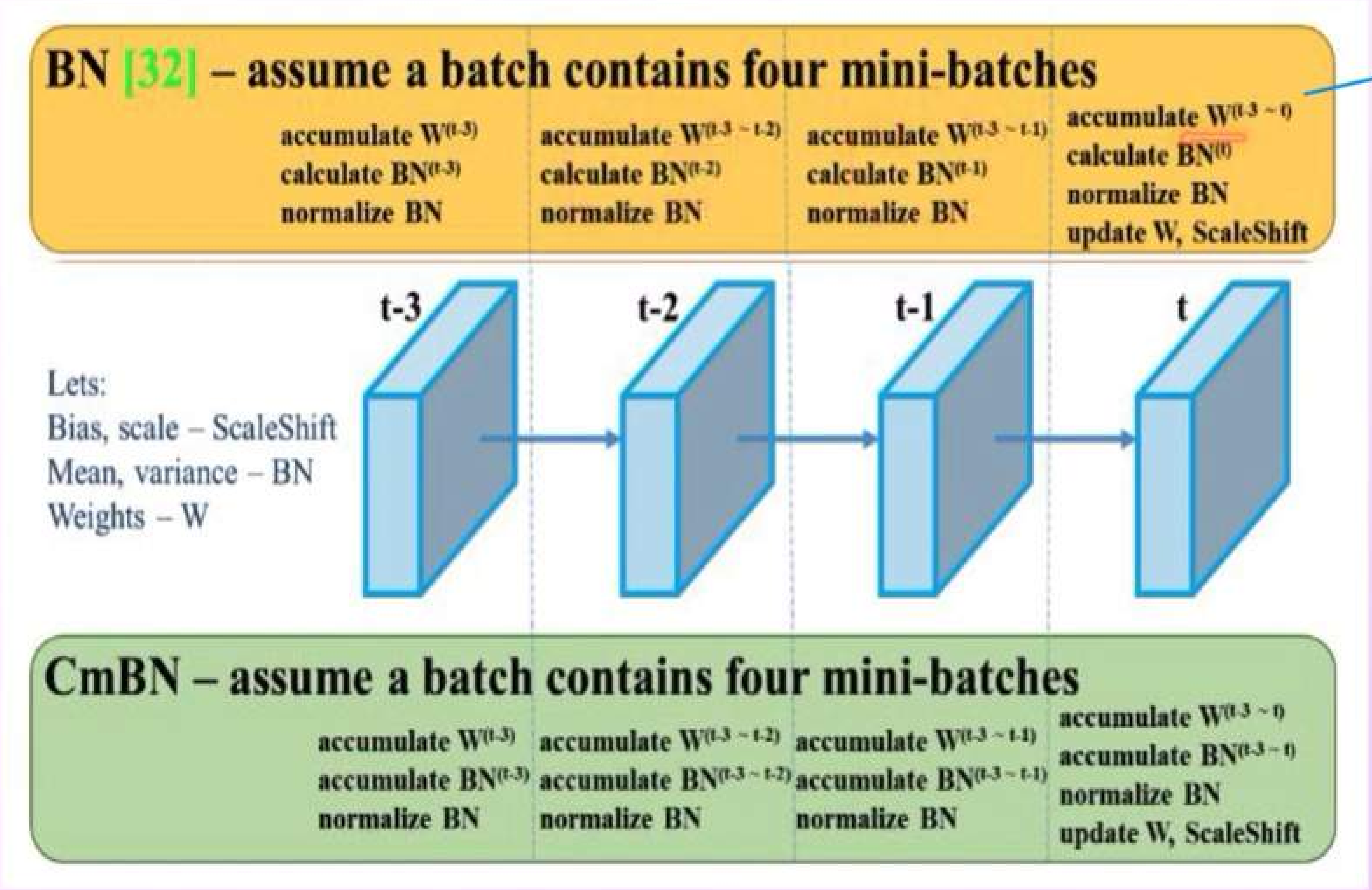


Cross Mini Batch Normalization



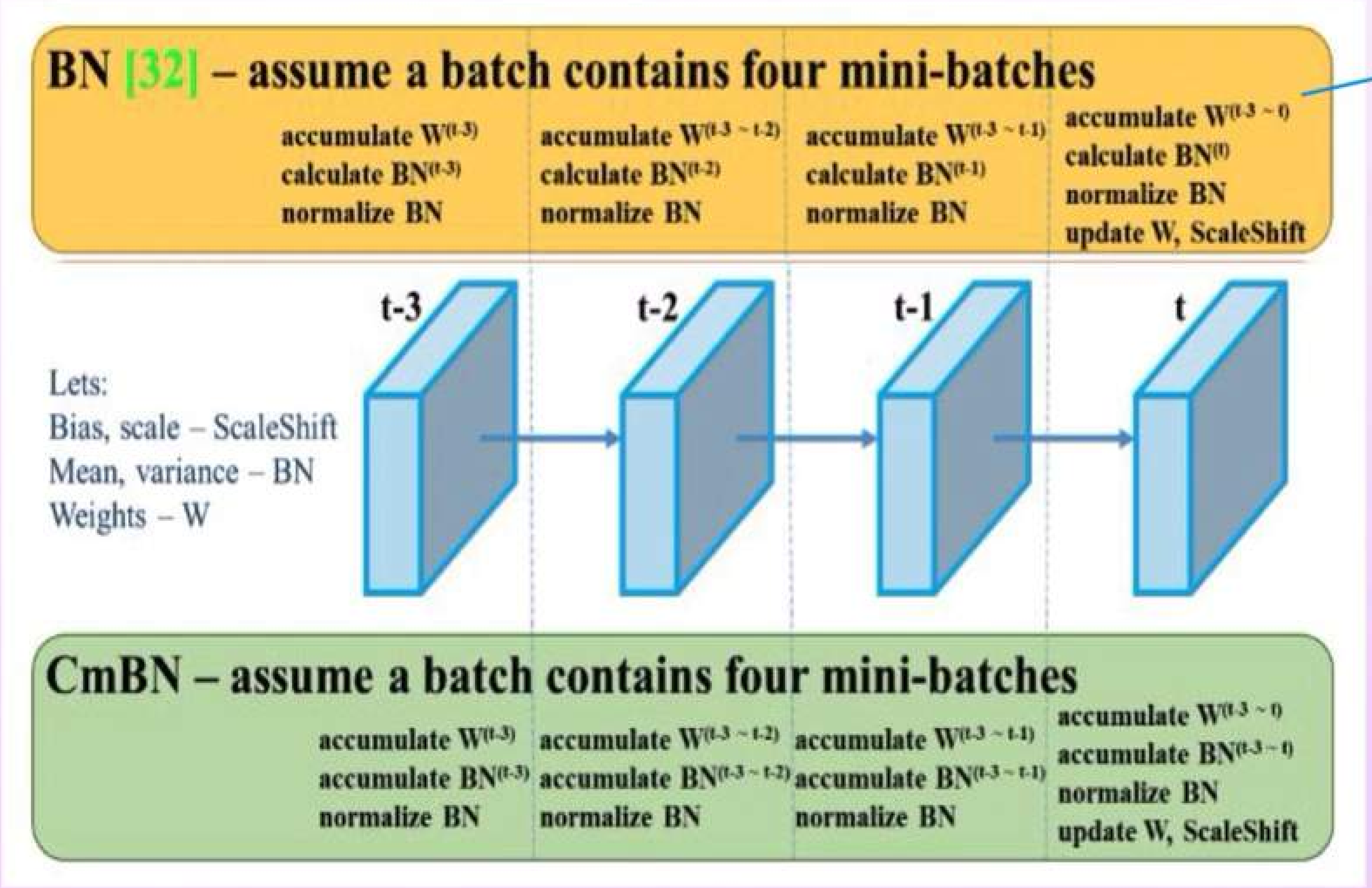
Gradient Accumulation

Cross Mini Batch Normalization



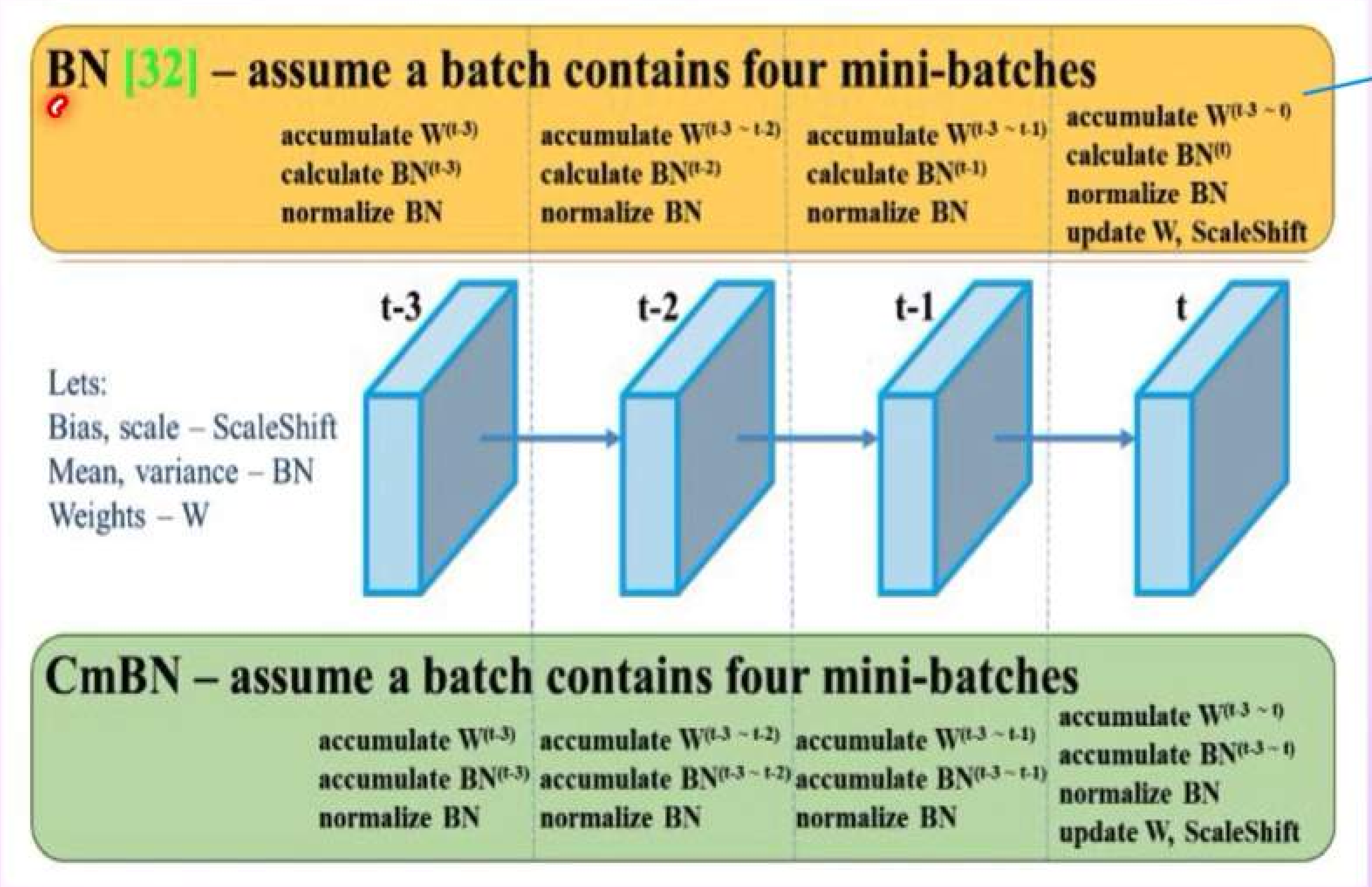
Gradient
Accumulation

Cross Mini Batch Normalization



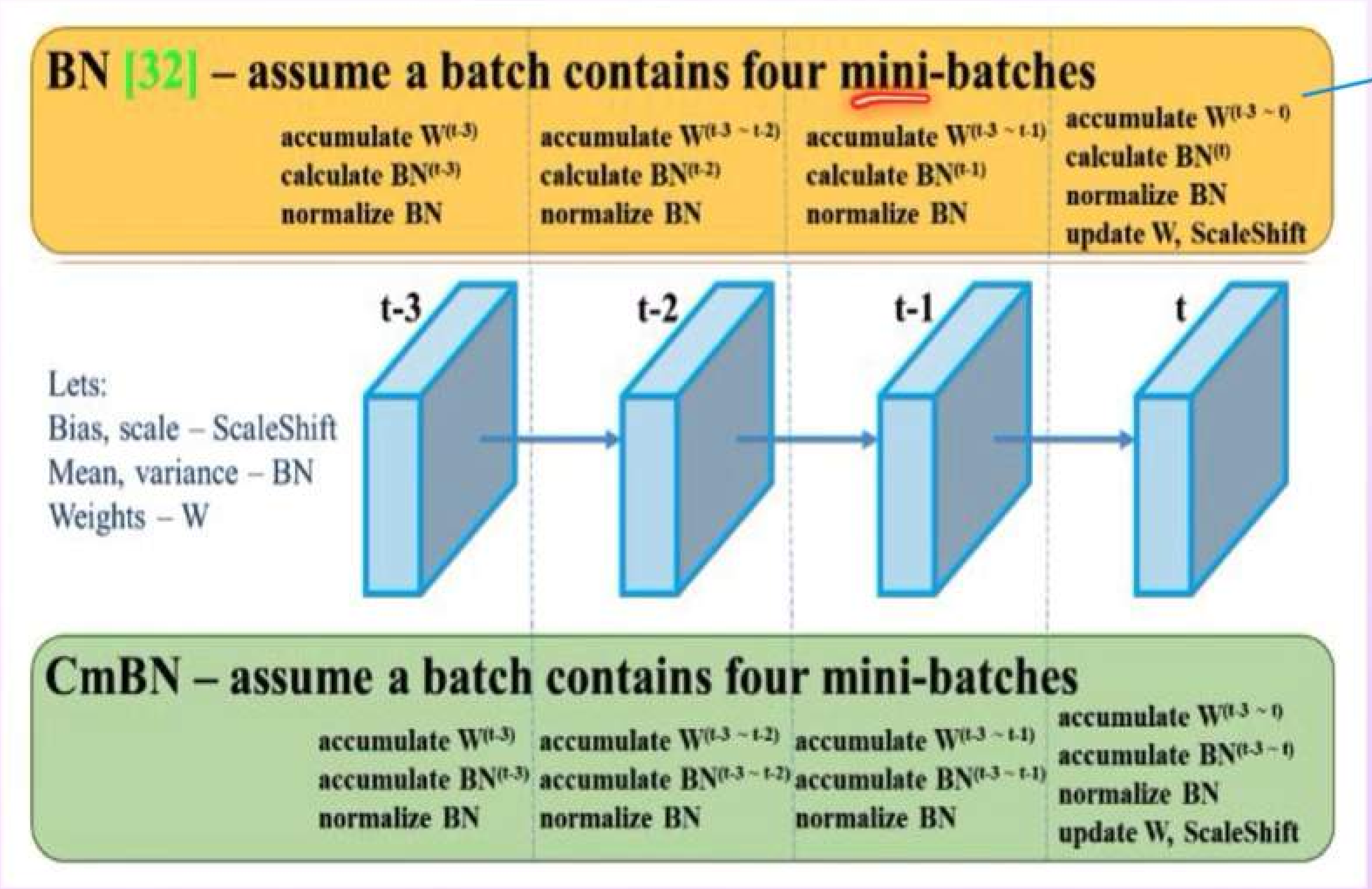
Gradient Accumulation

Cross Mini Batch Normalization



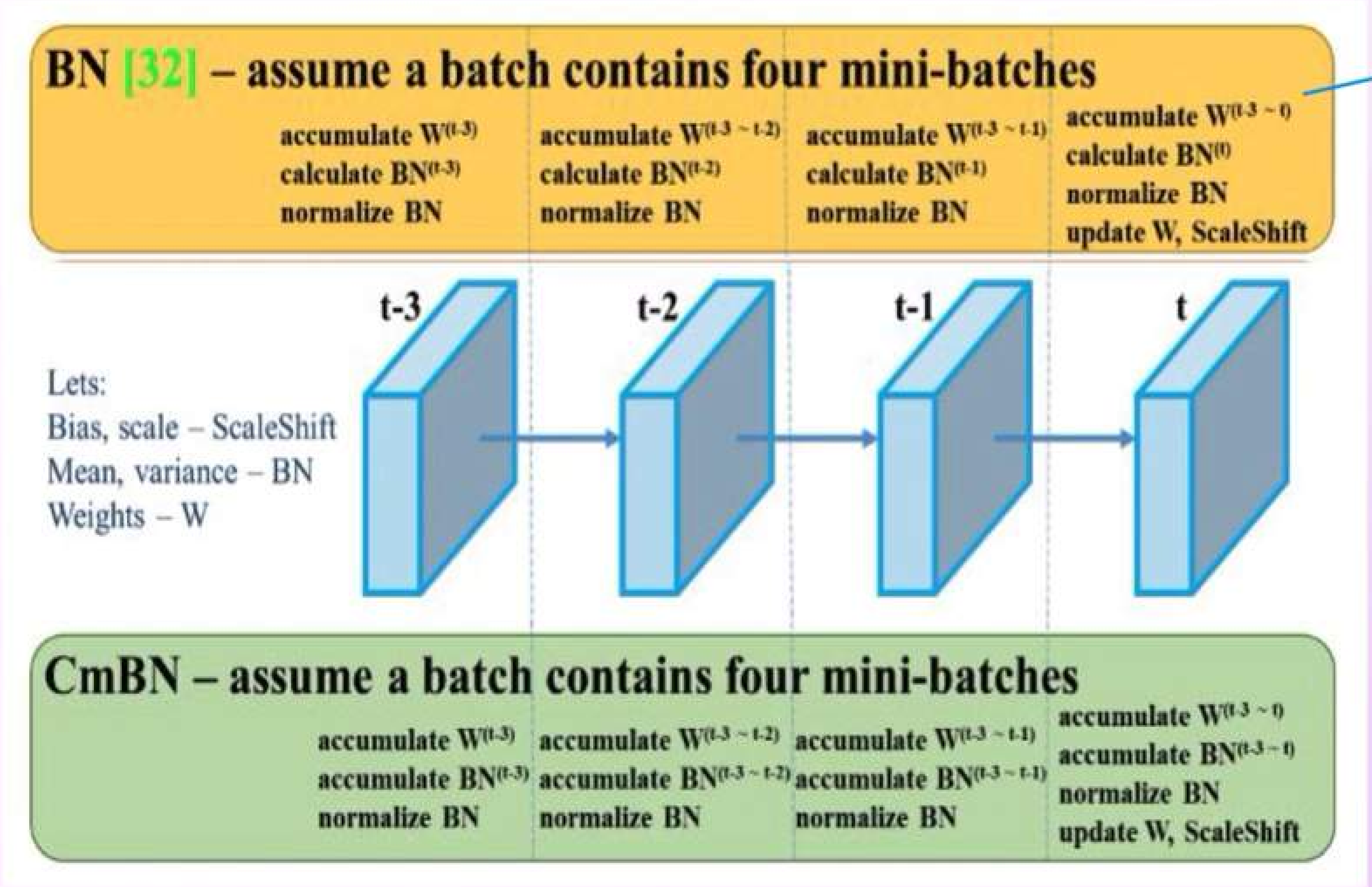
Gradient
Accumulation

Cross Mini Batch Normalization



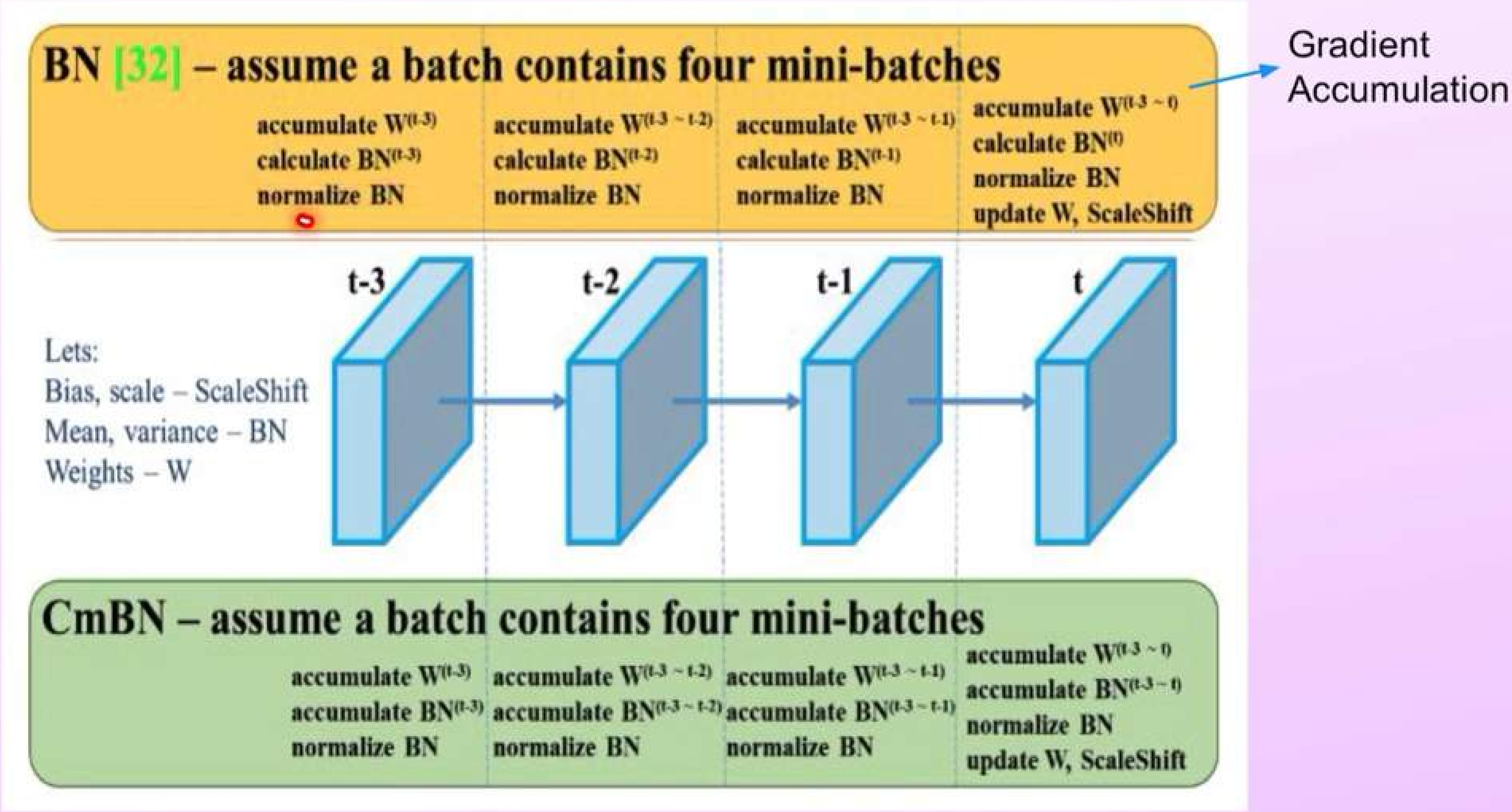
Gradient Accumulation

Cross Mini Batch Normalization

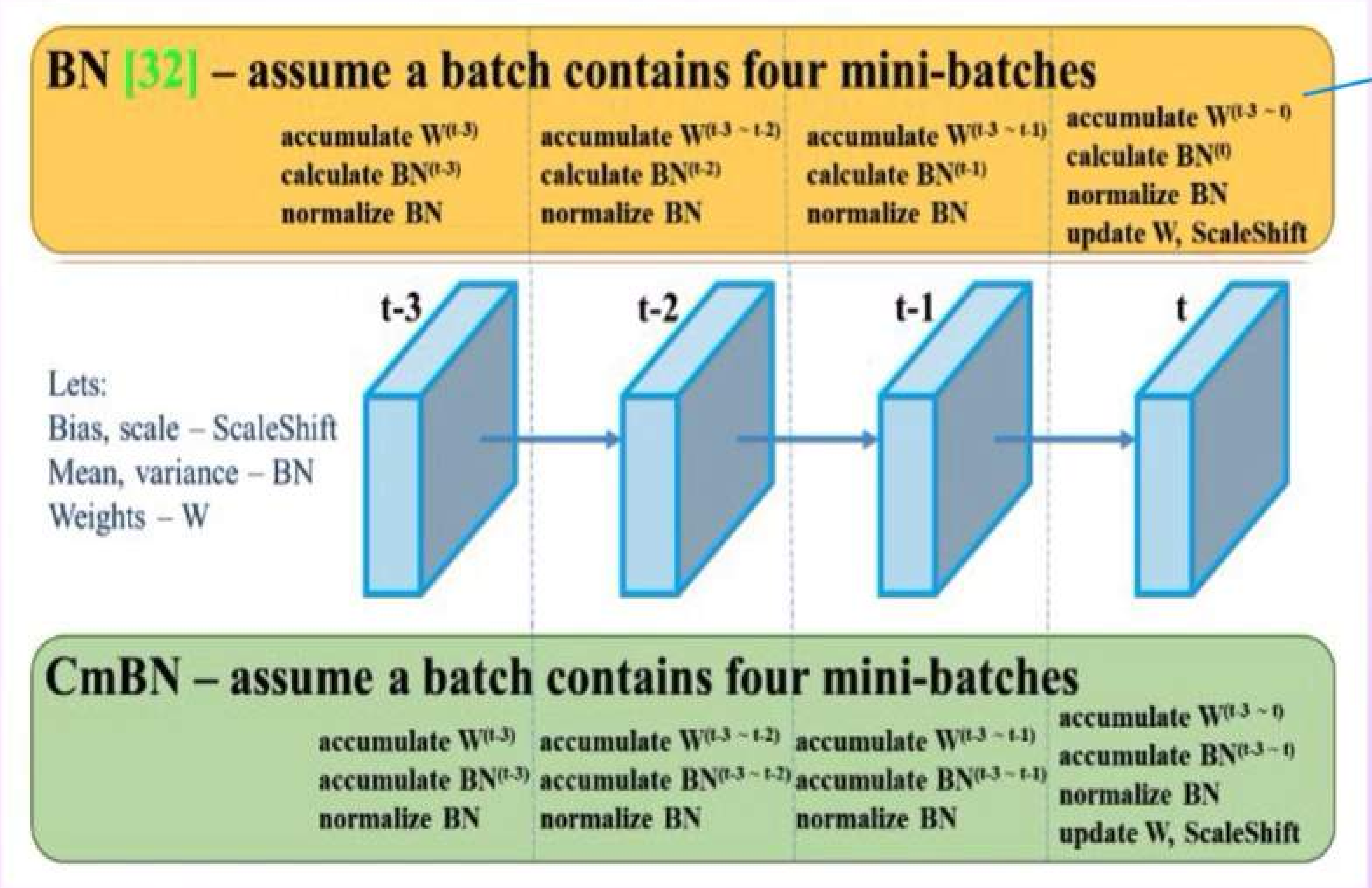


Gradient Accumulation

Cross Mini Batch Normalization

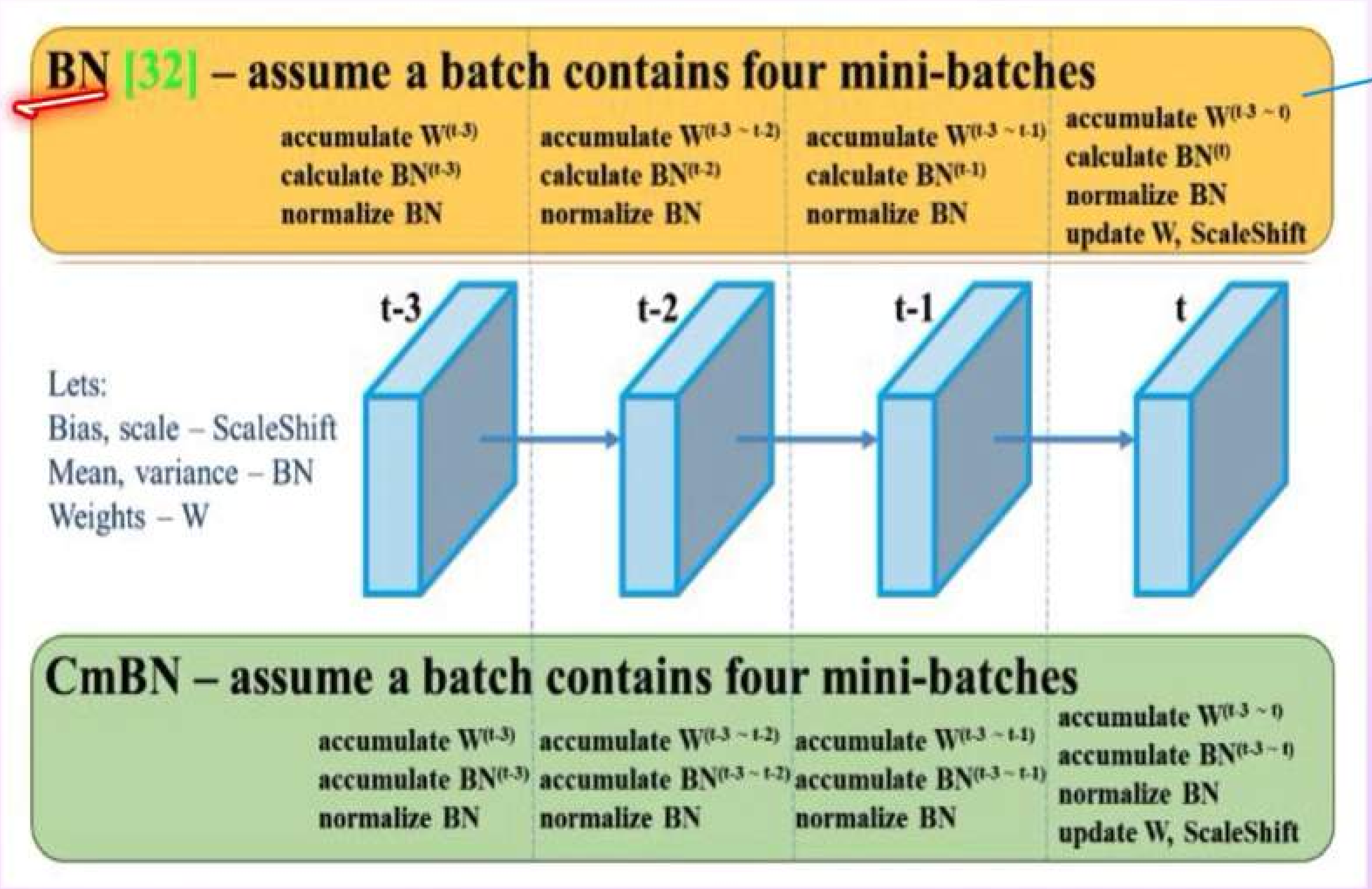


Cross Mini Batch Normalization



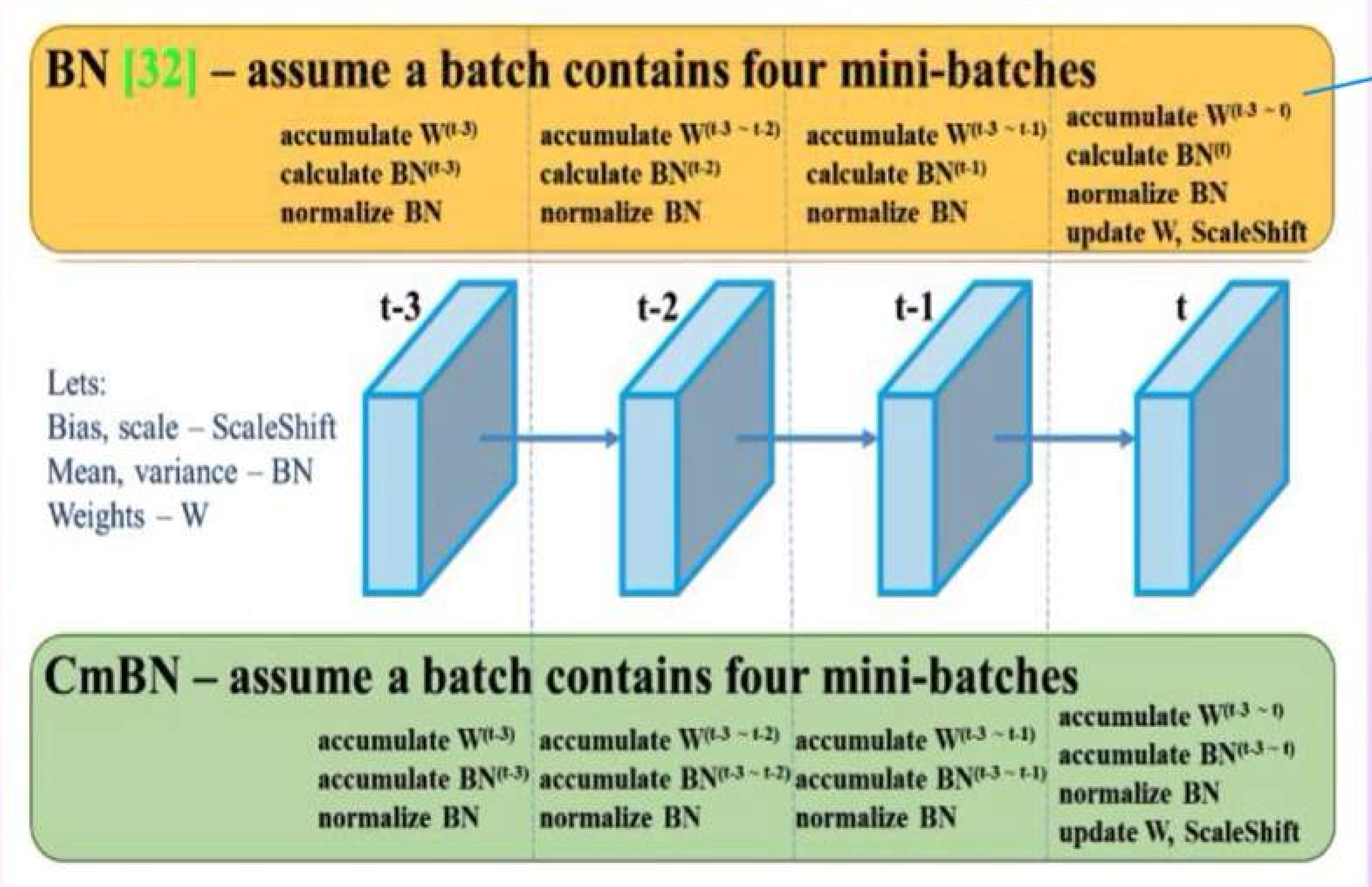
Gradient
Accumulation

Cross Mini Batch Normalization



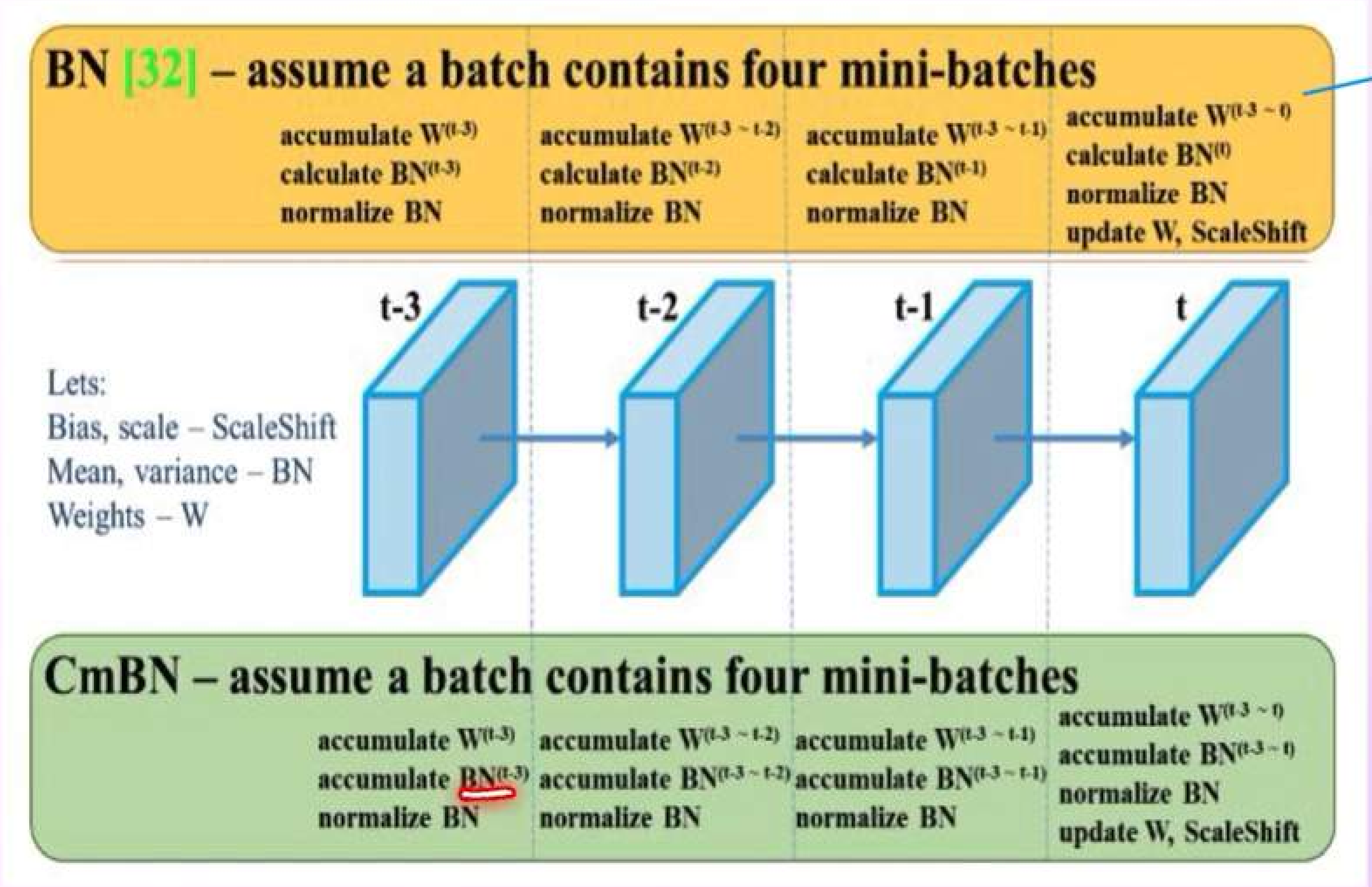
Gradient Accumulation

Cross Mini Batch Normalization



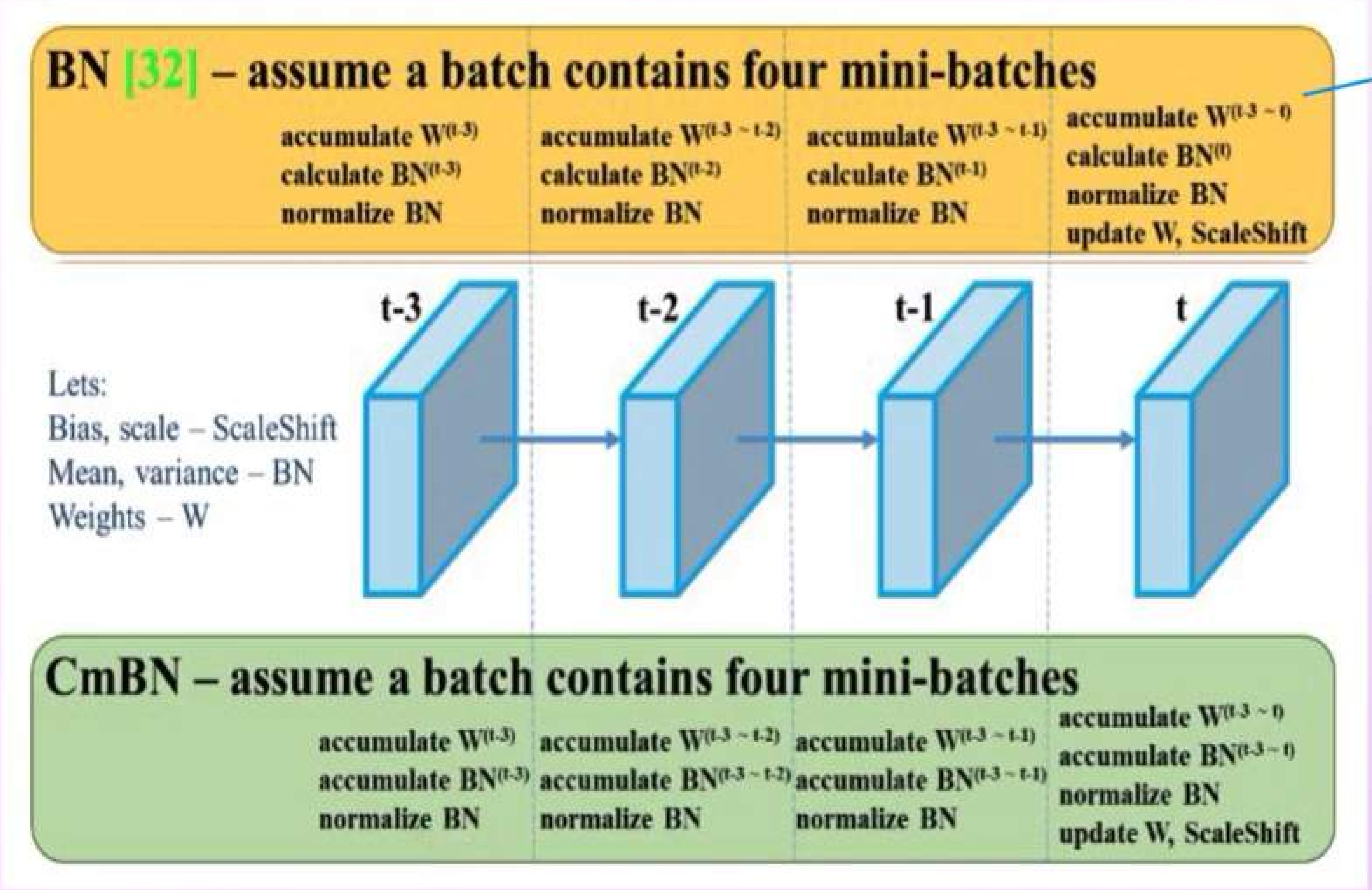
Gradient Accumulation

Cross Mini Batch Normalization



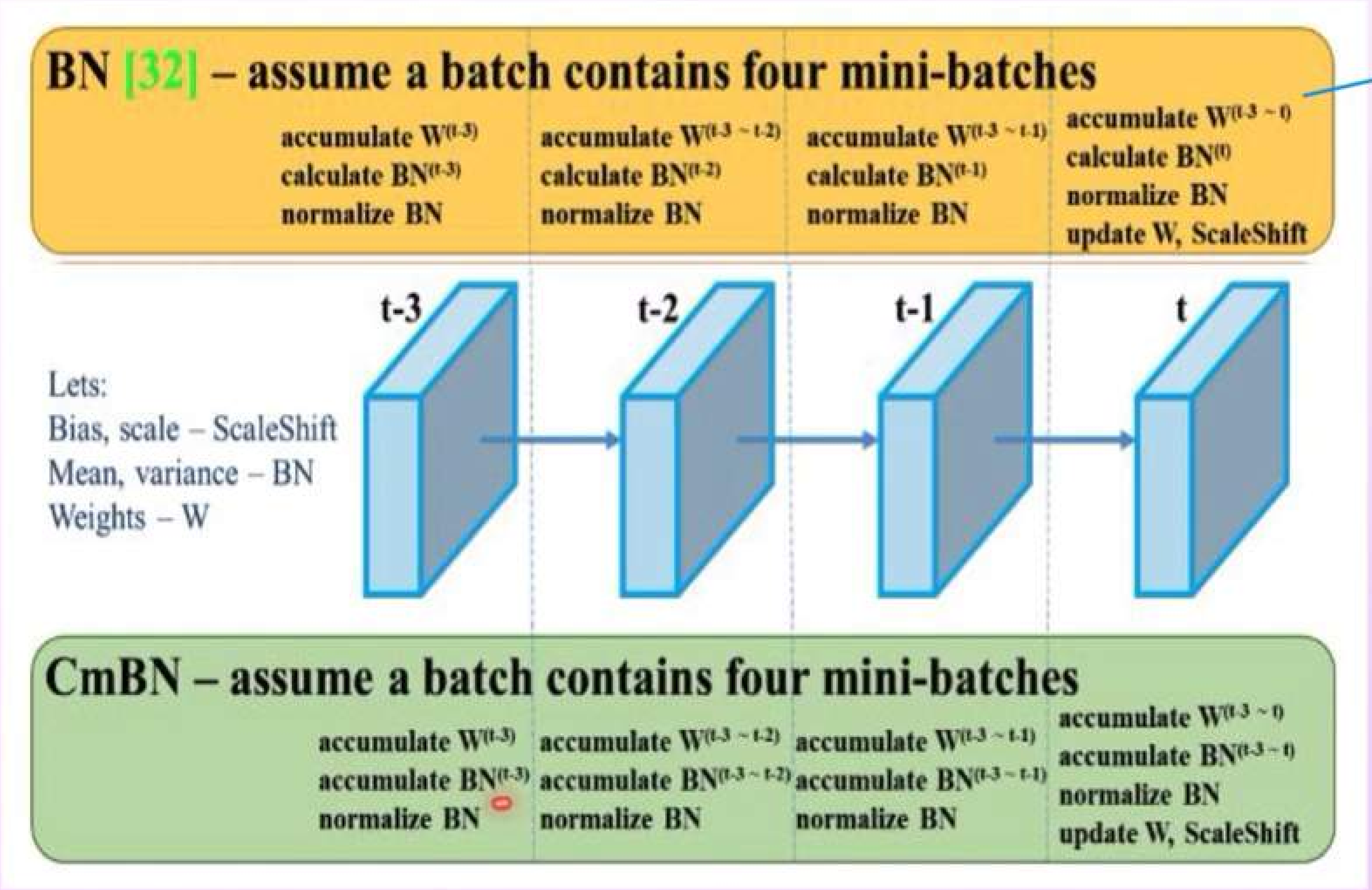
Gradient
Accumulation

Cross Mini Batch Normalization



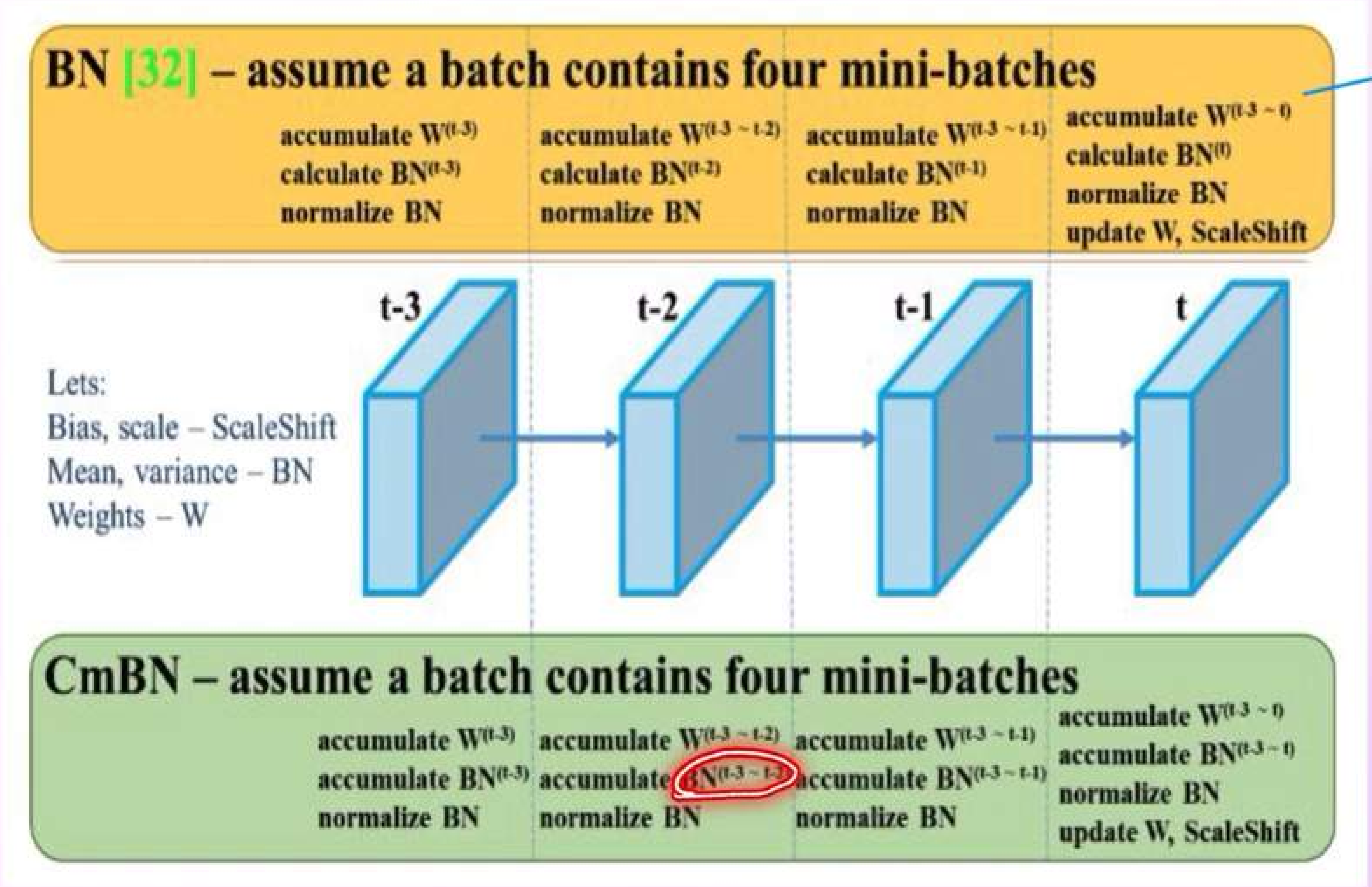
Gradient Accumulation

Cross Mini Batch Normalization



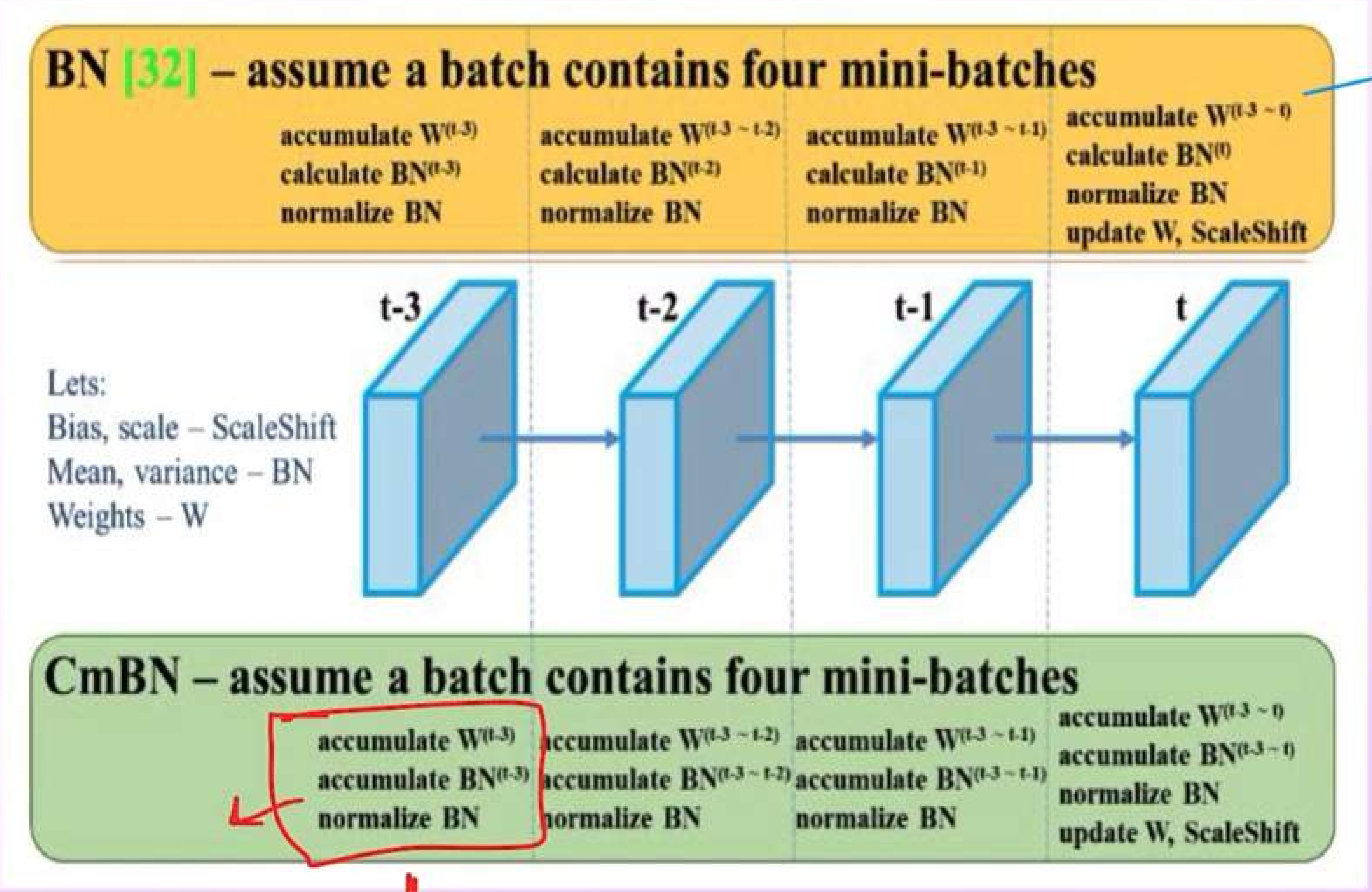
Gradient
Accumulation

Cross Mini Batch Normalization



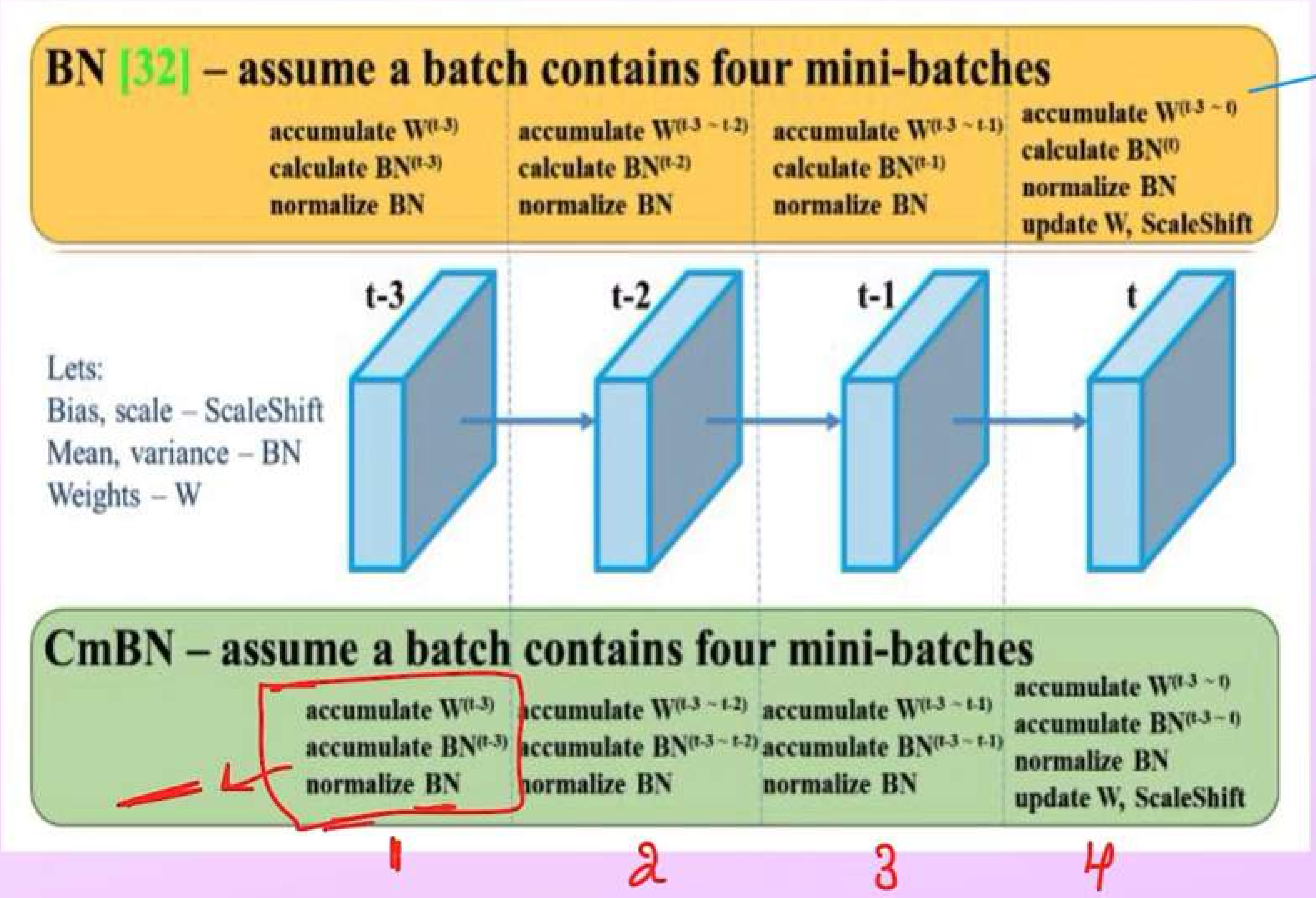
Gradient
Accumulation

Cross Mini Batch Normalization



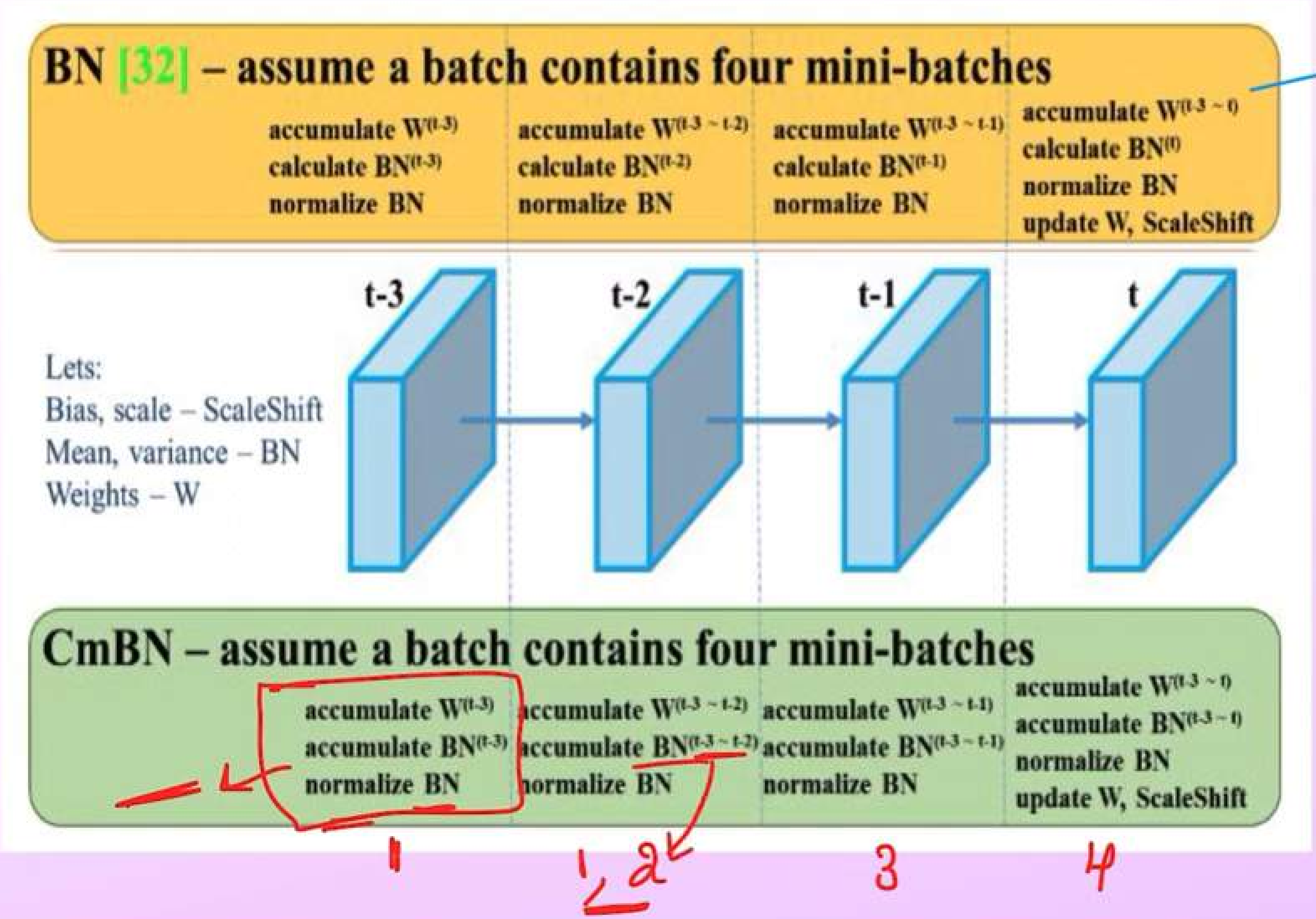
Gradient Accumulation

Cross Mini Batch Normalization



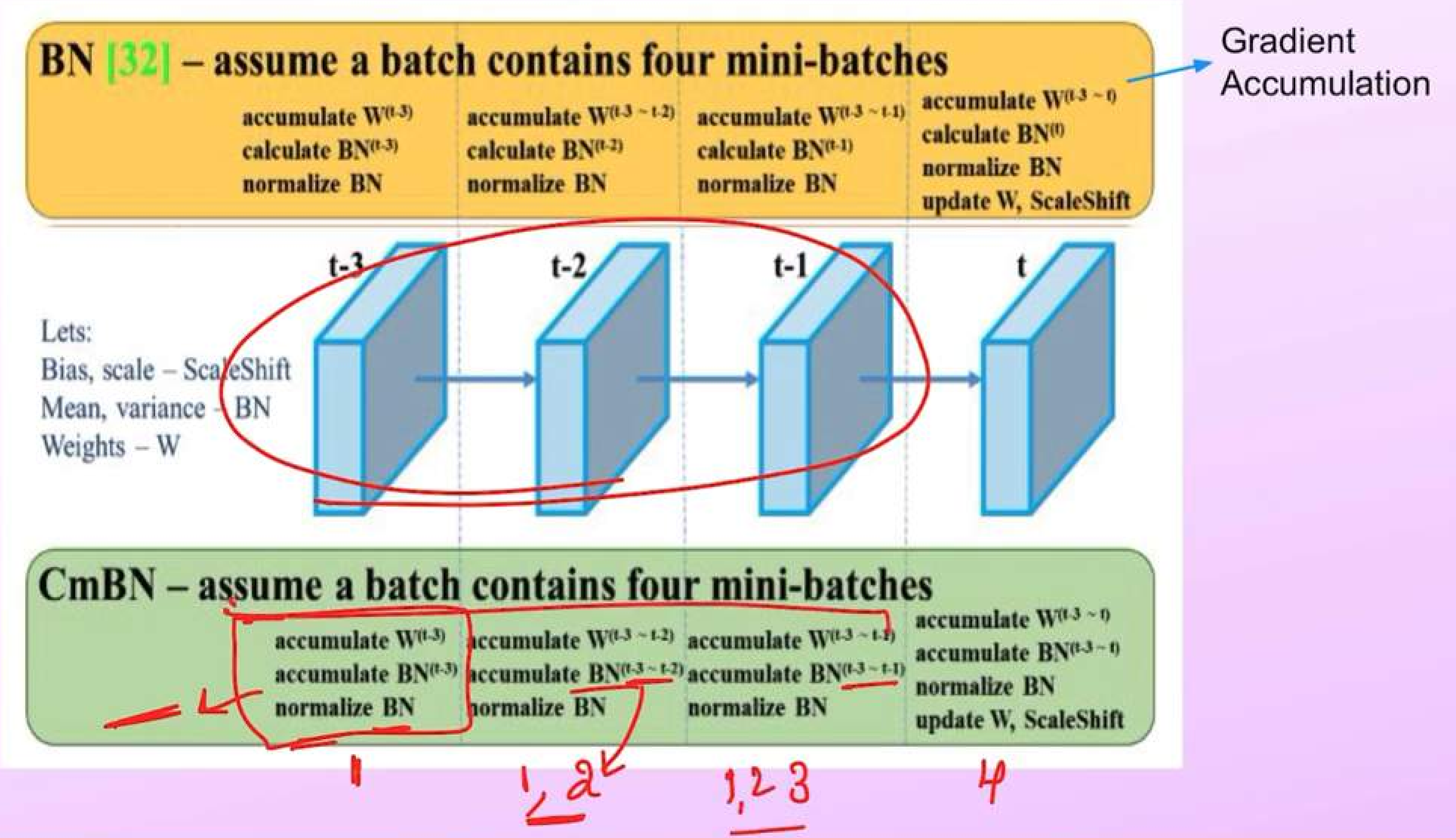
Gradient Accumulation

Cross Mini Batch Normalization

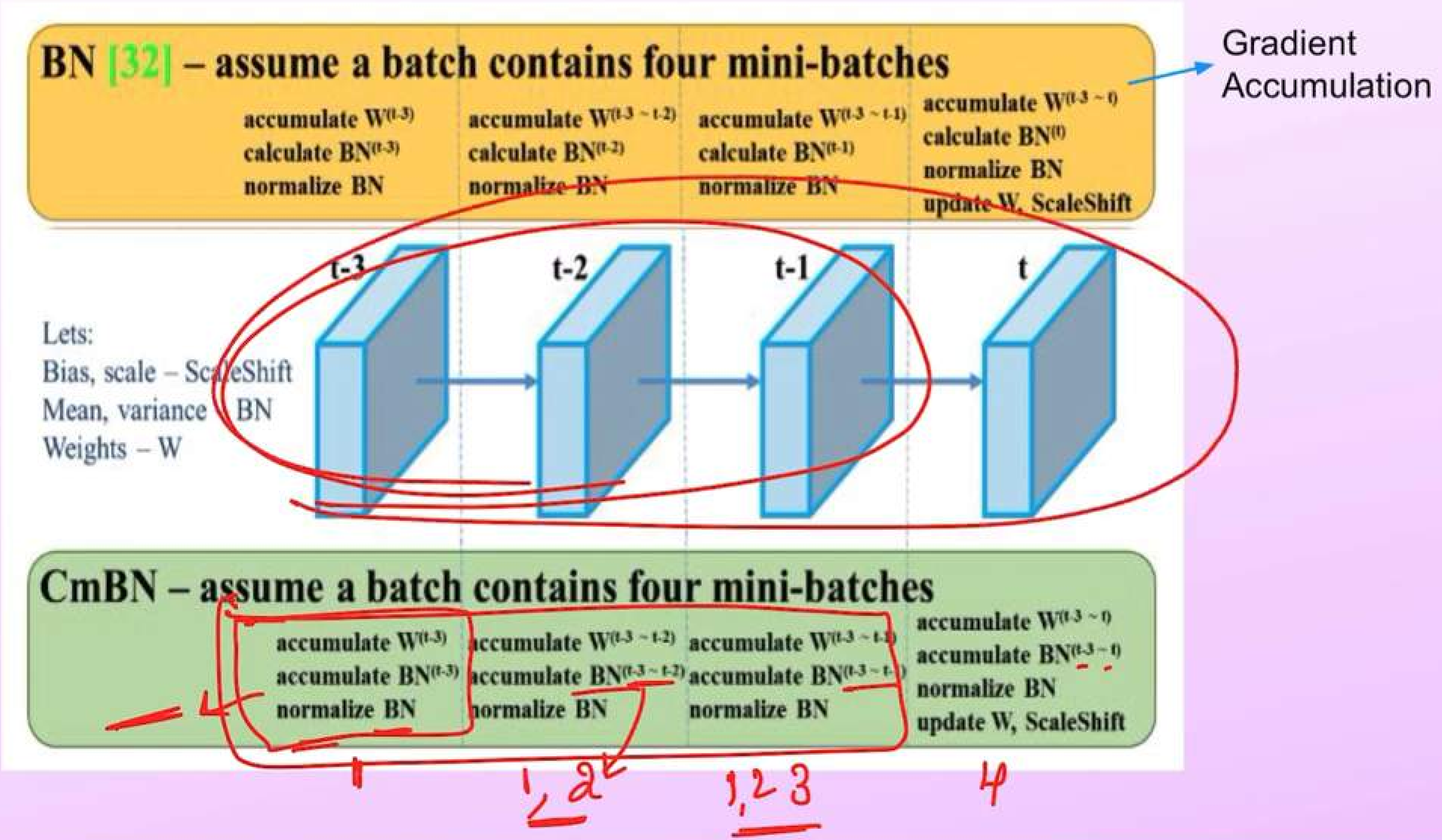


Gradient Accumulation

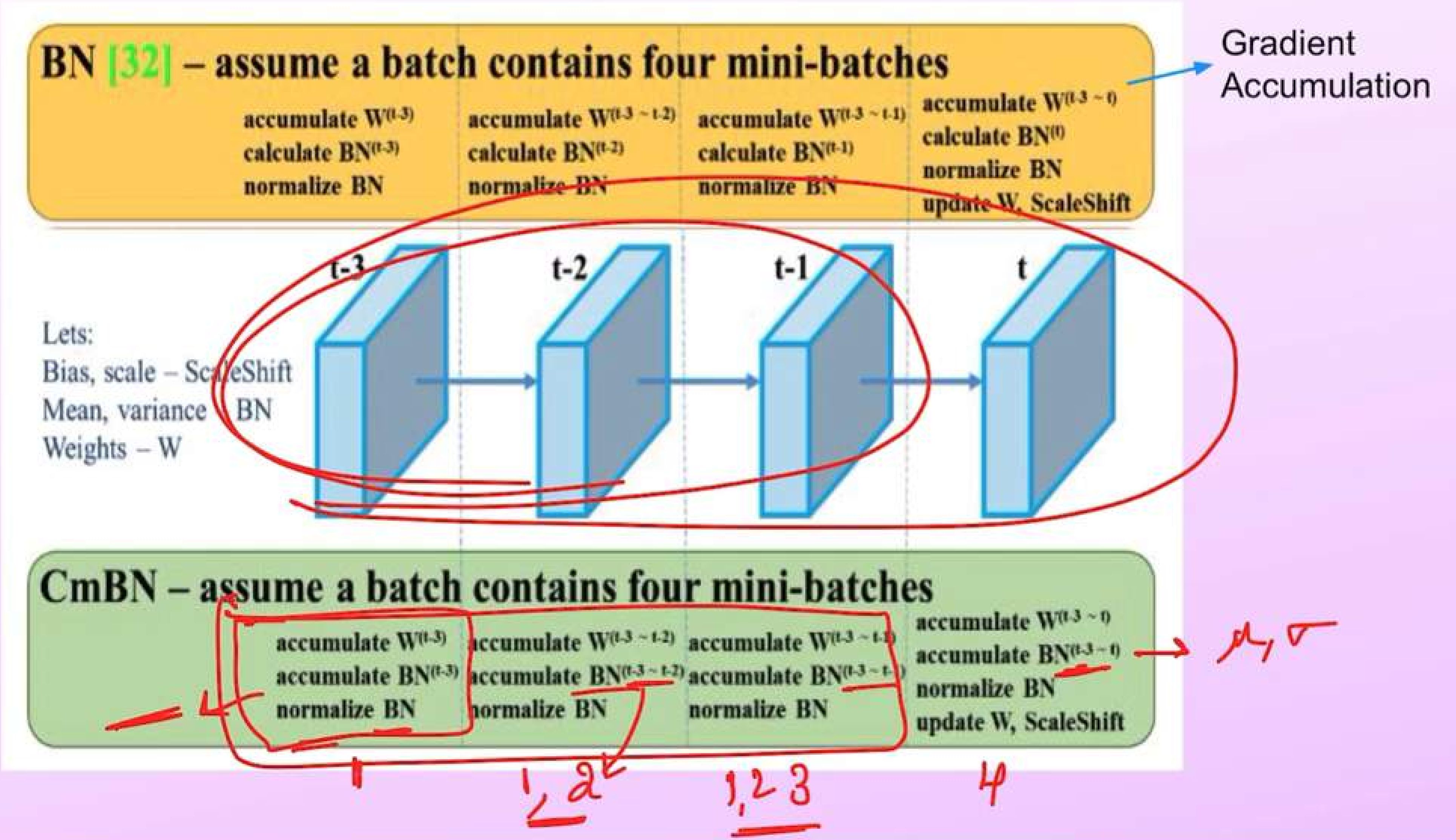
Cross Mini Batch Normalization



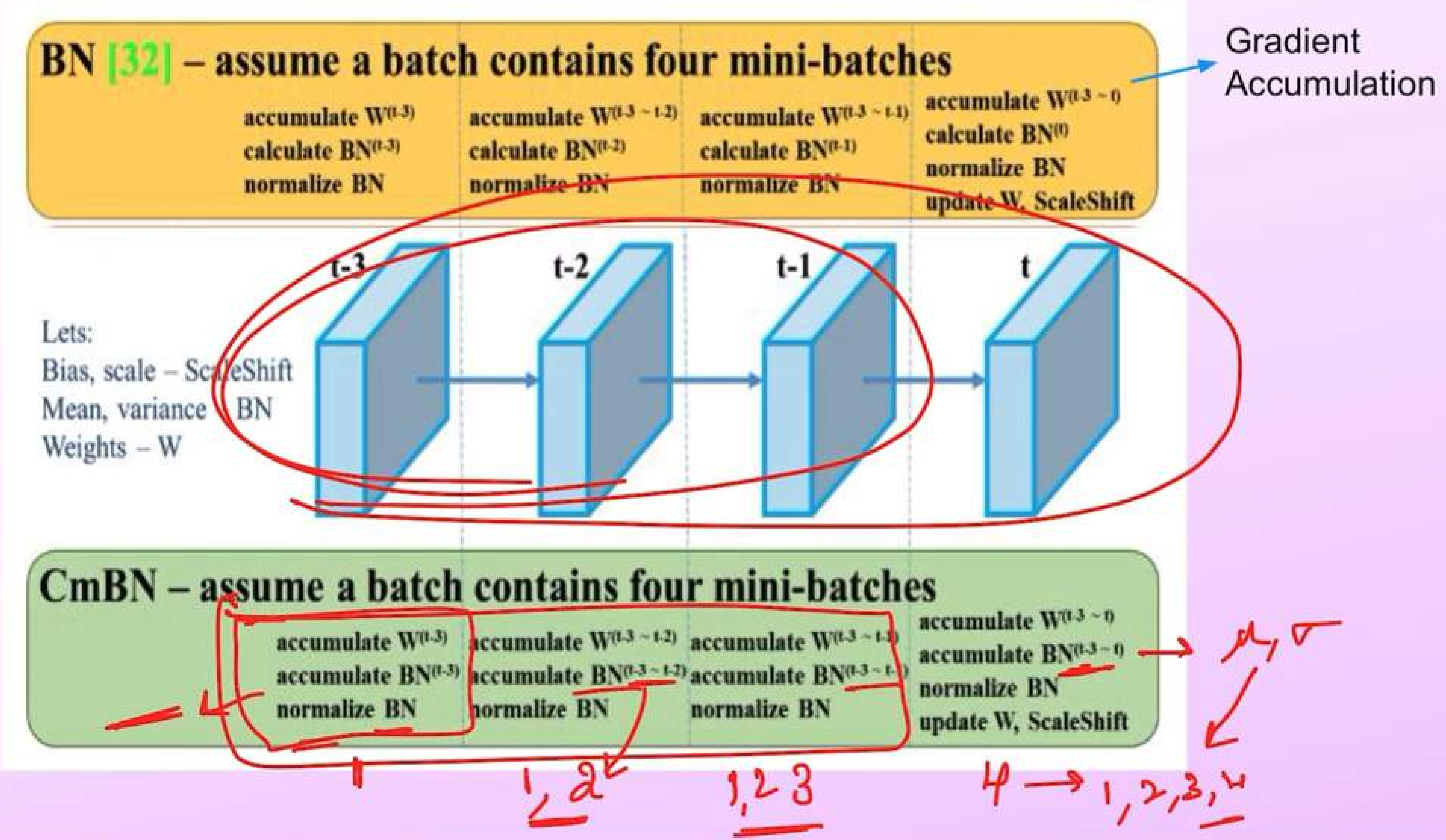
Cross Mini Batch Normalization



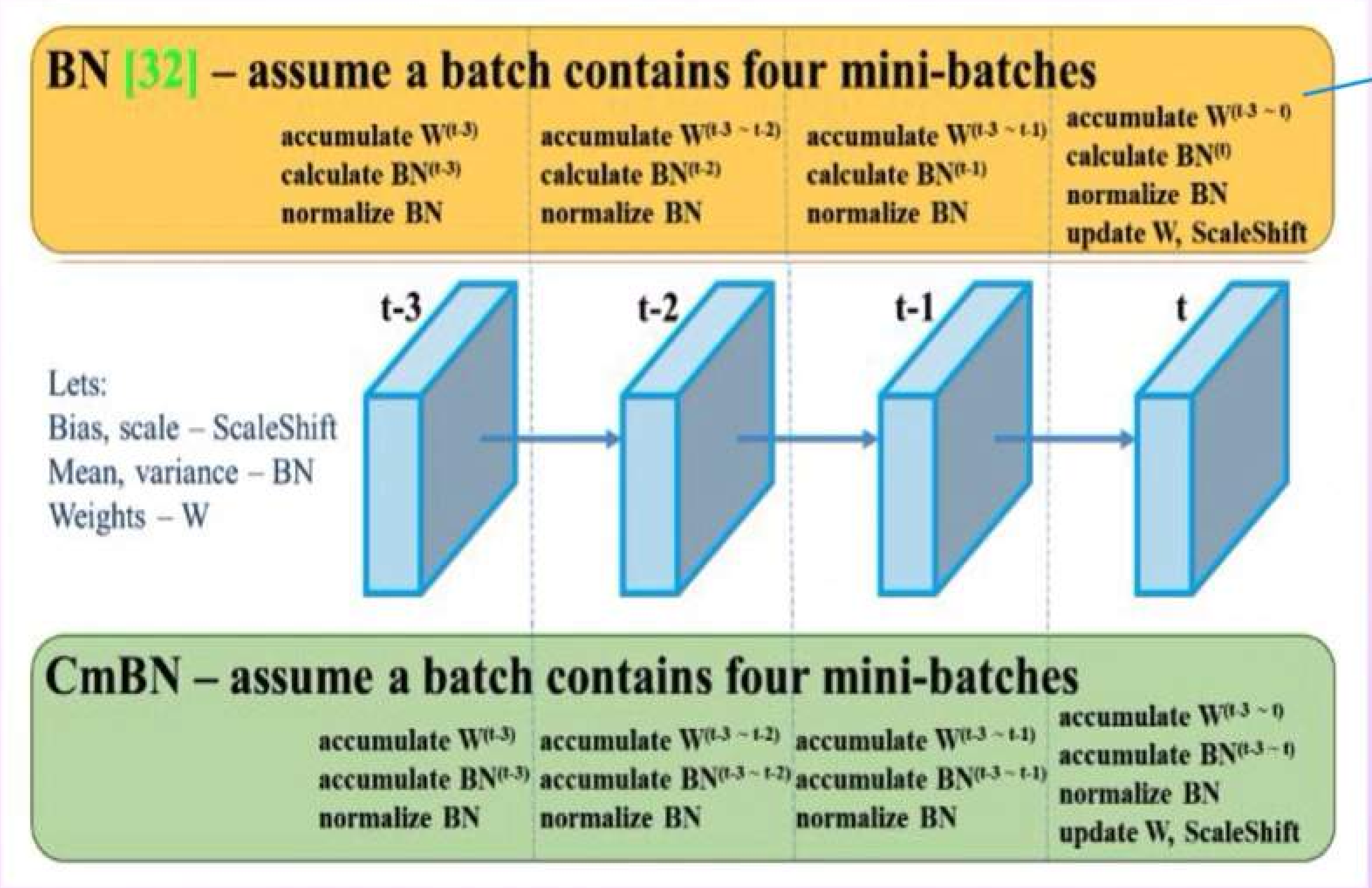
Cross Mini Batch Normalization



Cross Mini Batch Normalization

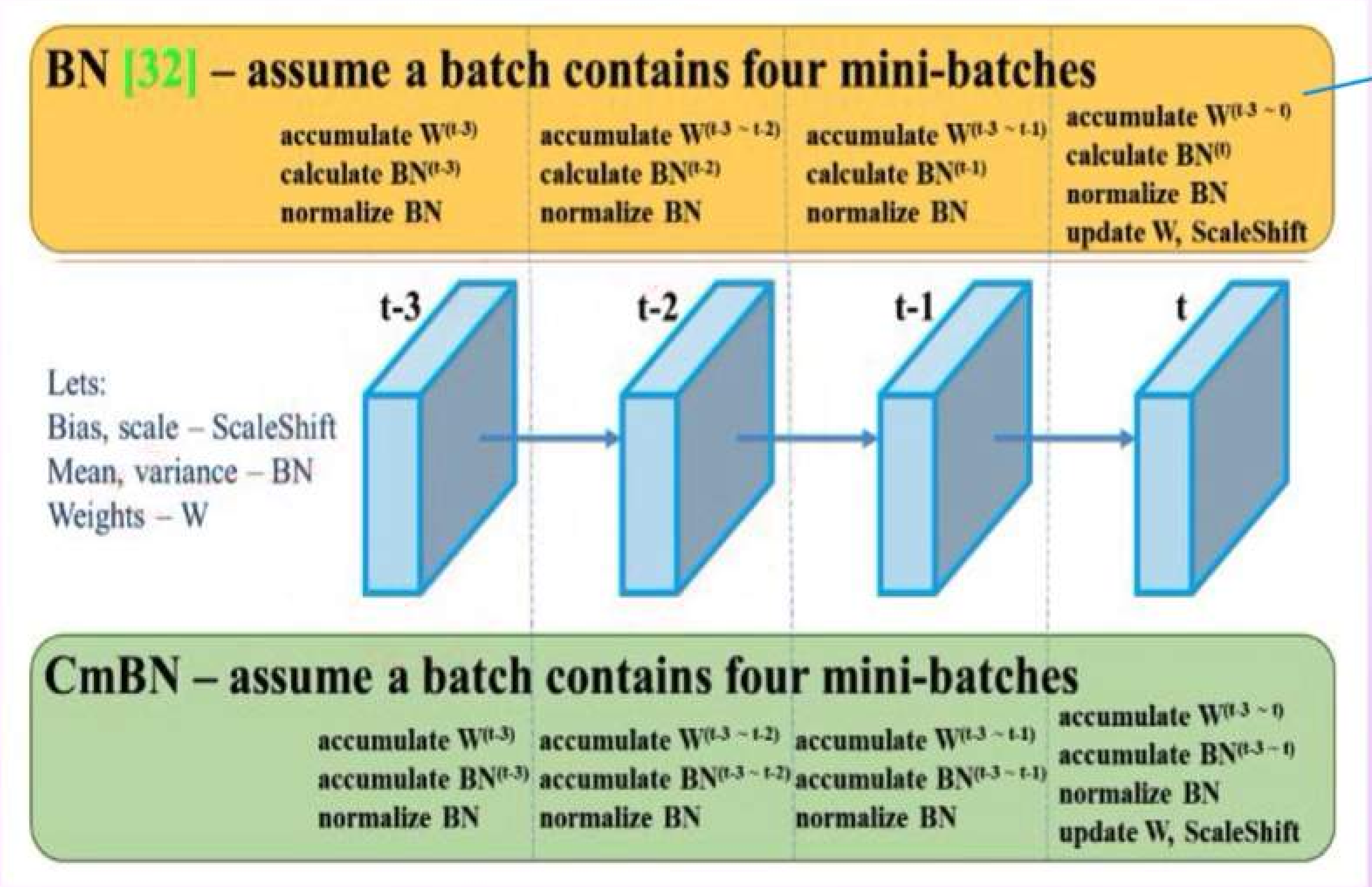


Cross Mini Batch Normalization



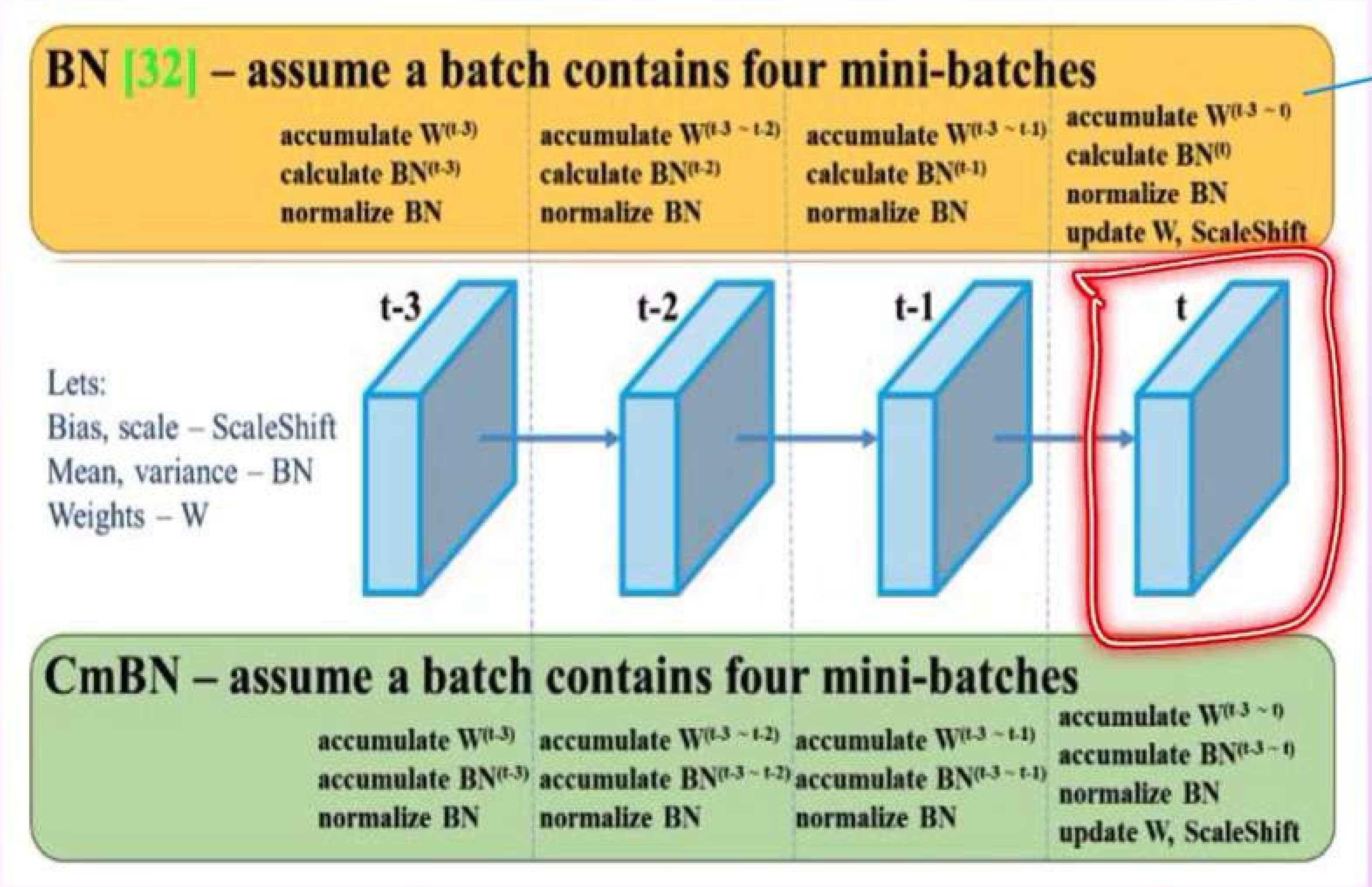
Gradient Accumulation

Cross Mini Batch Normalization



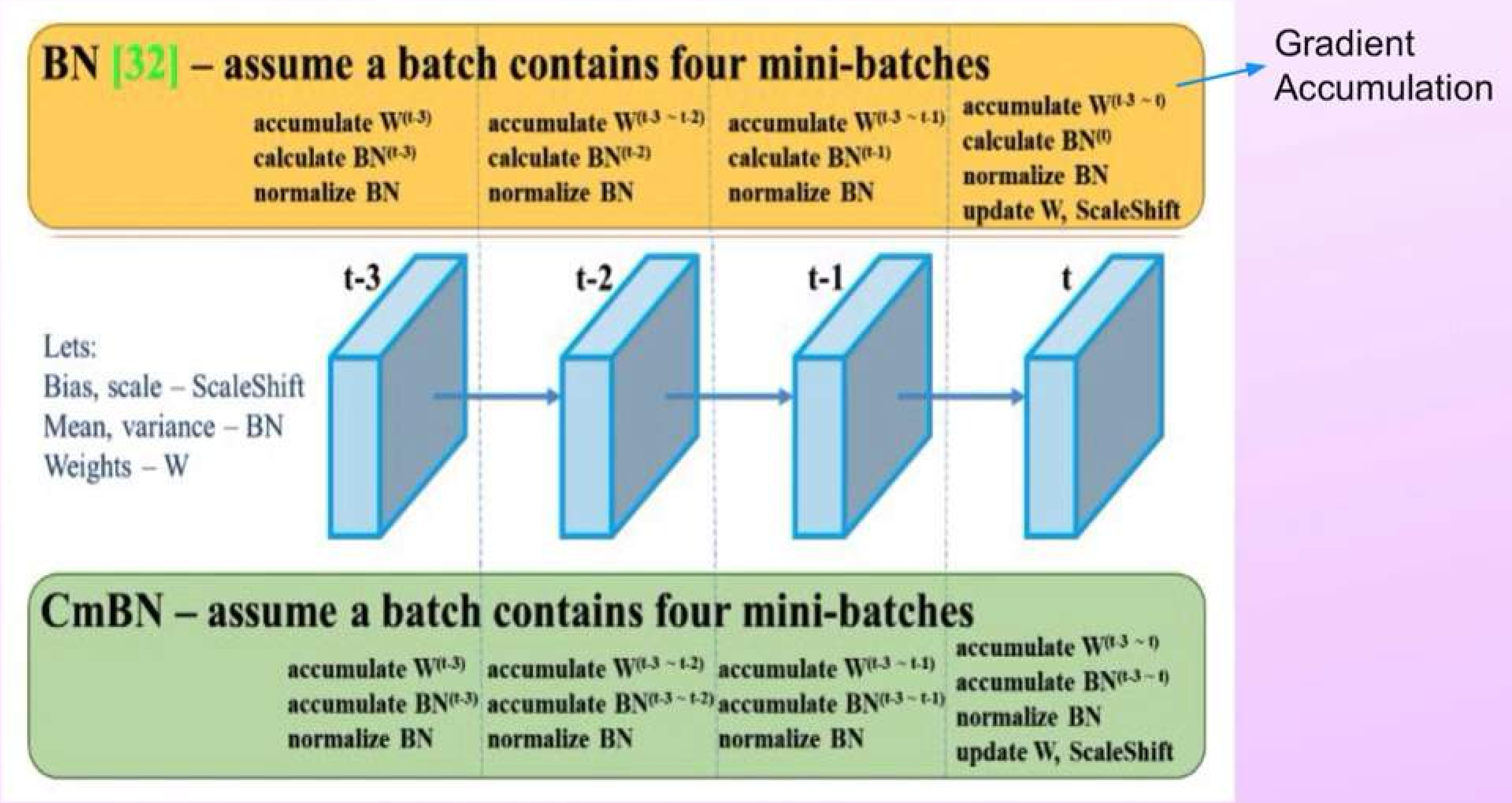
Gradient Accumulation

Cross Mini Batch Normalization



Gradient Accumulation

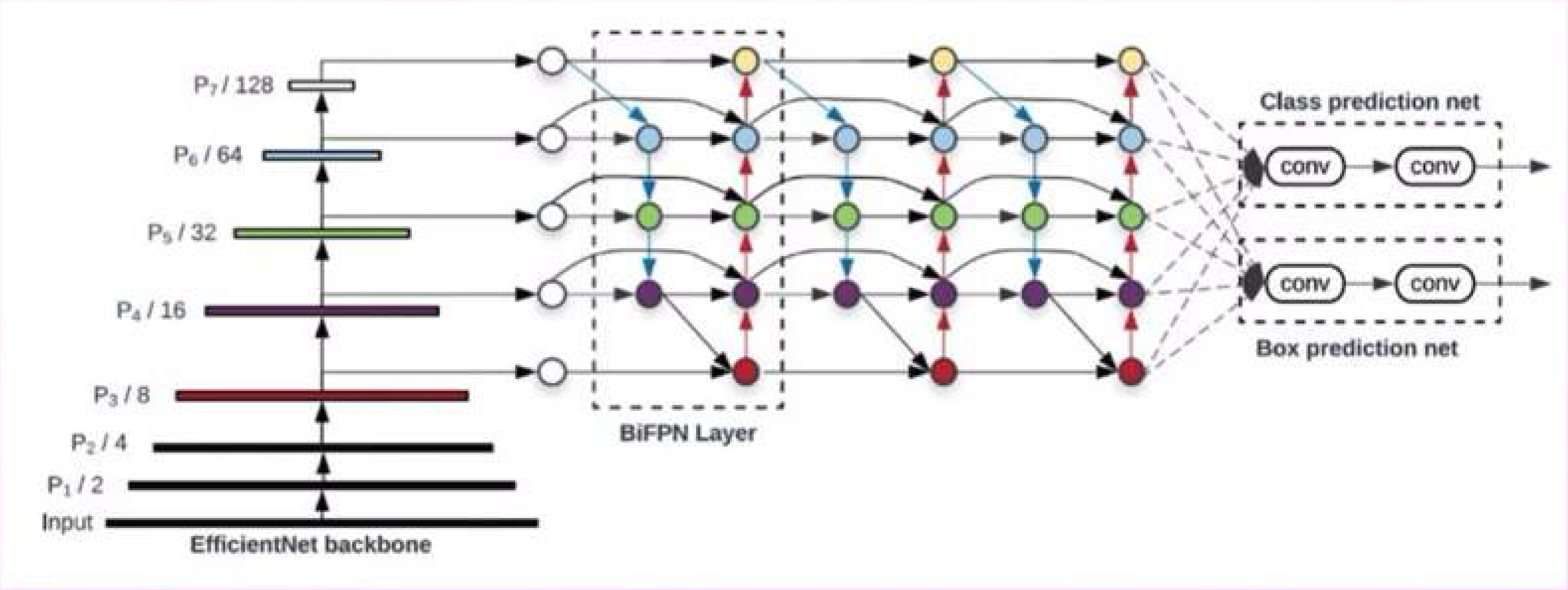
Cross Mini Batch Normalization



Multi-Input Weighted Residual Connections

- Idea from EfficientDet Paper
- Stage: Neck

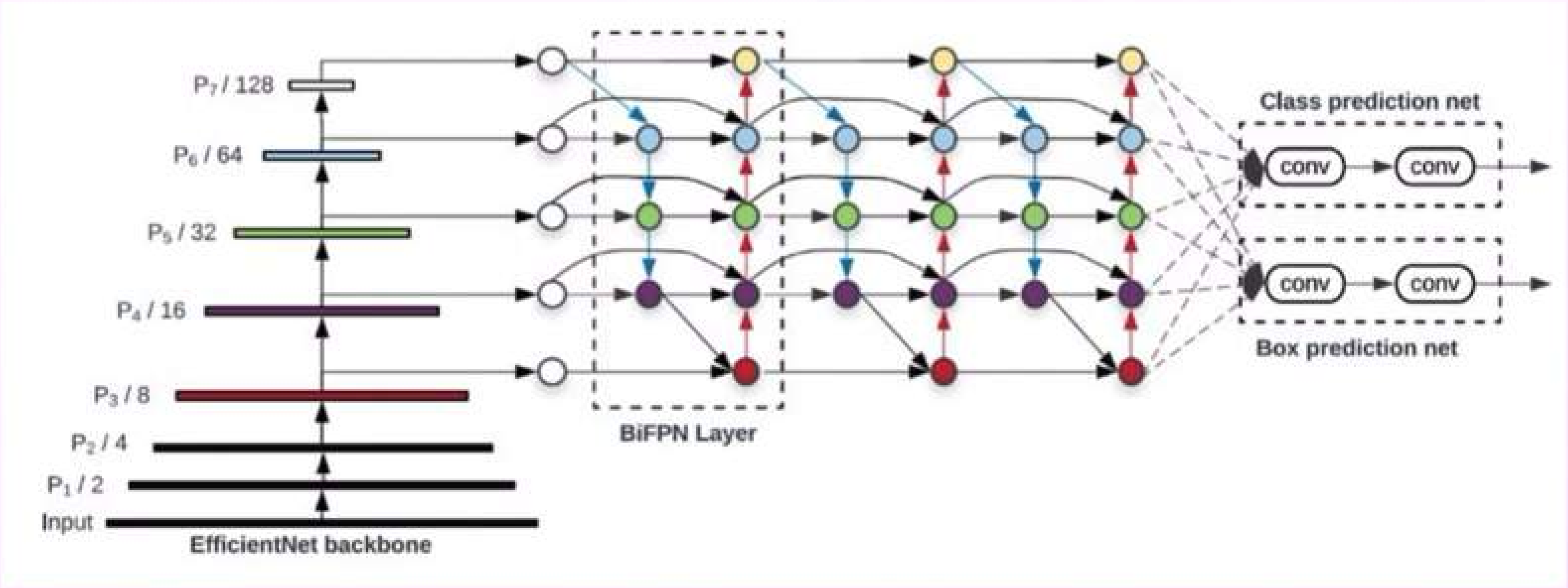
- EfficientDet —>



Multi-Input Weighted Residual Connections

- Idea from EfficientDet Paper
- Stage: Neck

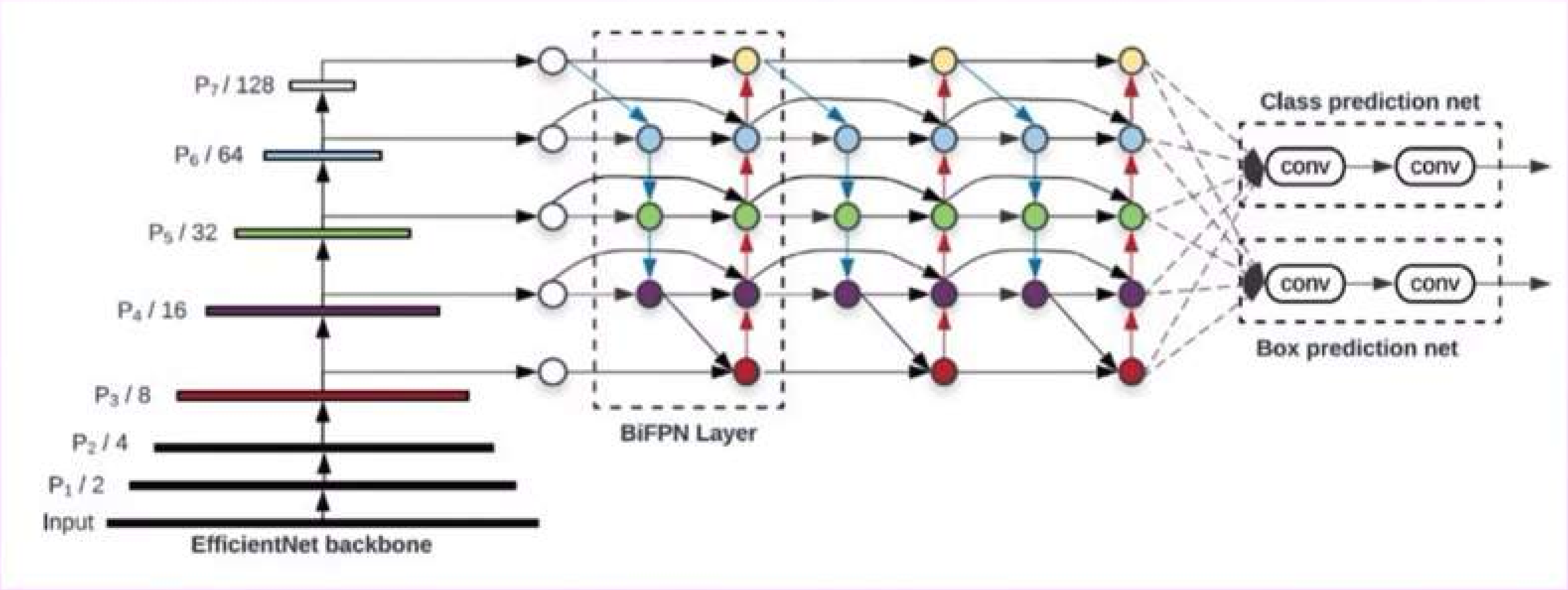
- EfficientDet —>



Multi-Input Weighted Residual Connections

- Idea from EfficientDet Paper
- Stage: Neck

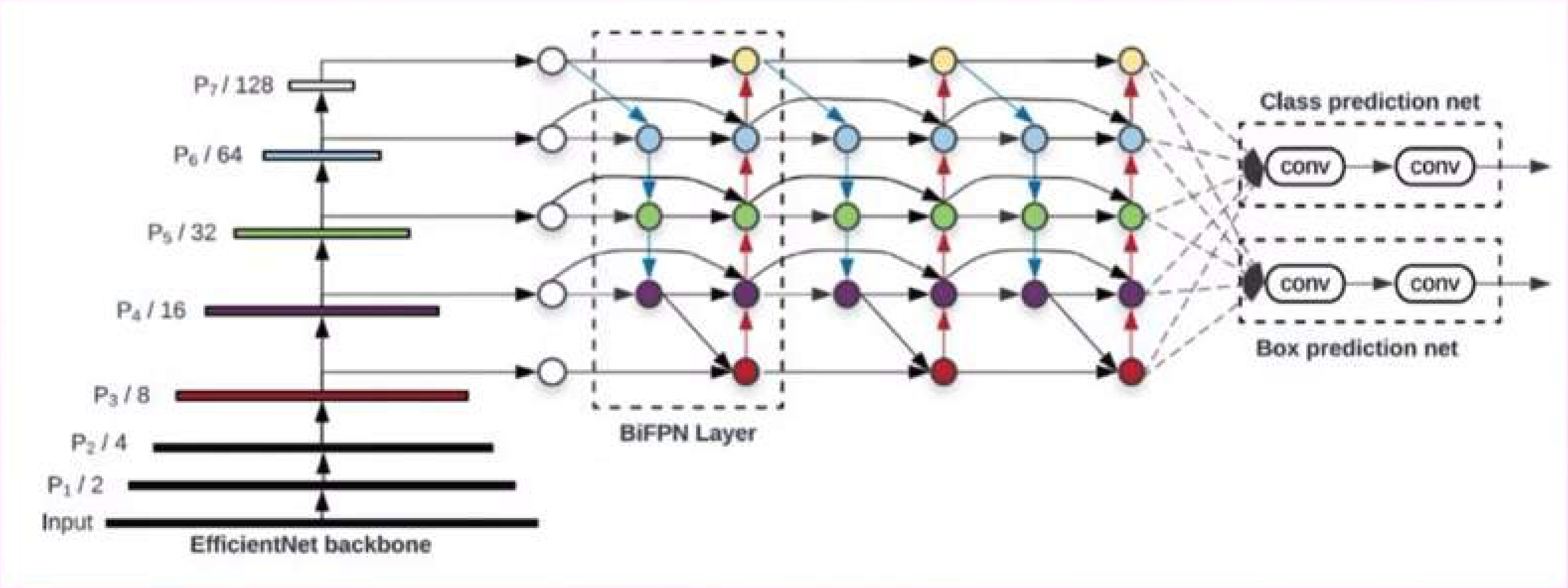
- EfficientDet —>



Multi-Input Weighted Residual Connections

- Idea from EfficientDet Paper
- Stage: Neck

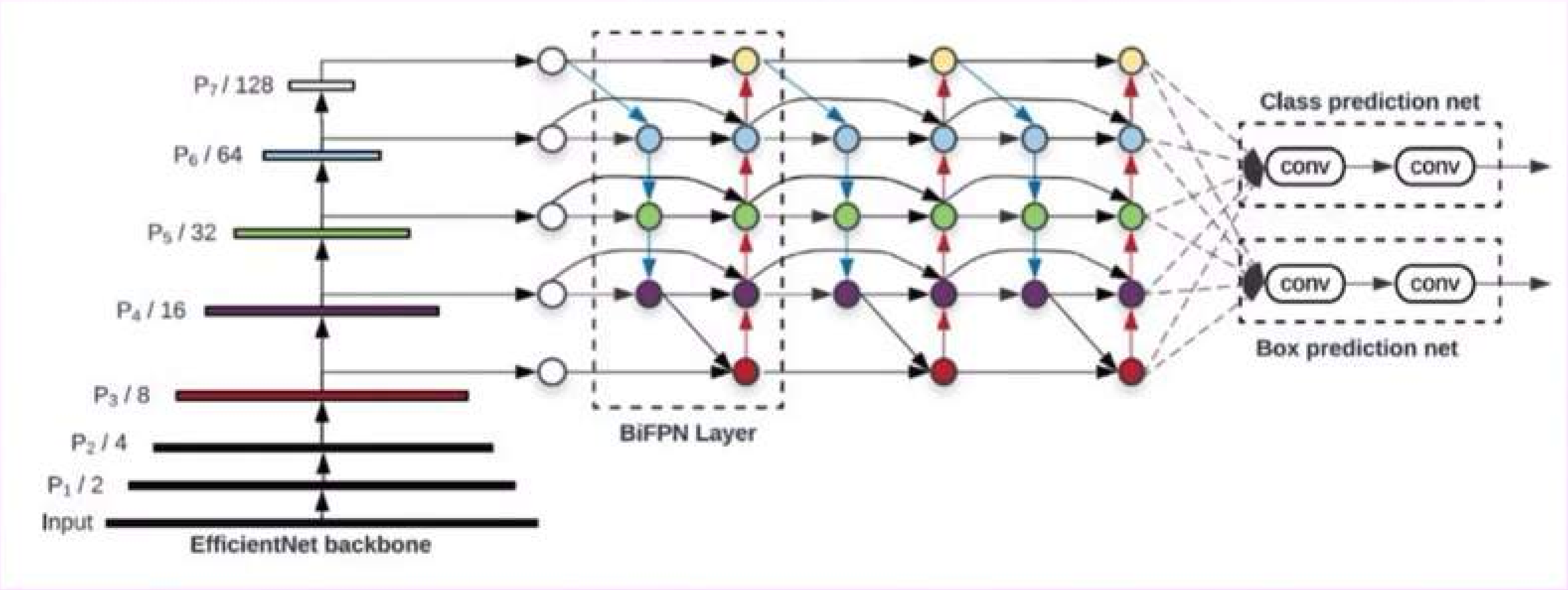
- EfficientDet —>



Multi-Input Weighted Residual Connections

- Idea from EfficientDet Paper
- Stage: Neck

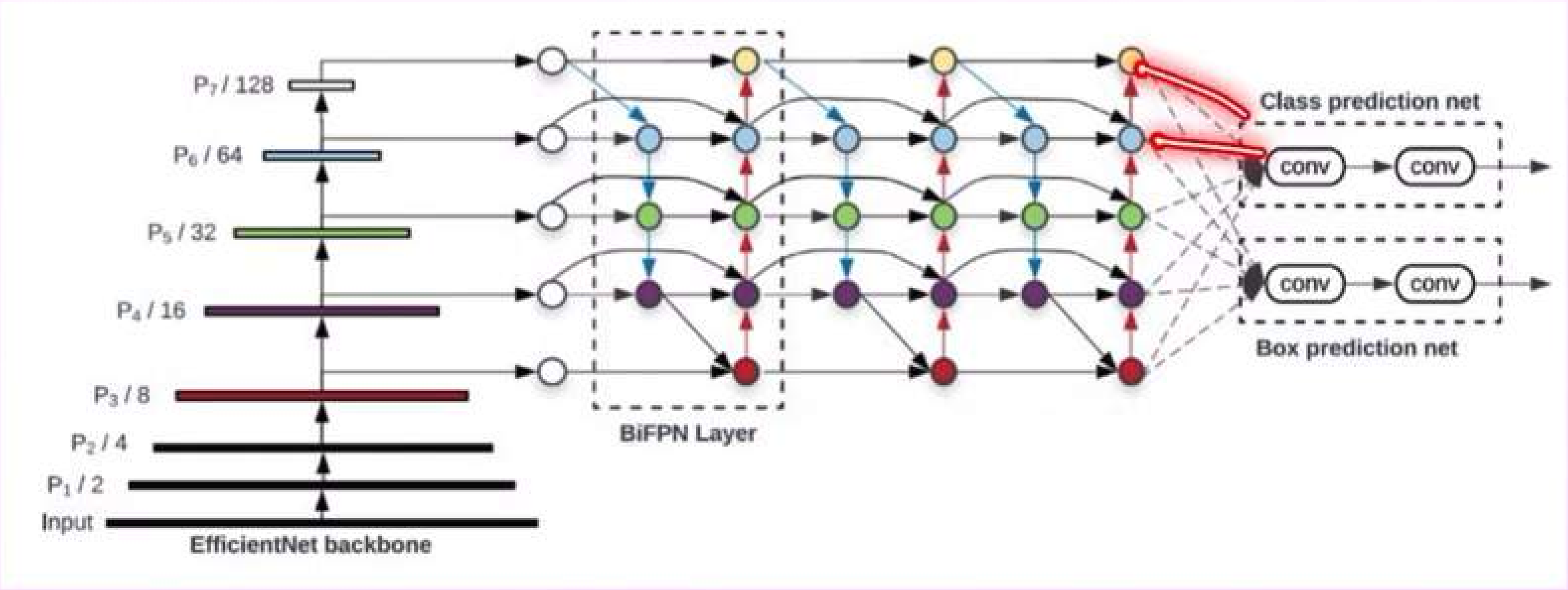
- EfficientDet —>



Multi-Input Weighted Residual Connections

- Idea from EfficientDet Paper
- Stage: Neck

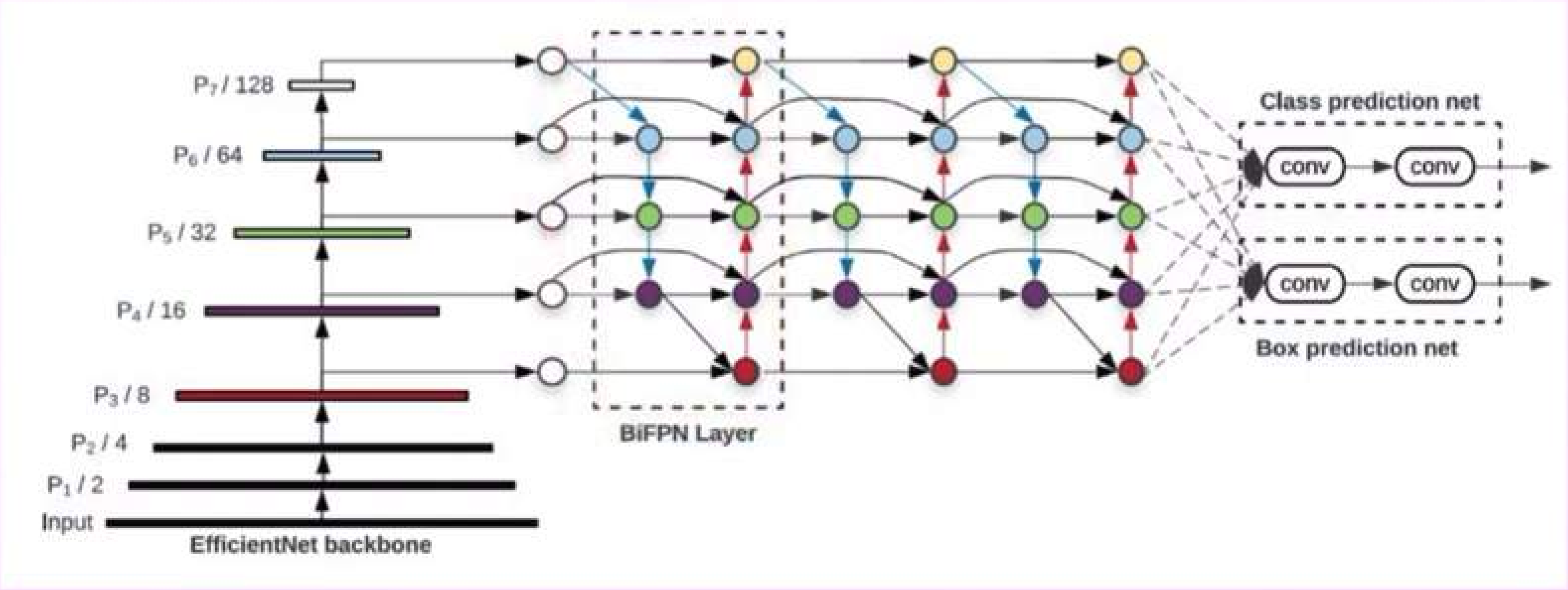
- EfficientDet —>



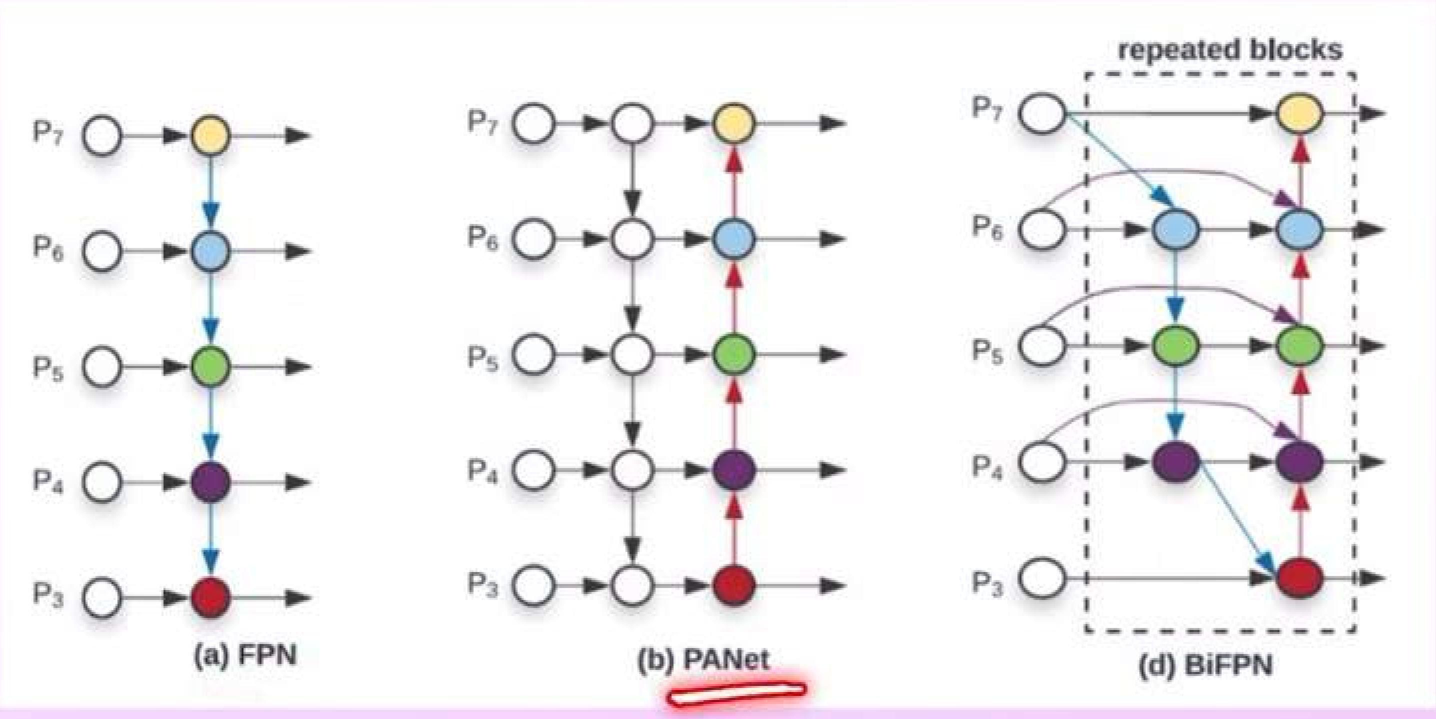
Multi-Input Weighted Residual Connections

- Idea from EfficientDet Paper
- Stage: Neck

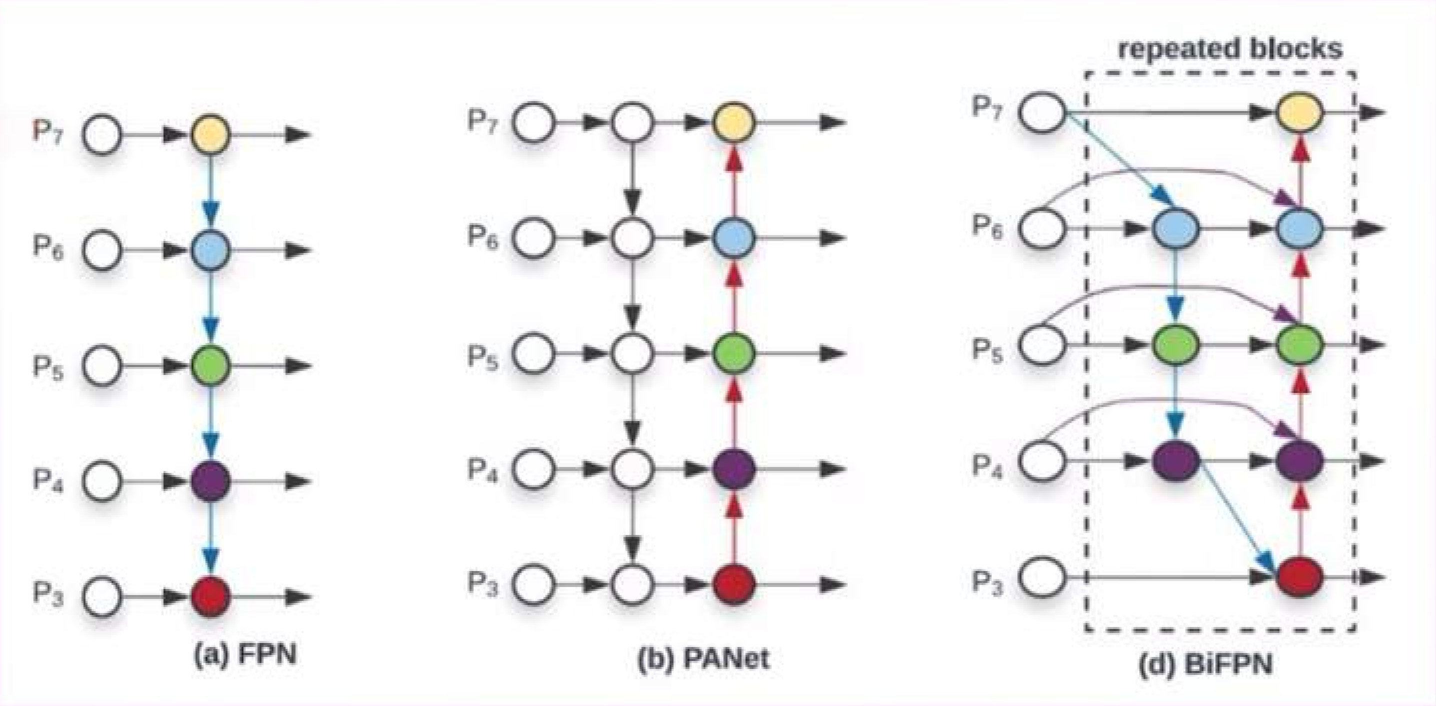
- EfficientDet —>



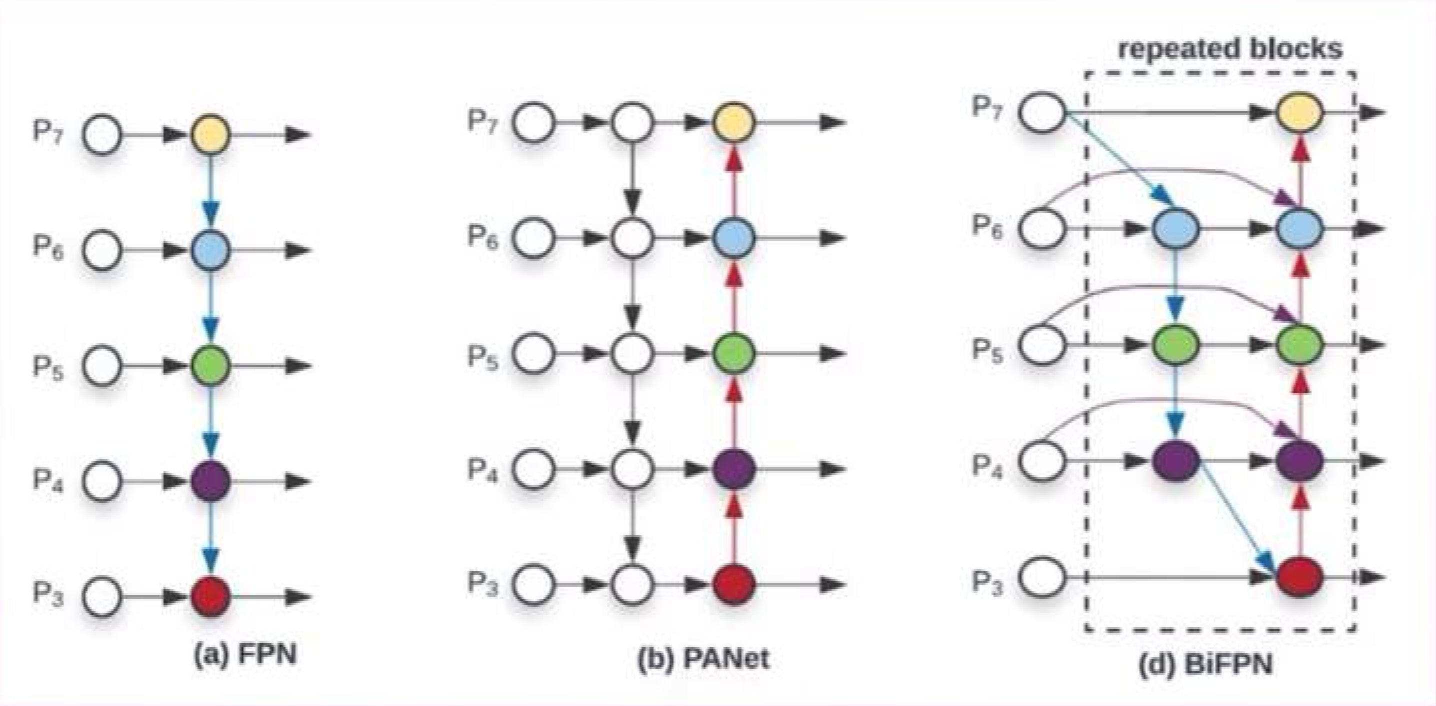
Multi-Input Weighted Residual Connections



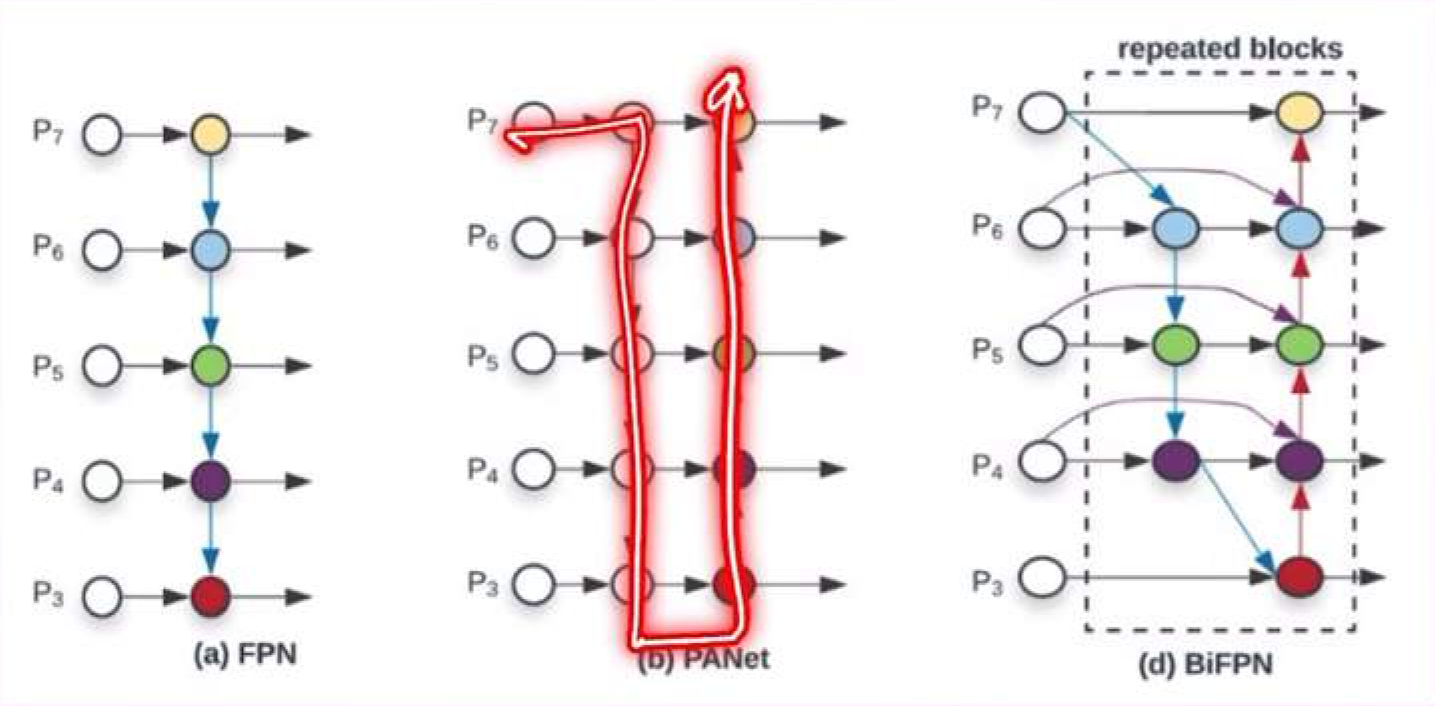
Multi-Input Weighted Residual Connections



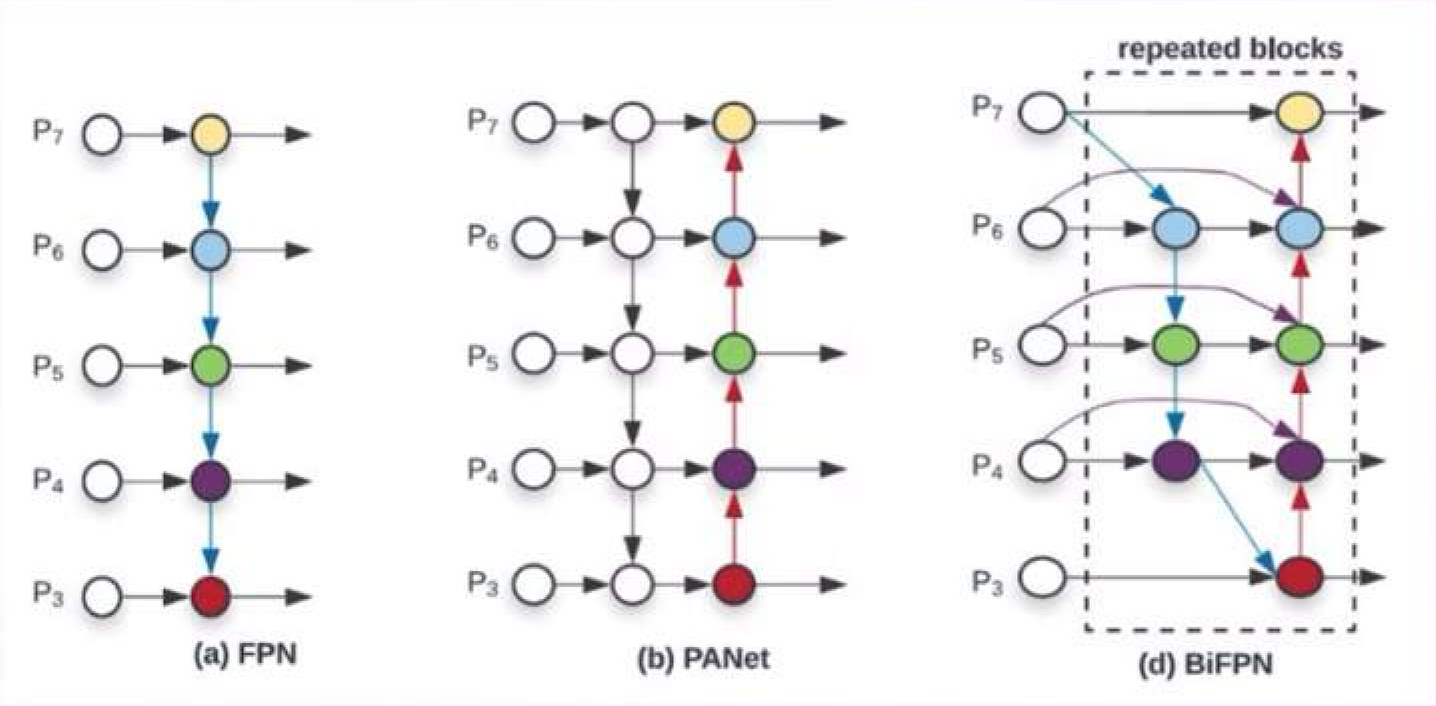
Multi-Input Weighted Residual Connections



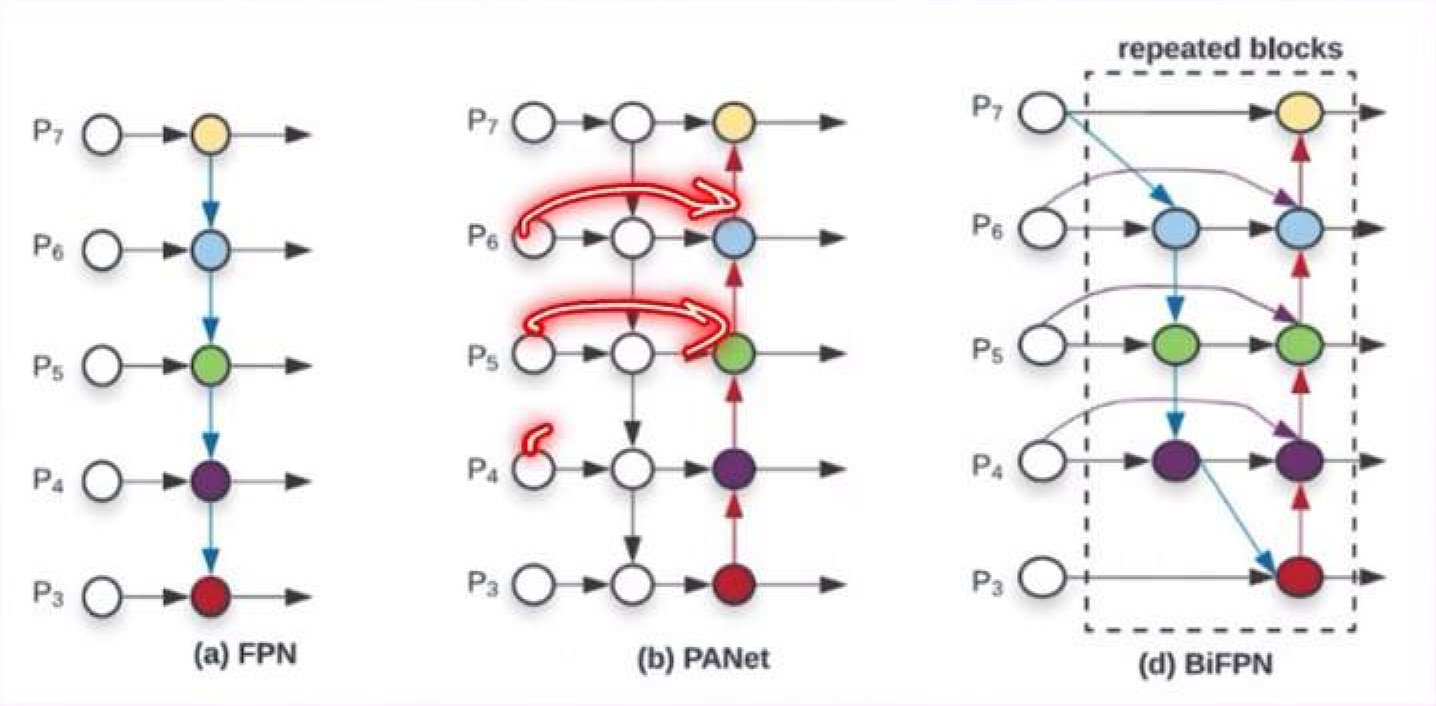
Multi-Input Weighted Residual Connections



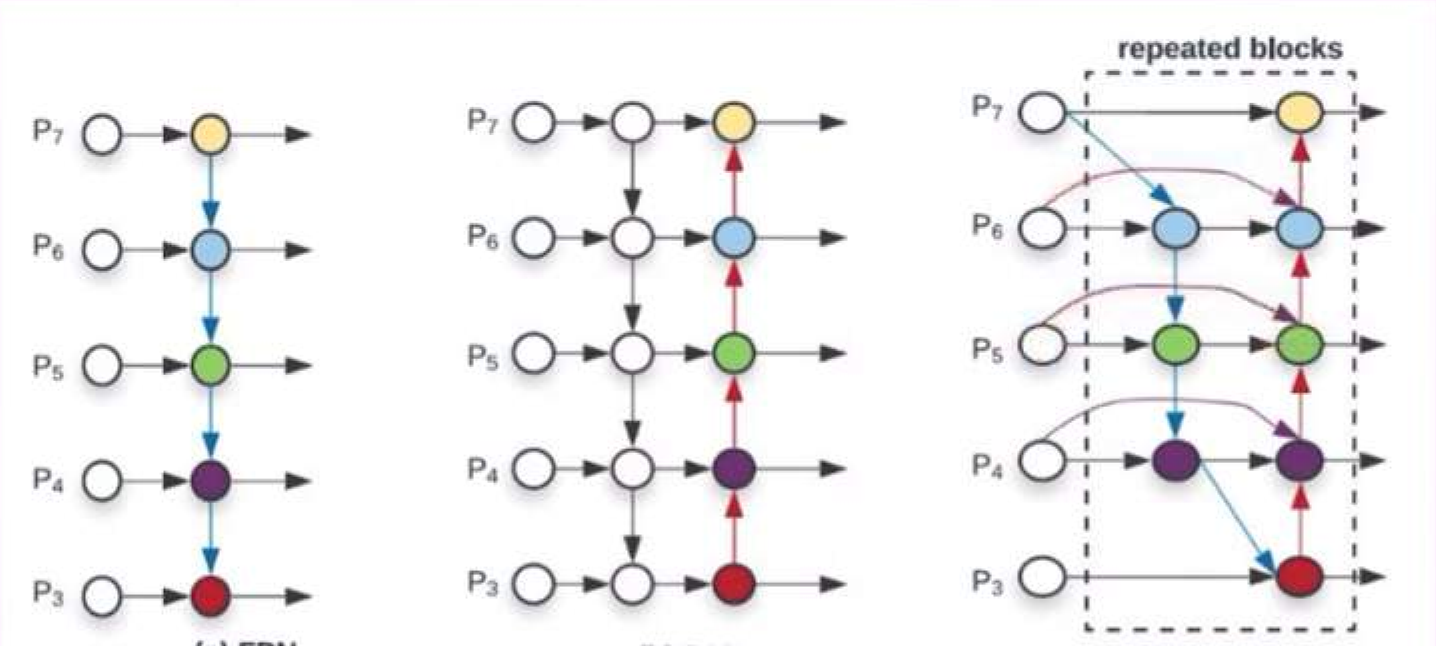
Multi-Input Weighted Residual Connections



Multi-Input Weighted Residual Connections

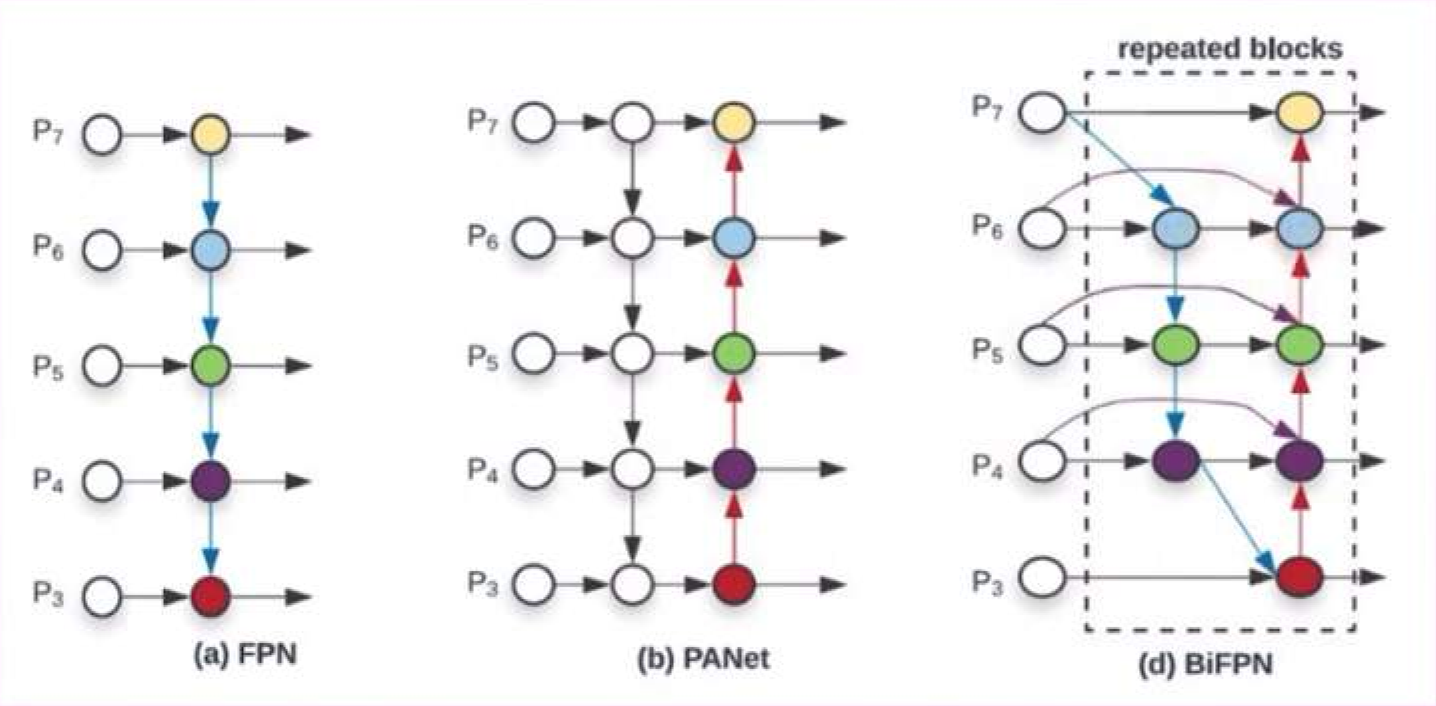


Multi-Input Weighted Residual Connections

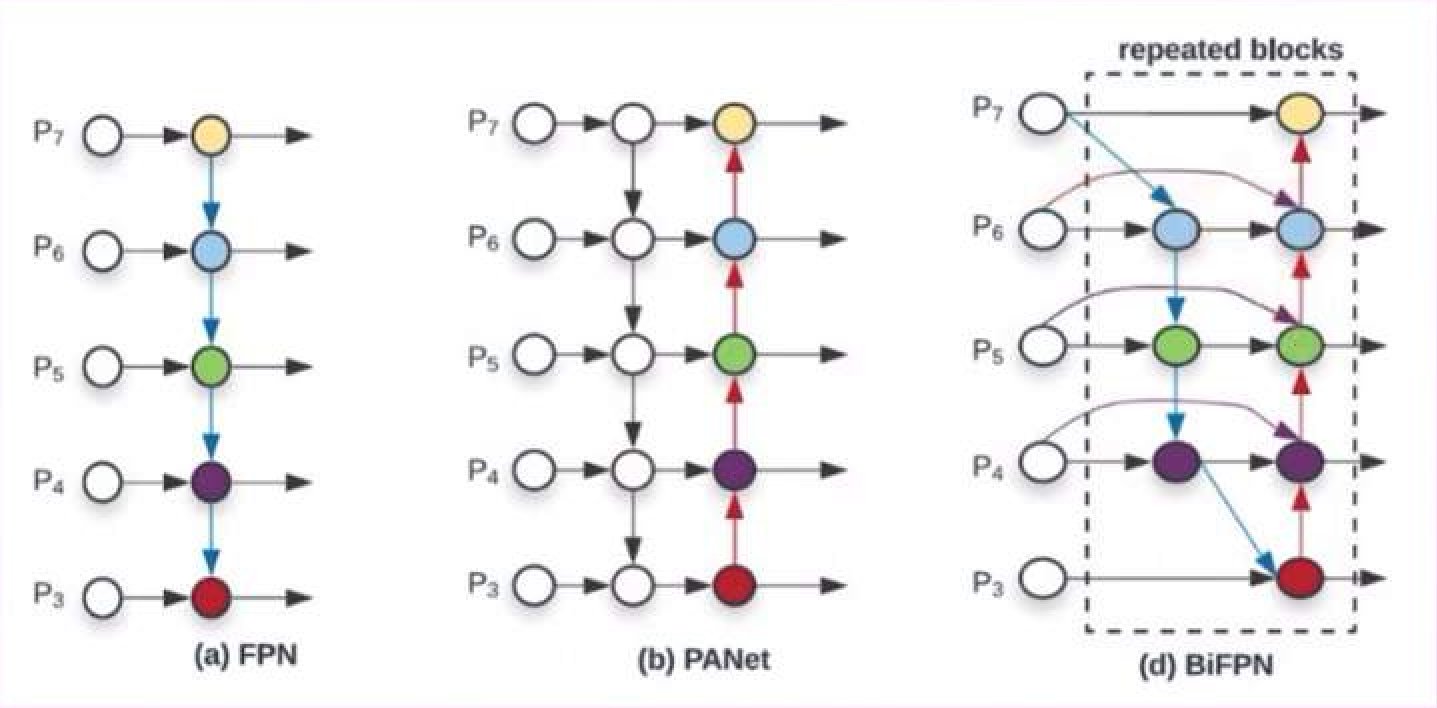


(a) FPN (b) PANet (d) BiFPN

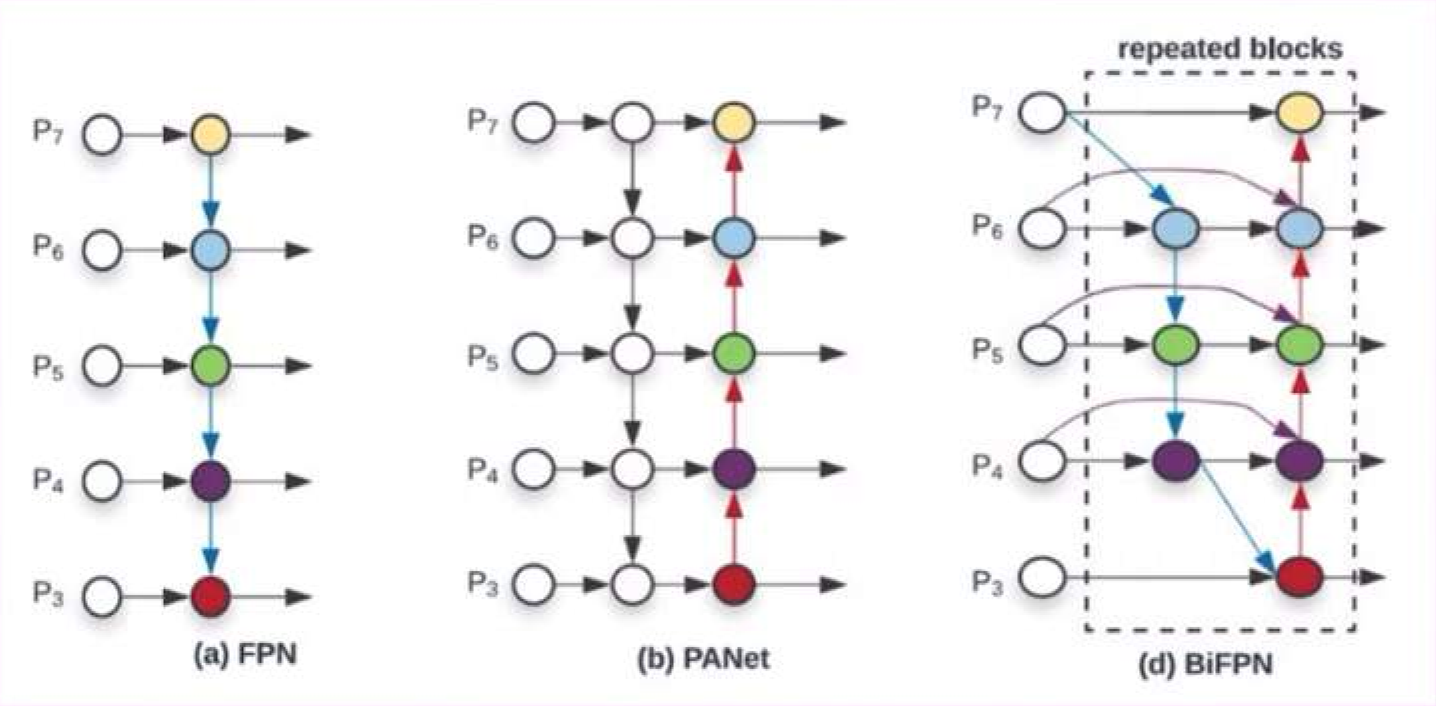
Multi-Input Weighted Residual Connections



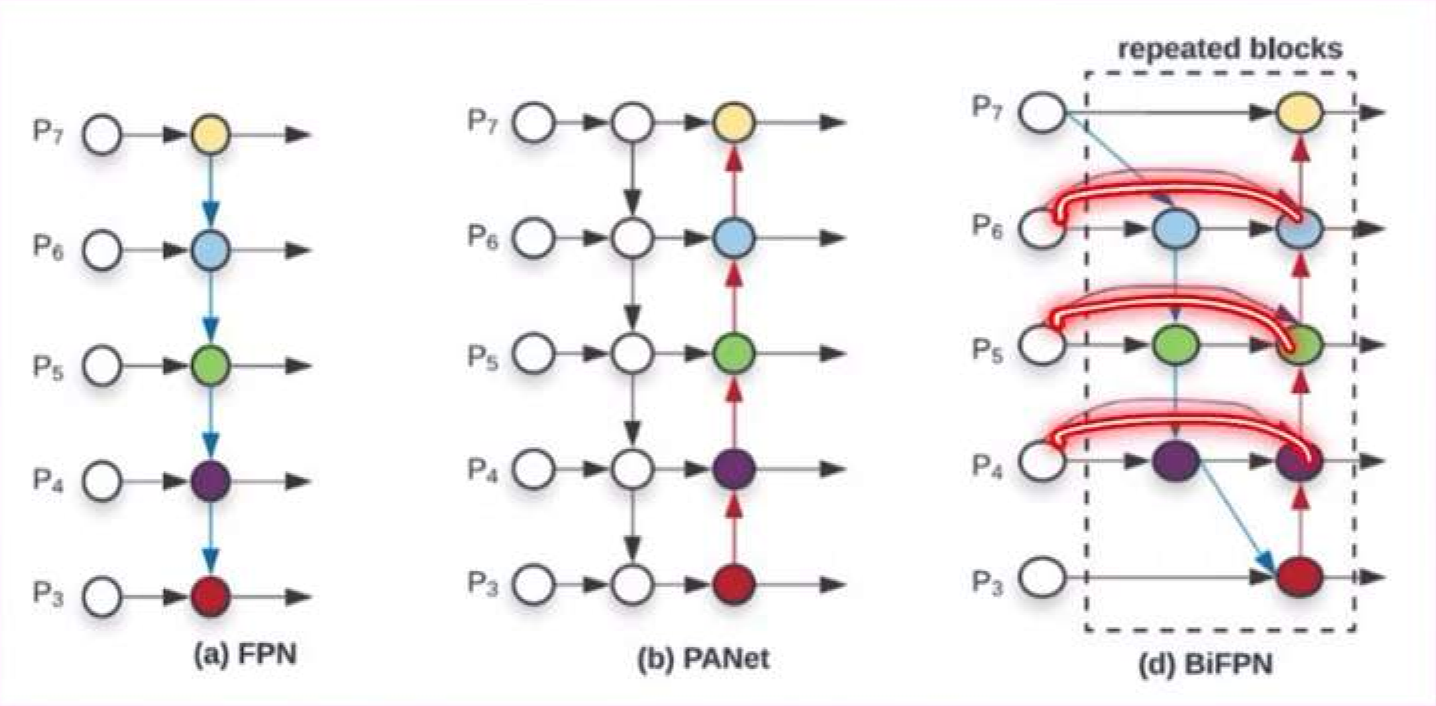
Multi-Input Weighted Residual Connections



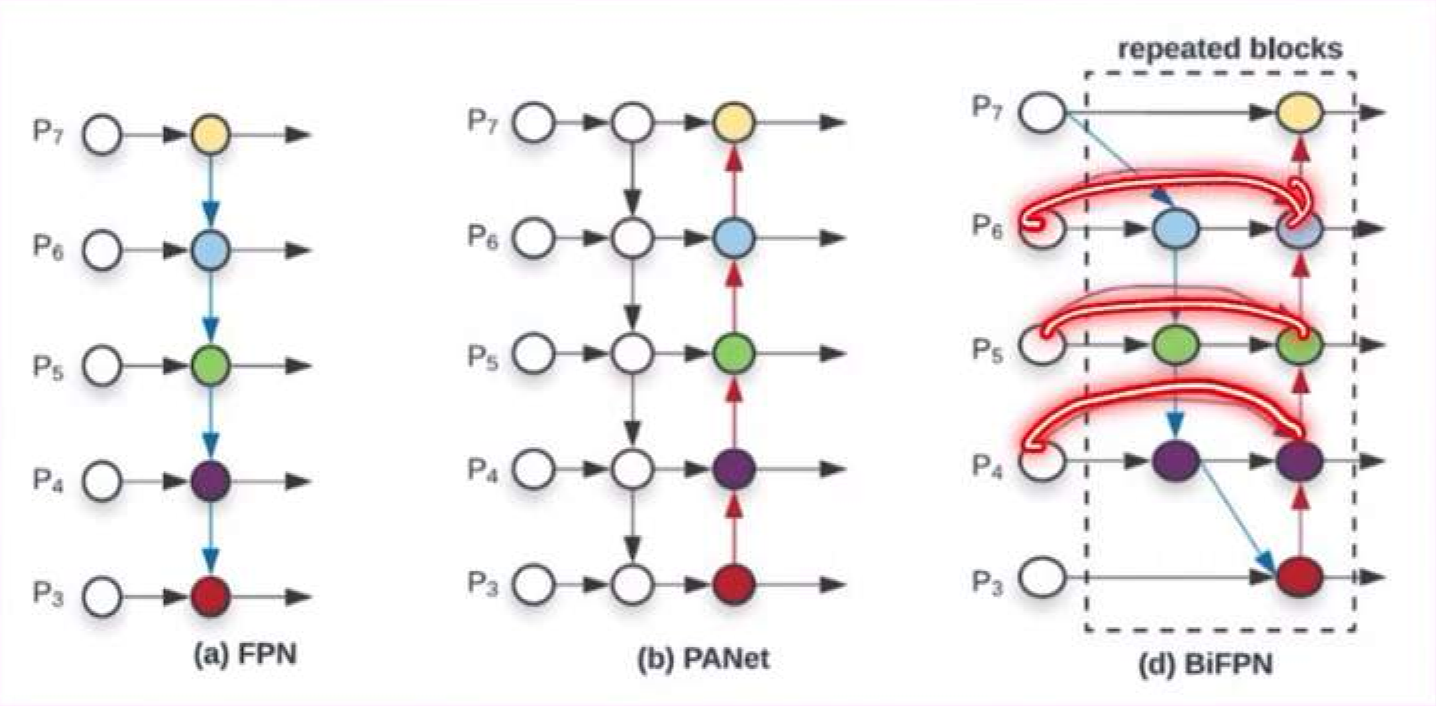
Multi-Input Weighted Residual Connections



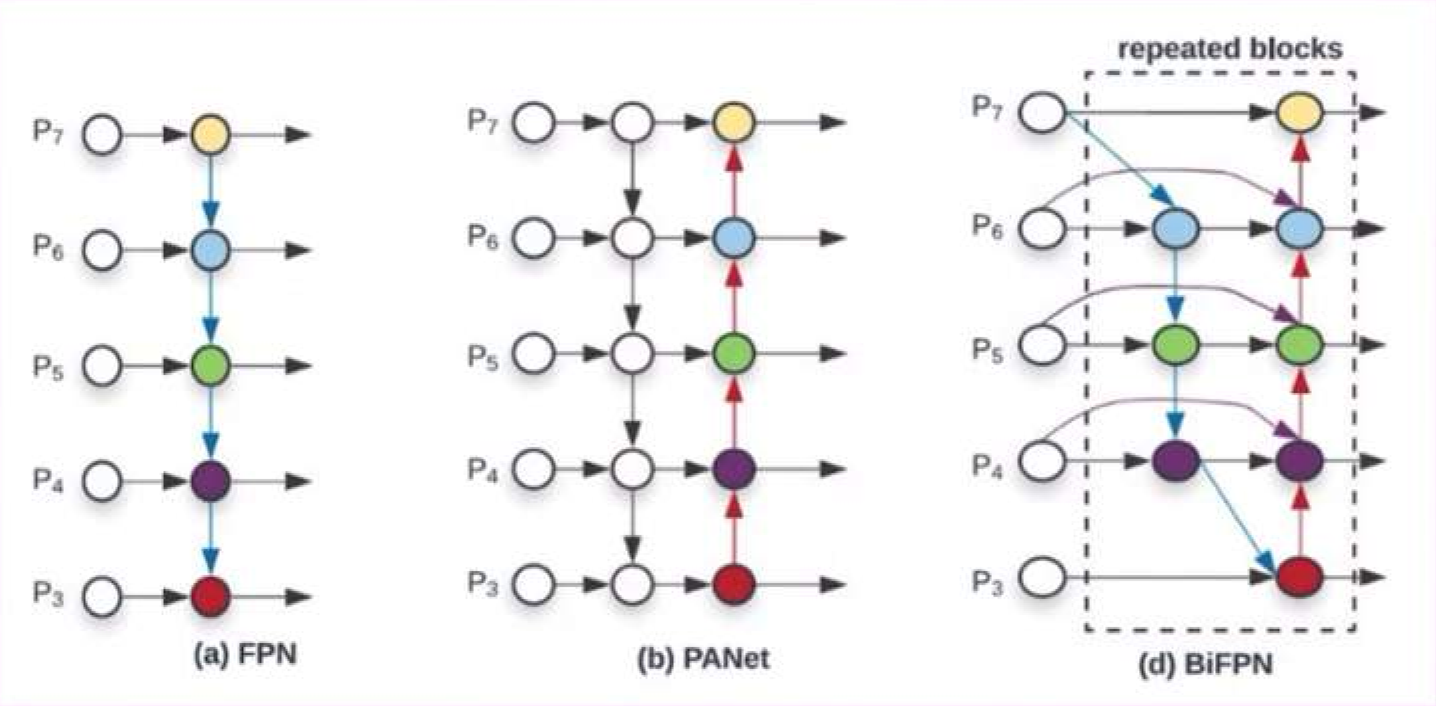
Multi-Input Weighted Residual Connections



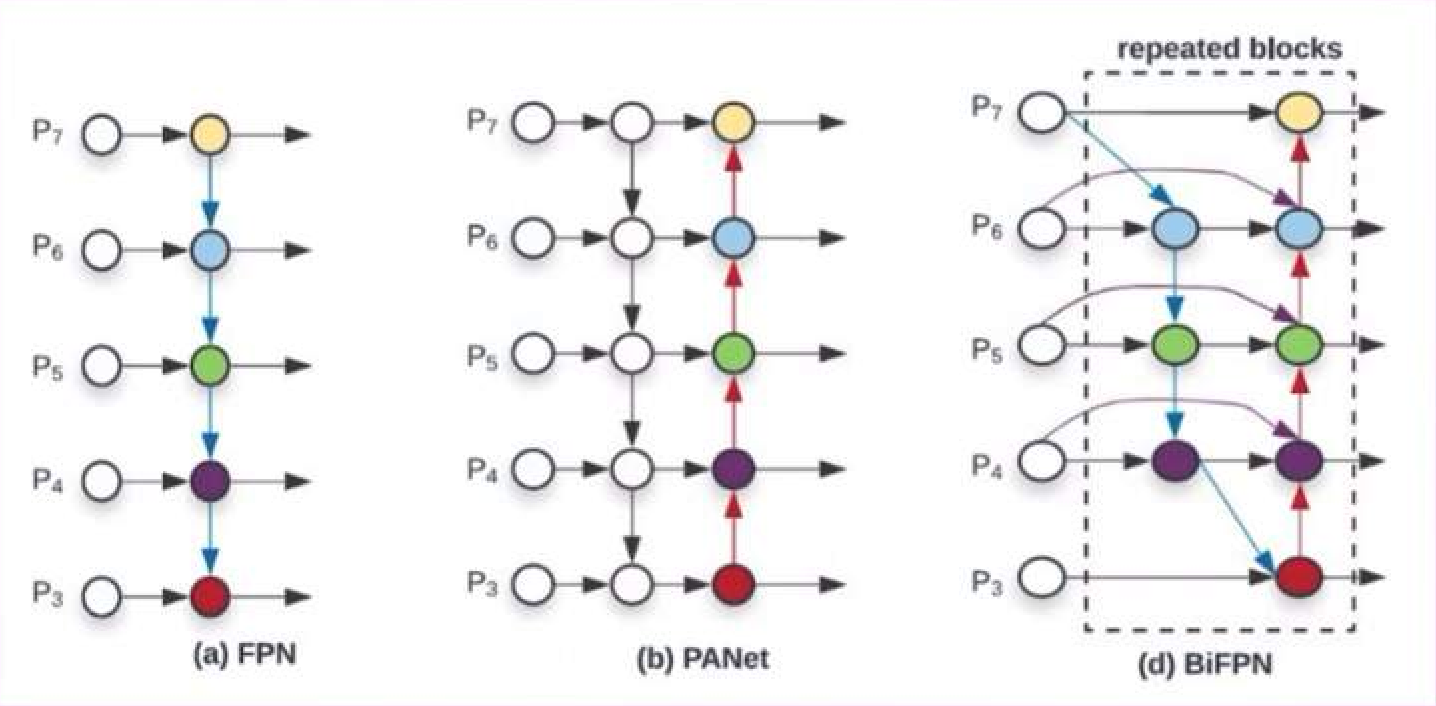
Multi-Input Weighted Residual Connections



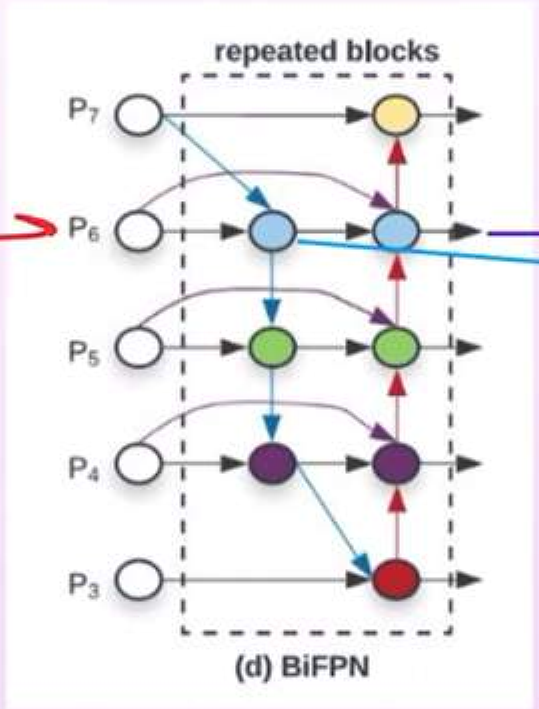
Multi-Input Weighted Residual Connections



Multi-Input Weighted Residual Connections



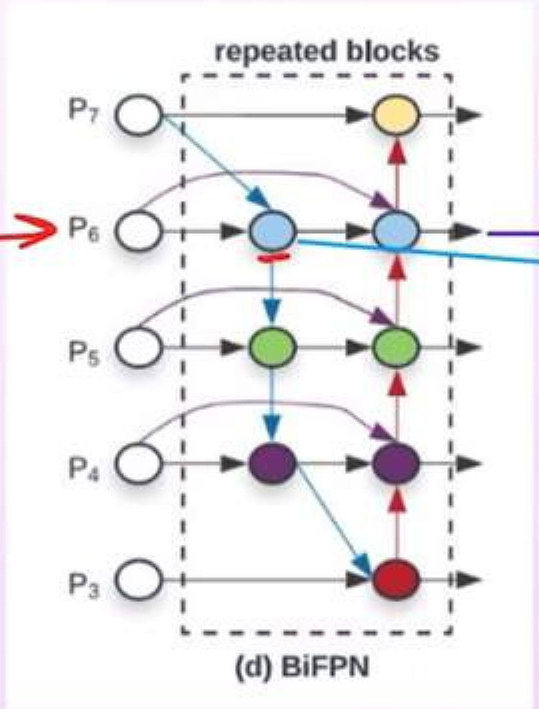
Multi-Input Weighted Residual Connections



$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

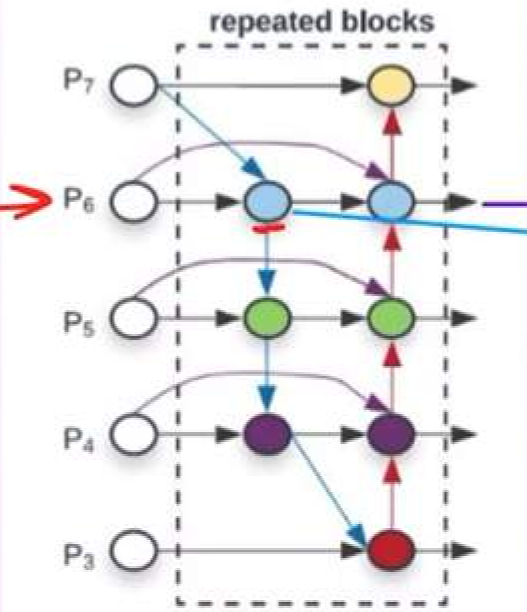
Multi-Input Weighted Residual Connections



$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

Multi-Input Weighted Residual Connections

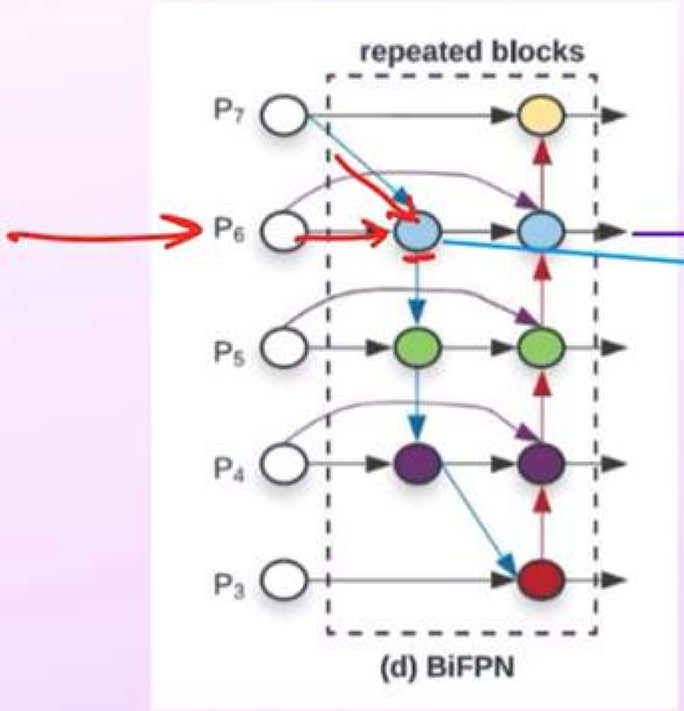


(d) BiFPN

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

Multi-Input Weighted Residual Connections

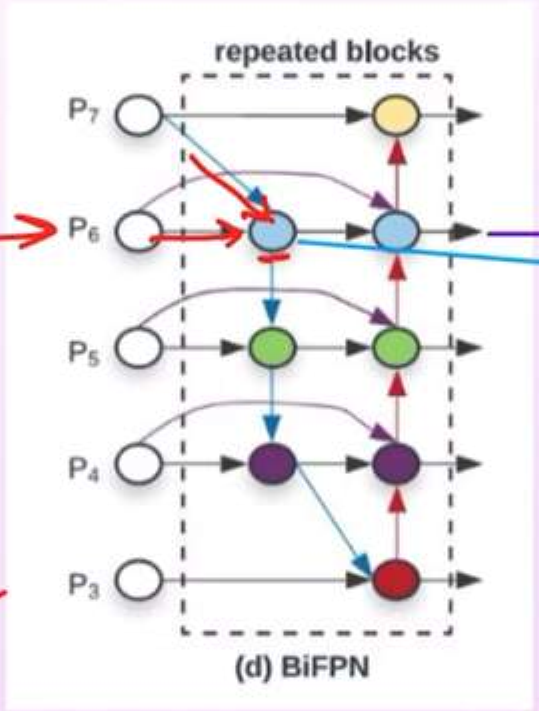


$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

Multi-Input Weighted Residual Connections

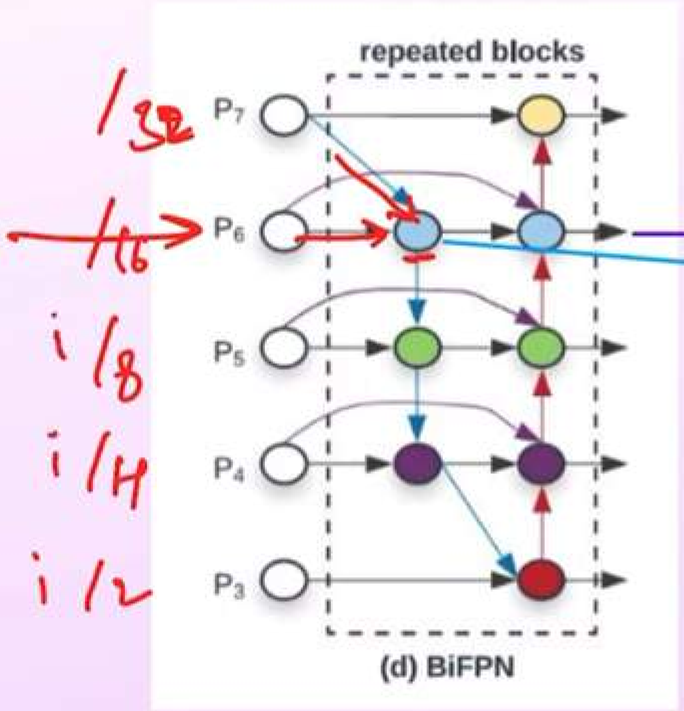
12



$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

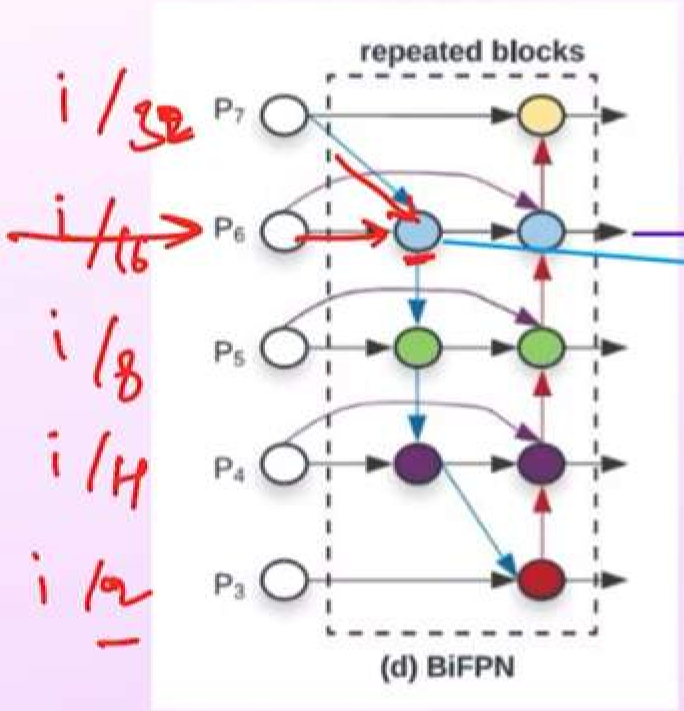
Multi-Input Weighted Residual Connections



$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

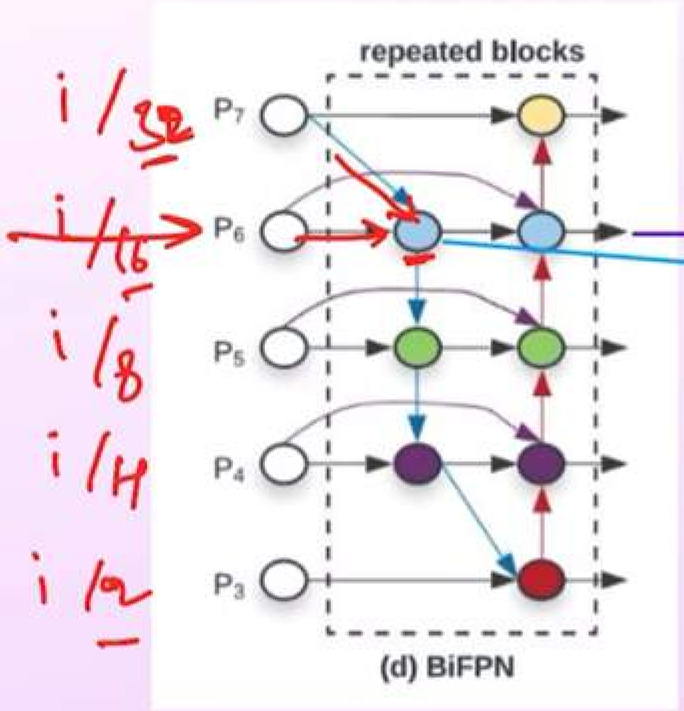
Multi-Input Weighted Residual Connections



$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w'_1 \cdot P_6^{in} + w'_2 \cdot P_6^{td} + w'_3 \cdot Resize(P_5^{out})}{w'_1 + w'_2 + w'_3 + \epsilon}\right)$$

Multi-Input Weighted Residual Connections

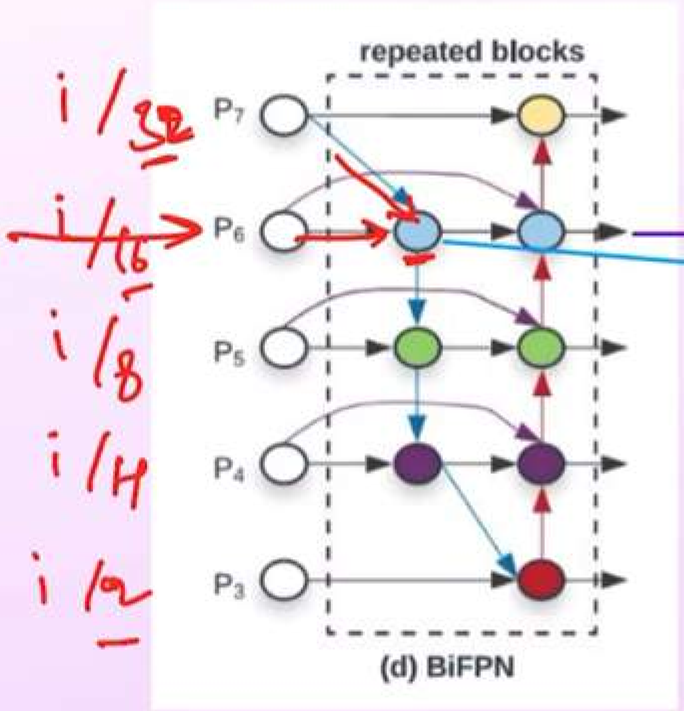


ups

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

Multi-Input Weighted Residual Connections



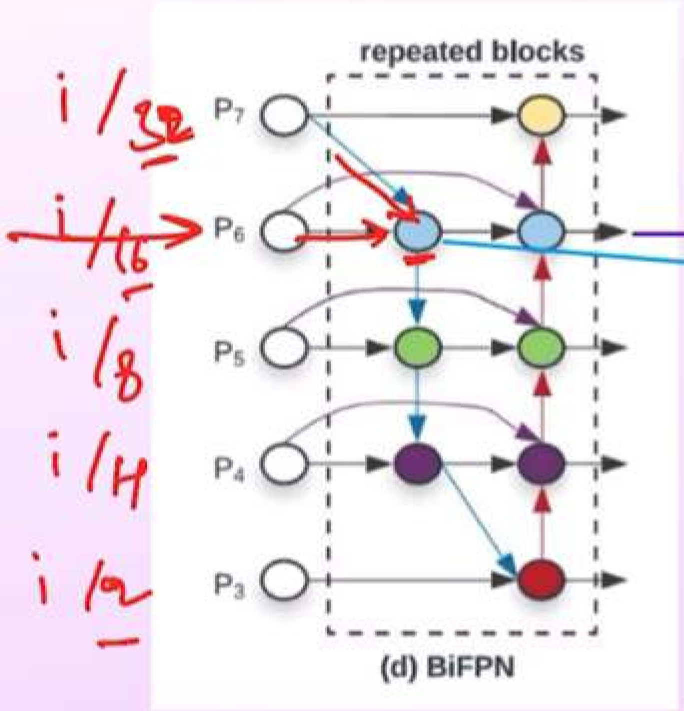
(d) BiFPN

upsampling

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

Multi-Input Weighted Residual Connections



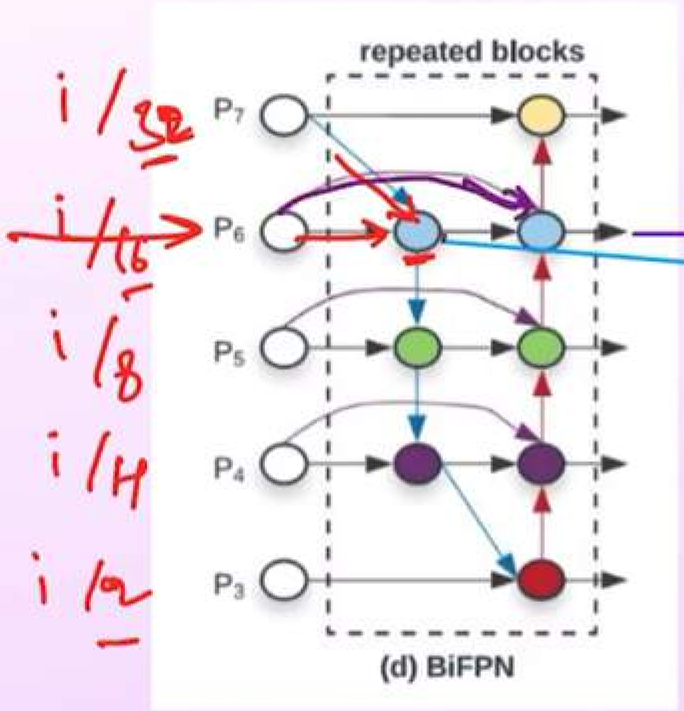
(d) BiFPN

upsampling

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

Multi-Input Weighted Residual Connections

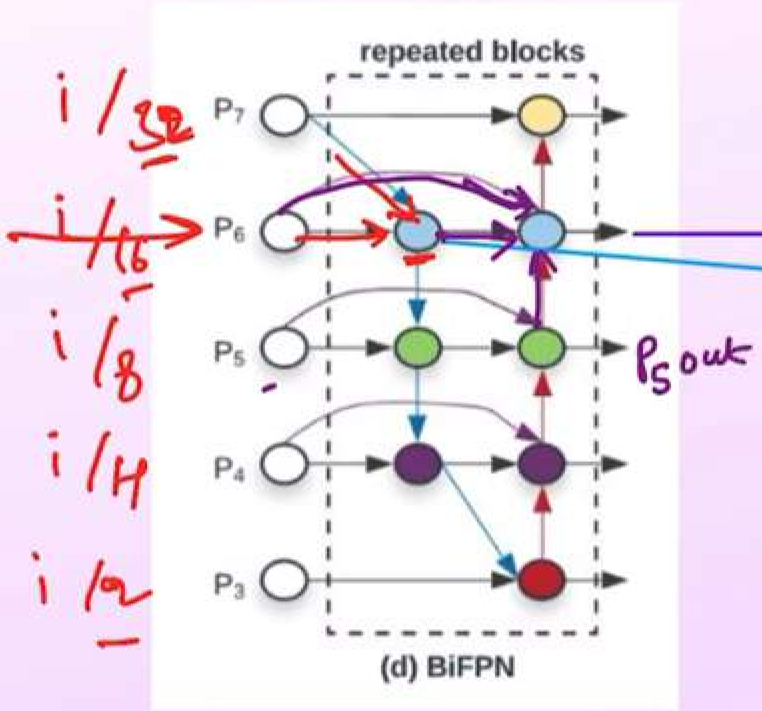


upsampling

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

Multi-Input Weighted Residual Connections



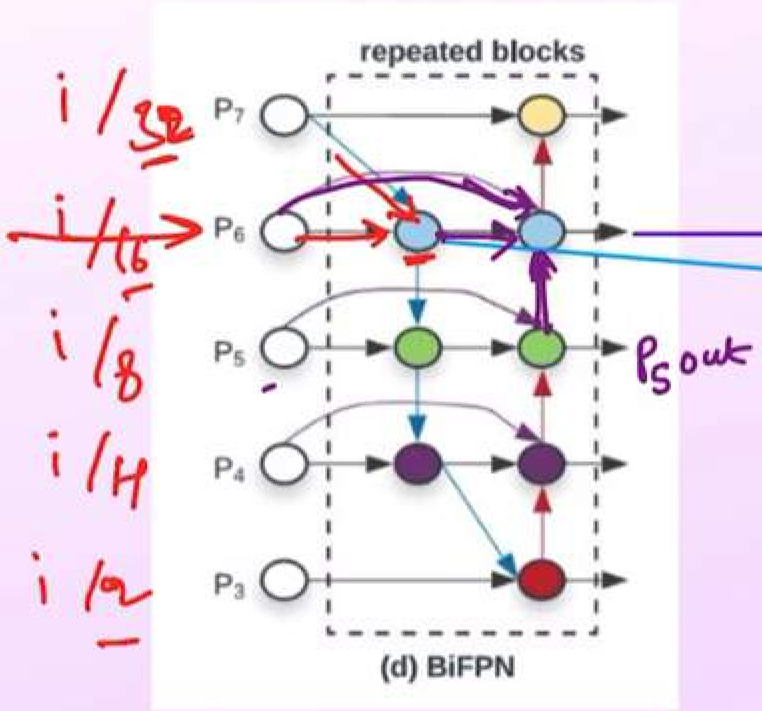
(d) BiFPN

upsampling

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w'_1 \cdot P_6^{in} + w'_2 \cdot P_6^{td} + w'_3 \cdot Resize(P_5^{out})}{w'_1 + w'_2 + w'_3 + \epsilon}\right)$$

Multi-Input Weighted Residual Connections

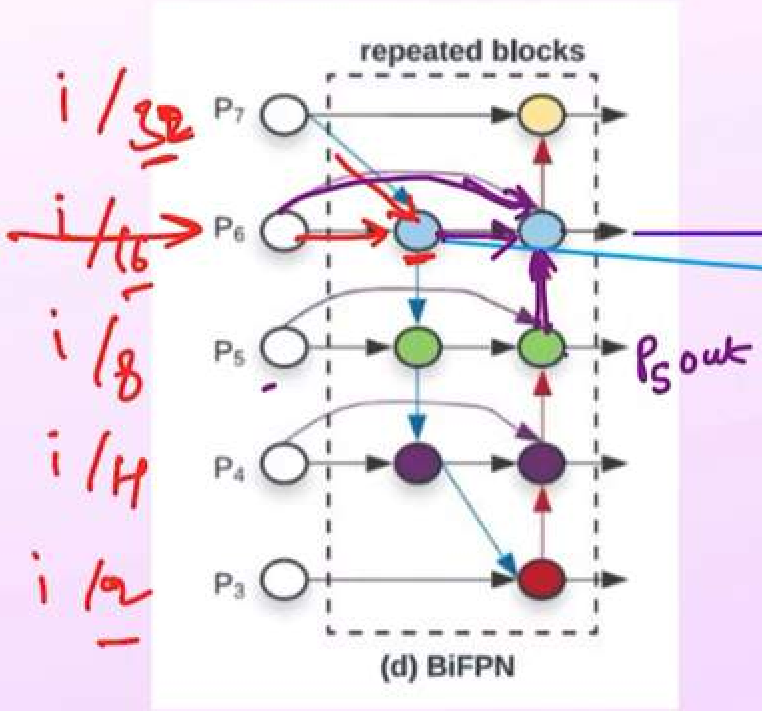


upsampling

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w'_1 \cdot P_6^{in} + w'_2 \cdot P_6^{td} + w'_3 \cdot Resize(P_5^{out})}{w'_1 + w'_2 + w'_3 + \epsilon}\right)$$

Multi-Input Weighted Residual Connections

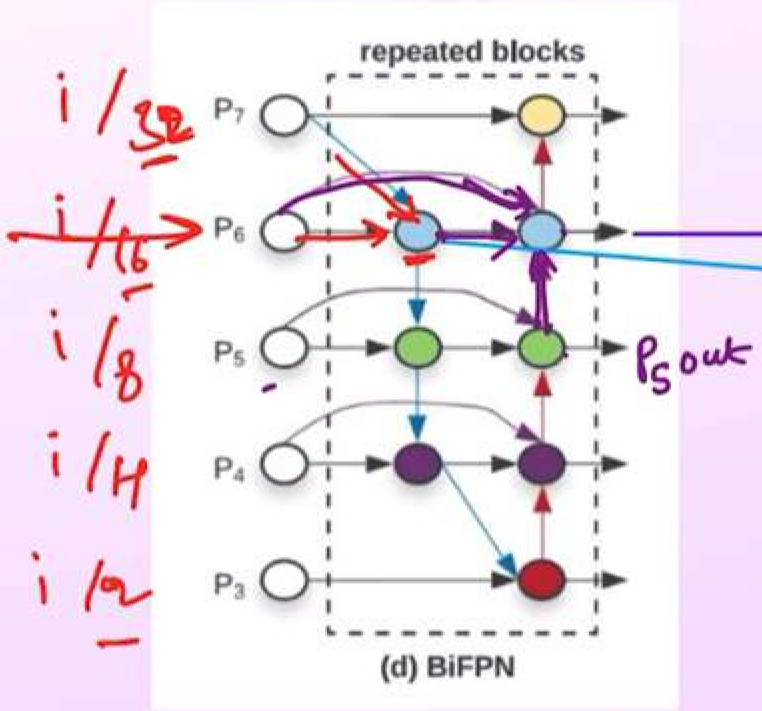


upsampling

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

Multi-Input Weighted Residual Connections



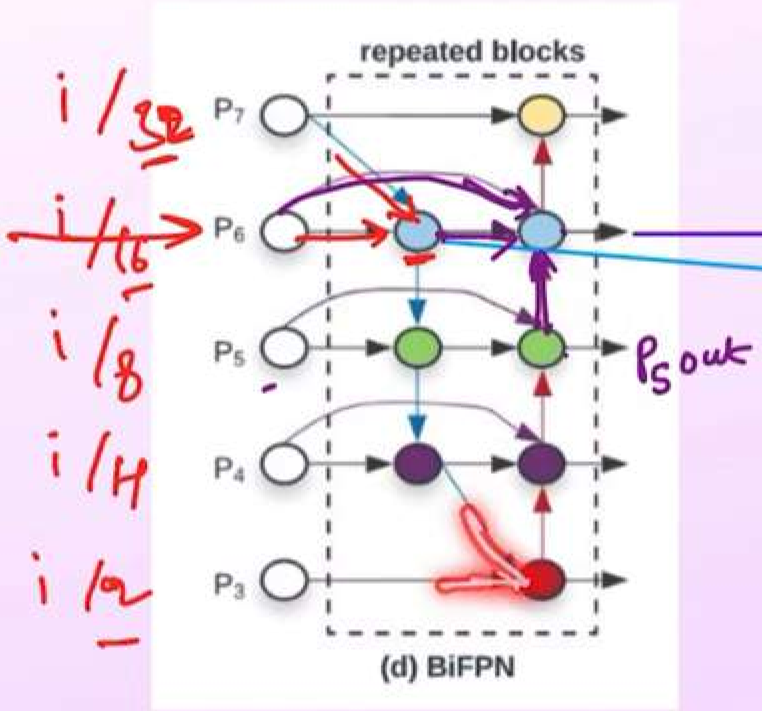
(d) BiFPN

upsampling

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

Multi-Input Weighted Residual Connections

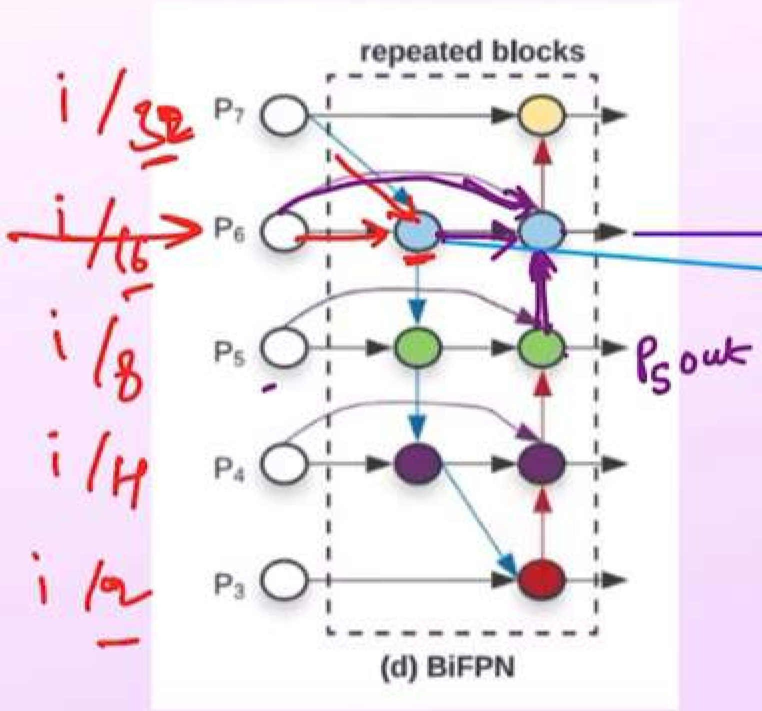


upsampling

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

Multi-Input Weighted Residual Connections



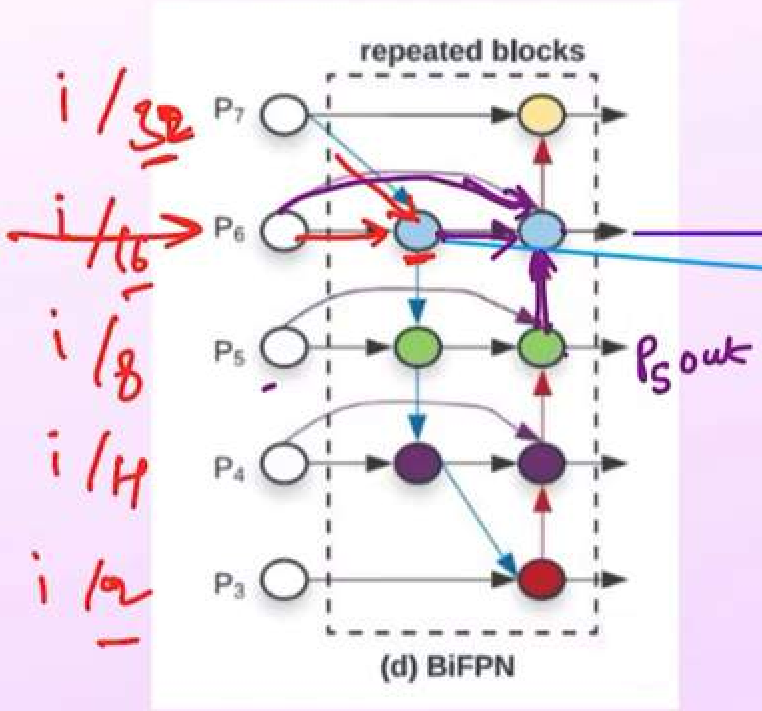
(d) BiFPN

upsampling

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w'_1 \cdot P_6^{in} + w'_2 \cdot P_6^{td} + w'_3 \cdot Resize(P_5^{out})}{w'_1 + w'_2 + w'_3 + \epsilon}\right)$$

Multi-Input Weighted Residual Connections

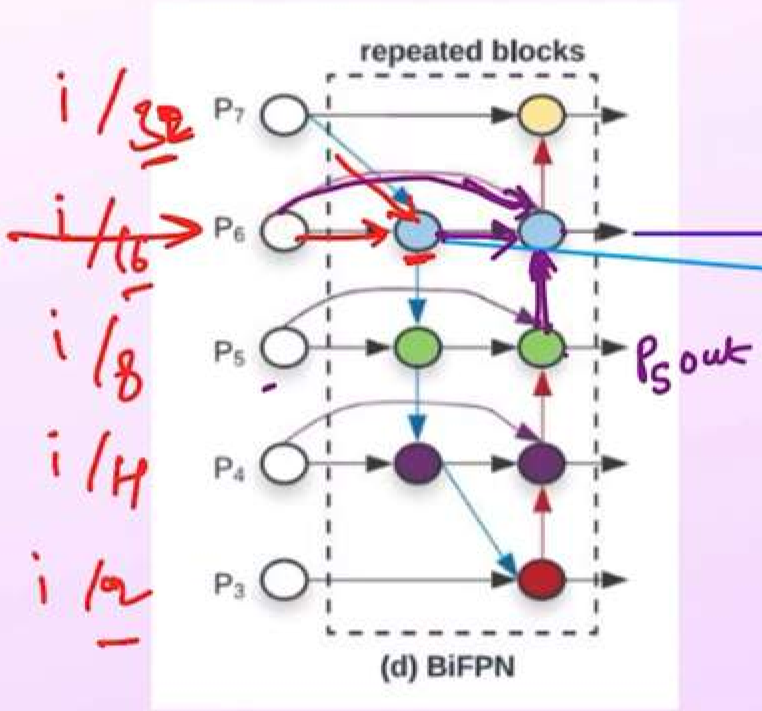


upsampling

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w'_1 \cdot P_6^{in} + w'_2 \cdot P_6^{td} + w'_3 \cdot Resize(P_5^{out})}{w'_1 + w'_2 + w'_3 + \epsilon}\right)$$

Multi-Input Weighted Residual Connections



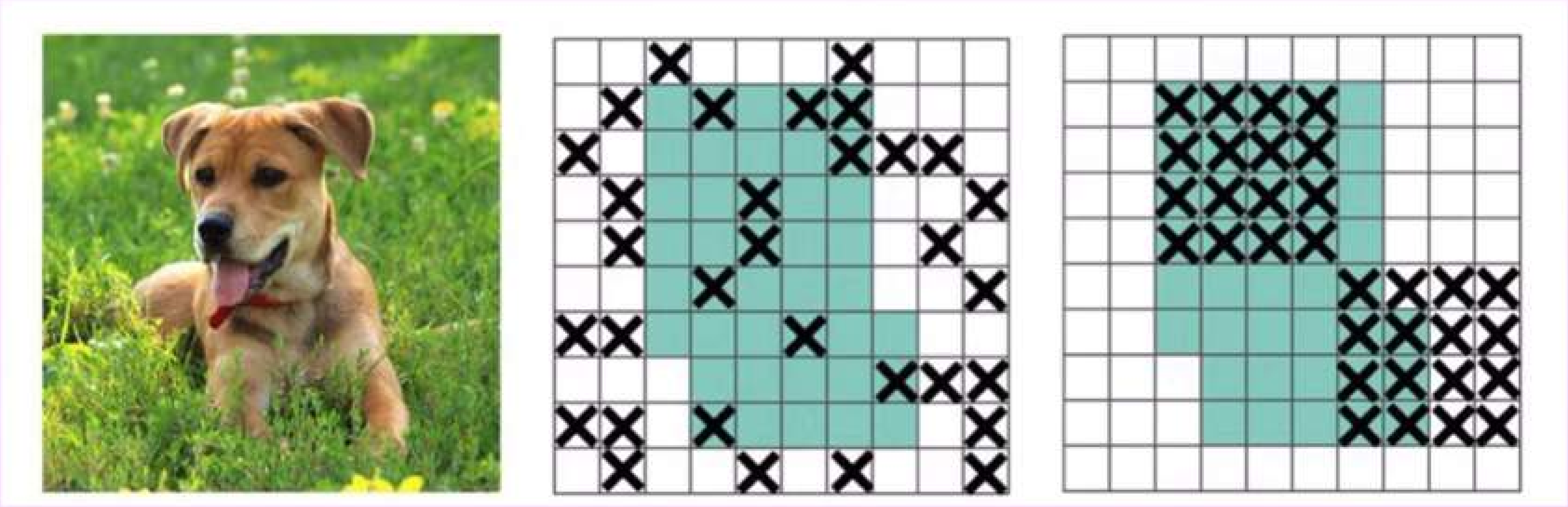
upsampling

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

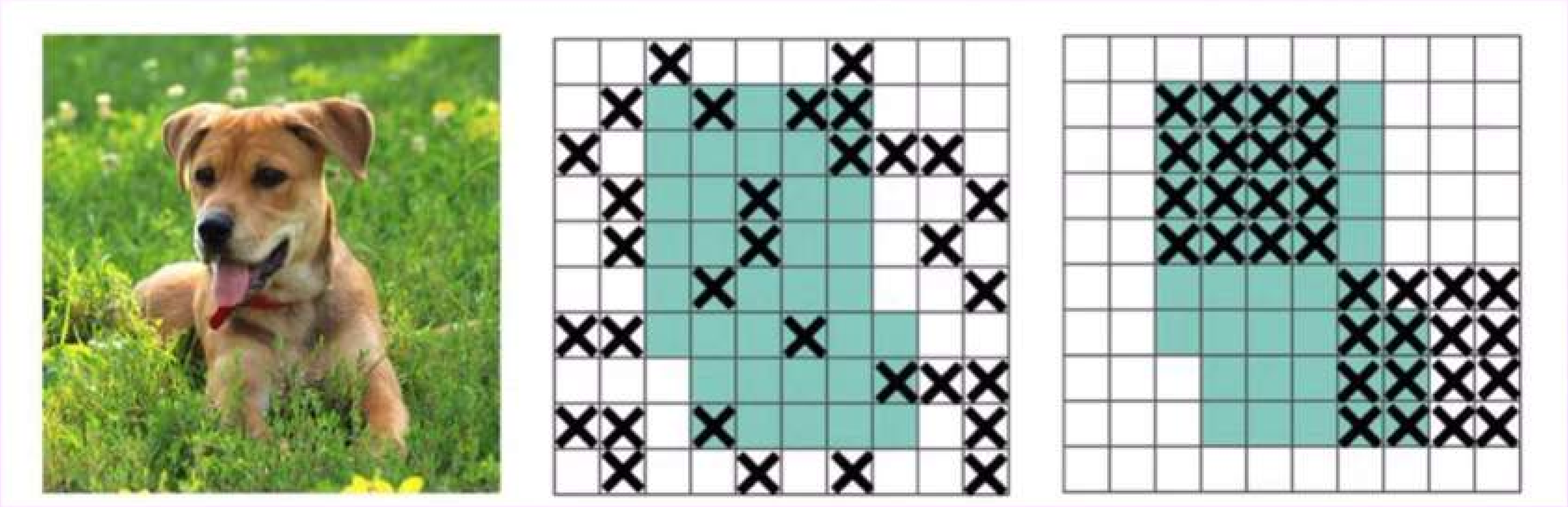
Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers



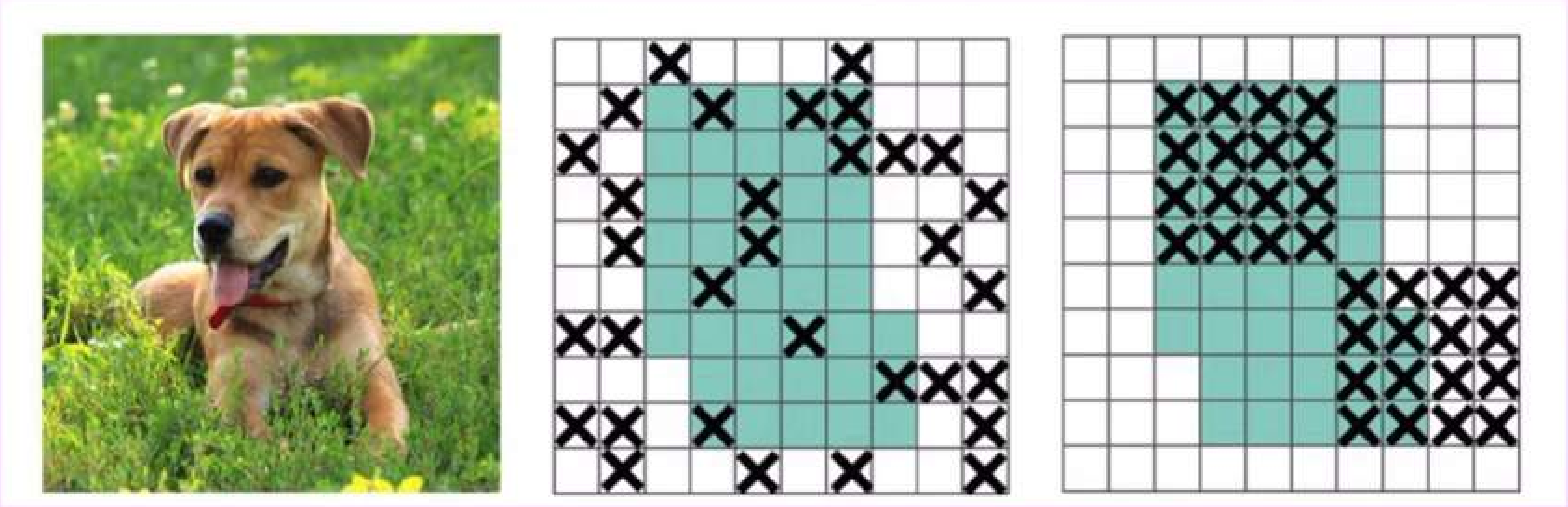
Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers



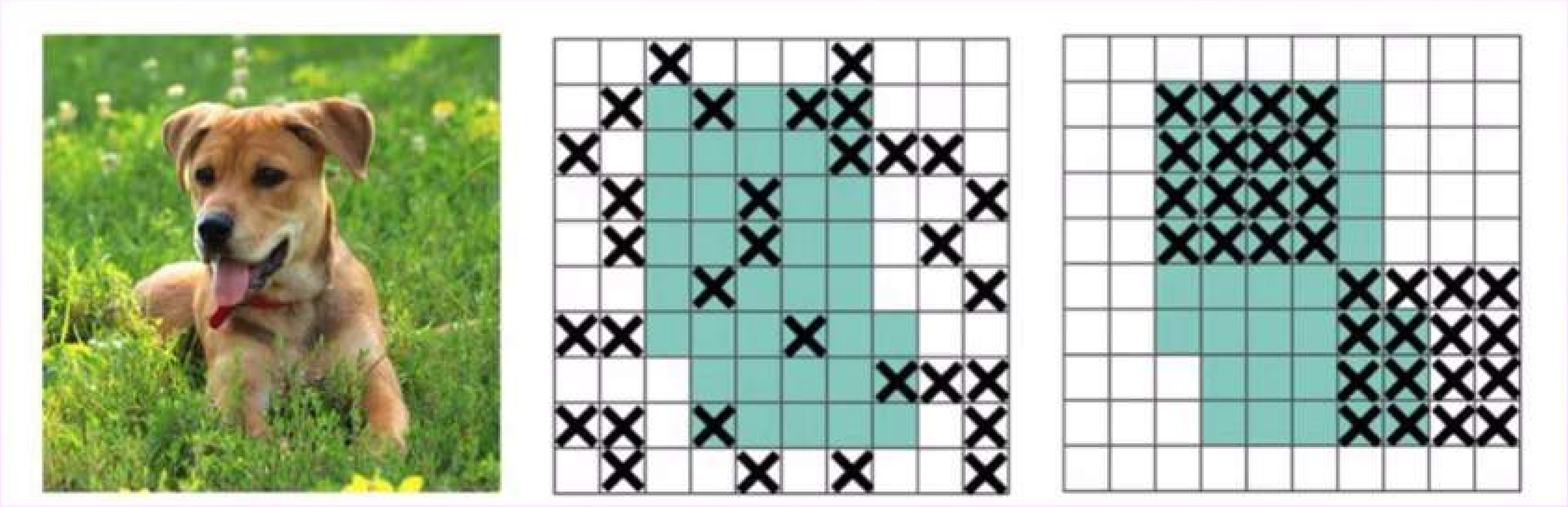
Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers



Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

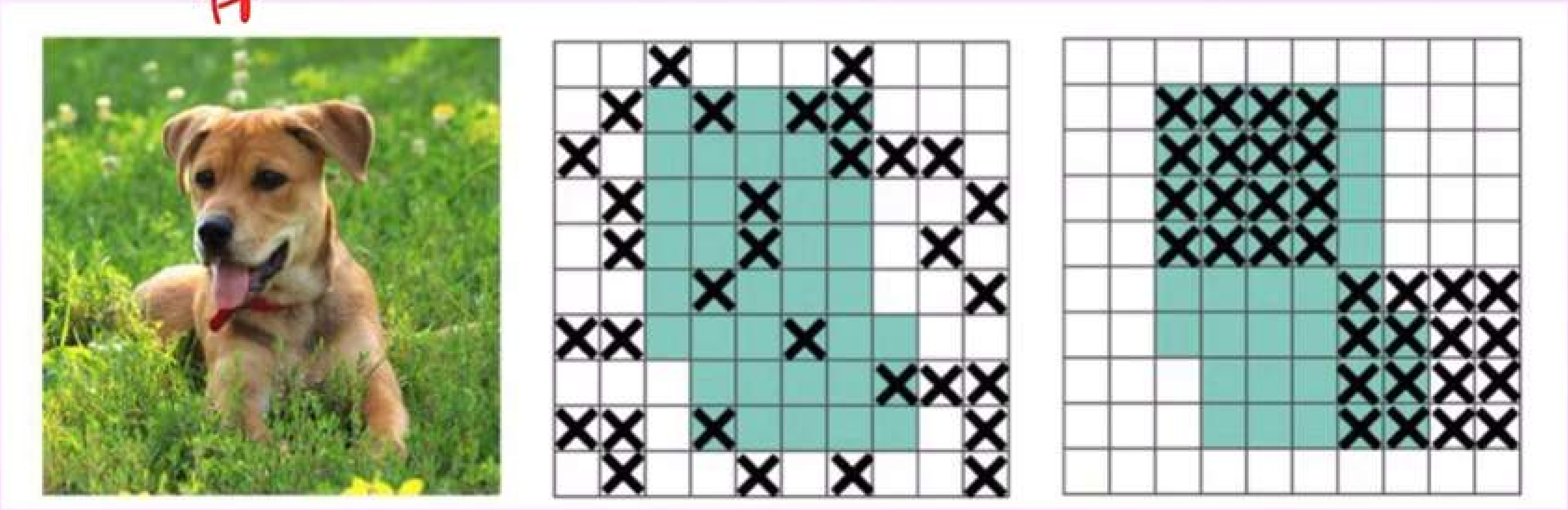


Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

i/p

Feature map



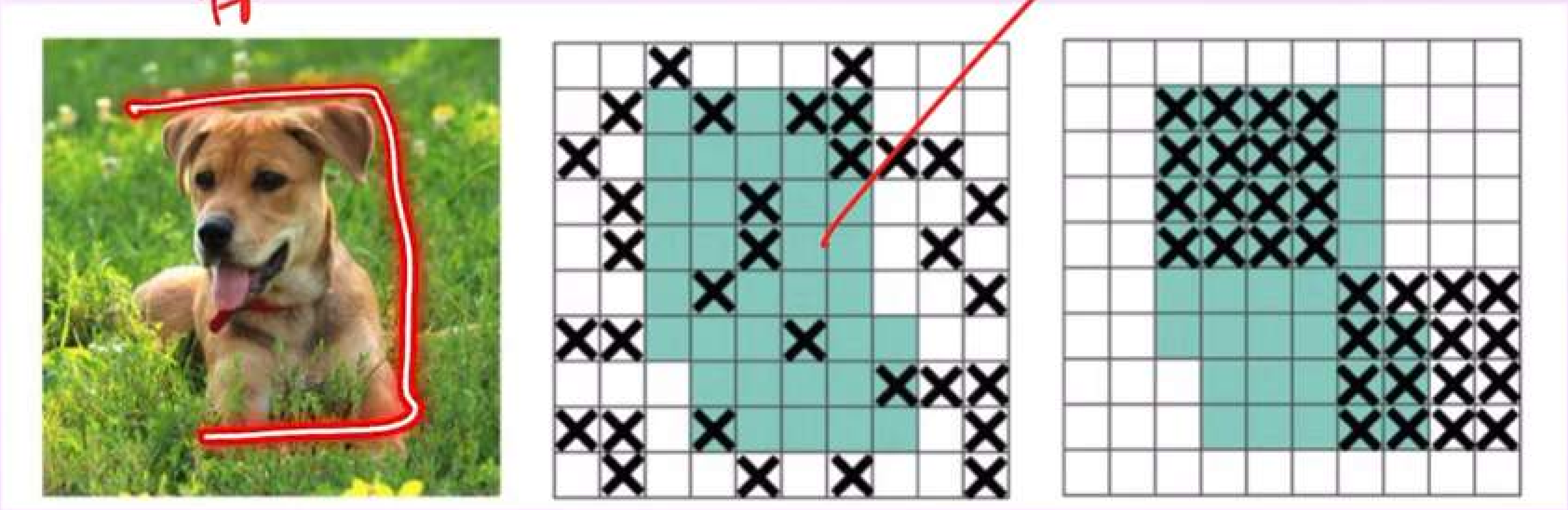
Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

Activations

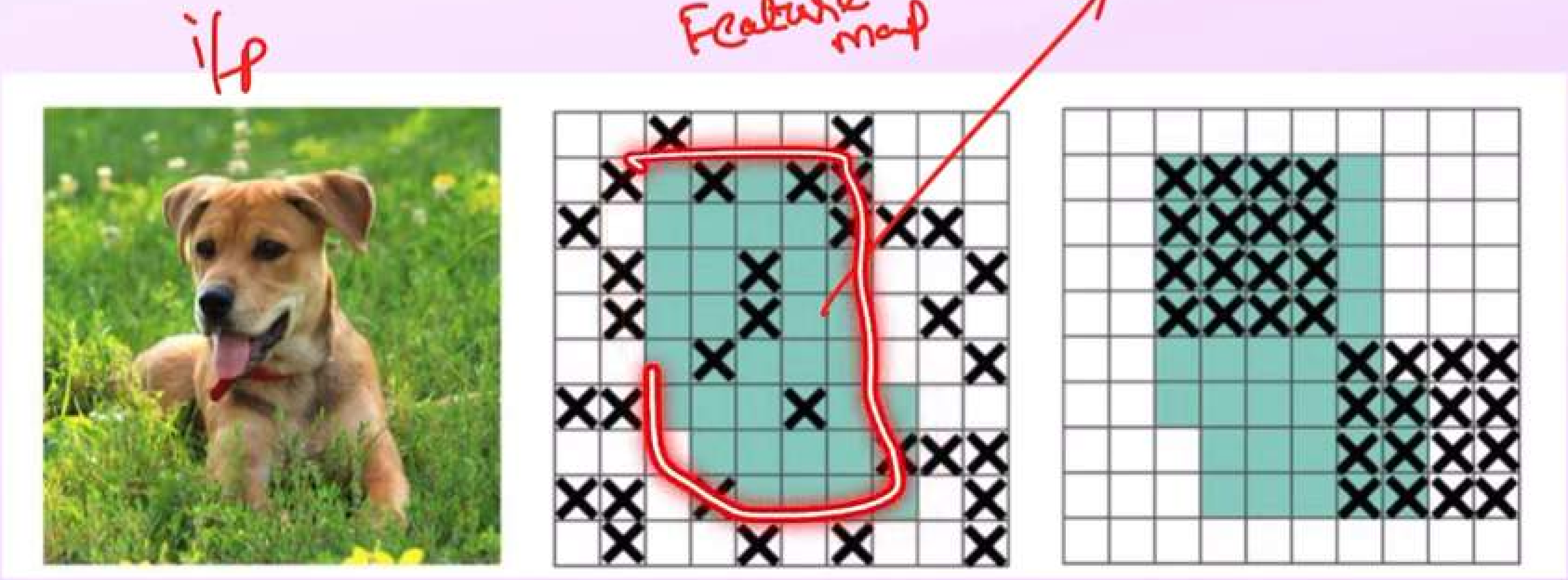
i/p

Feature map



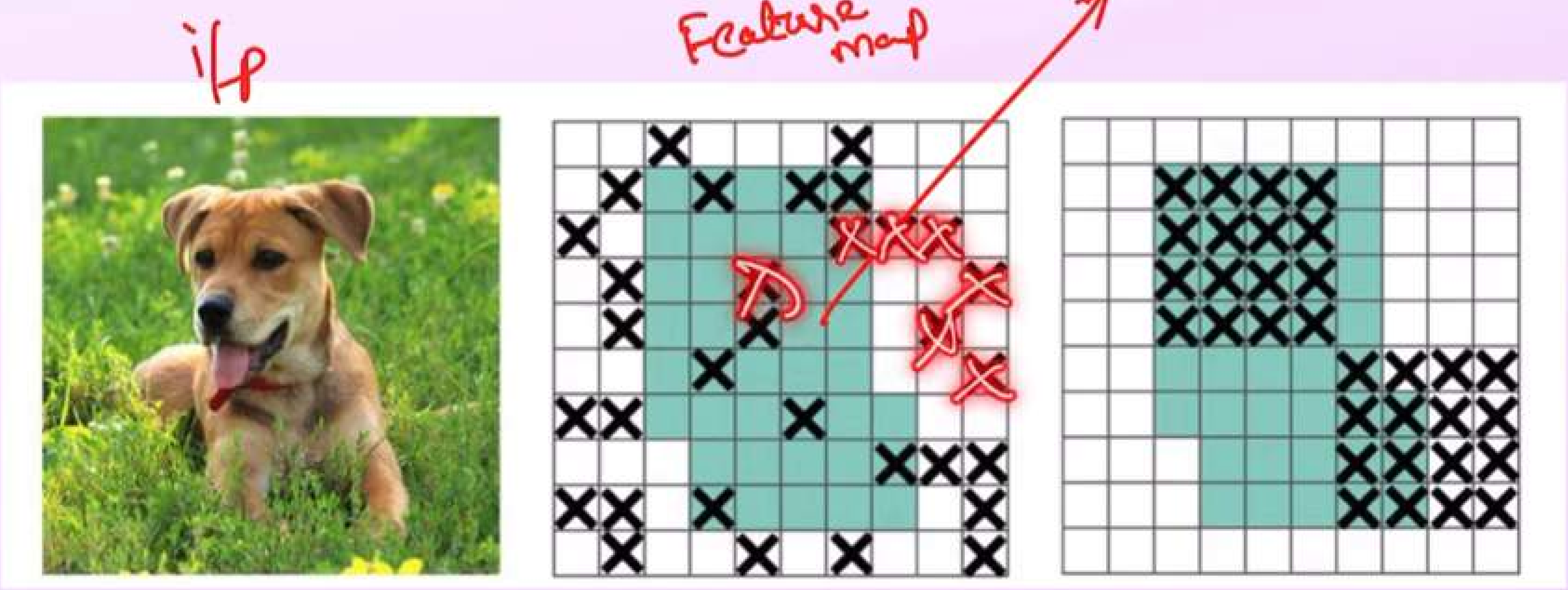
Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers



Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

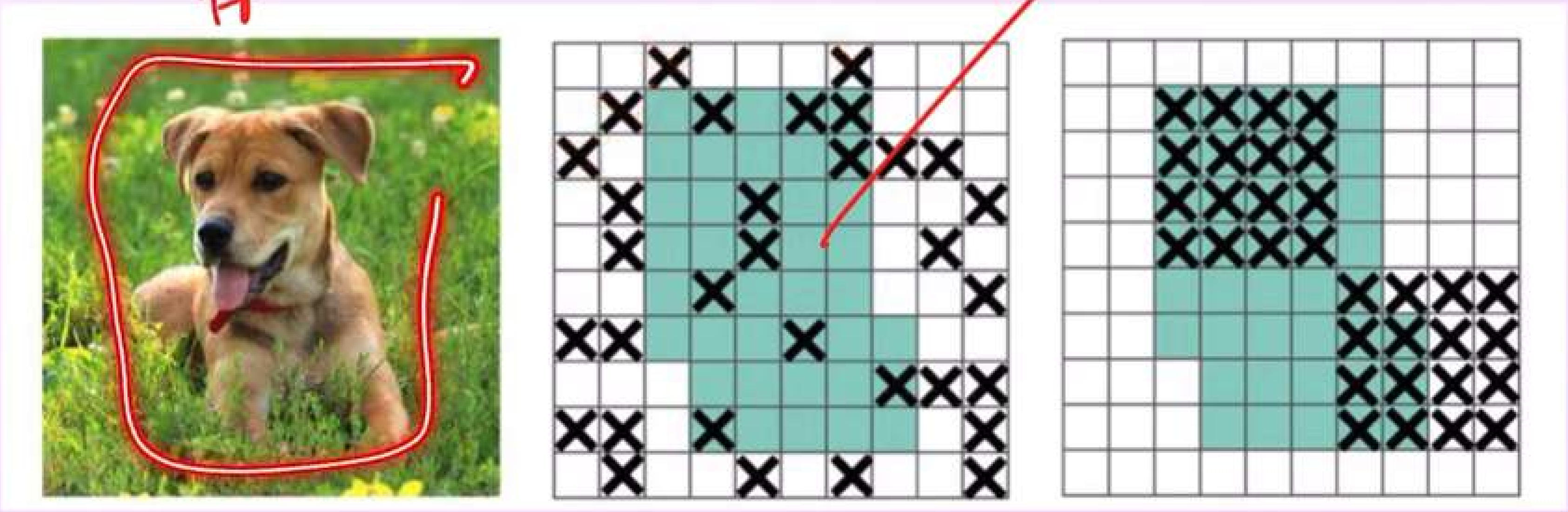


Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

Activations

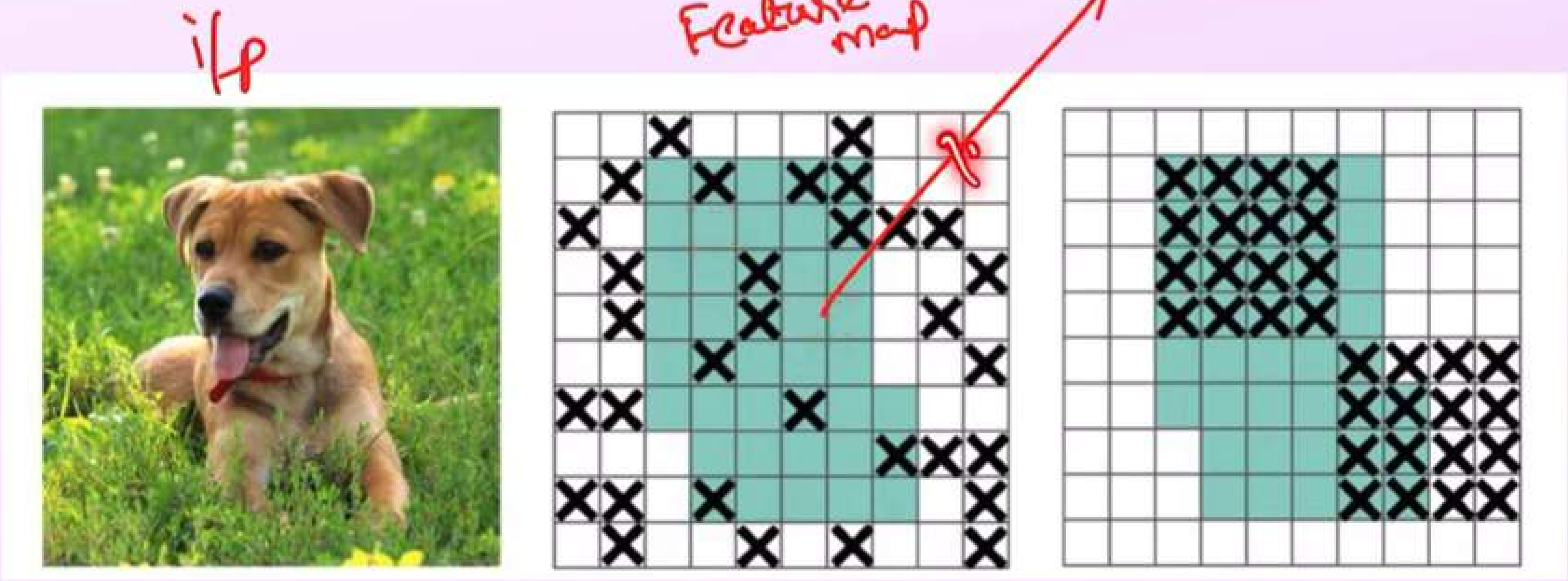
i/p Feature map



Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

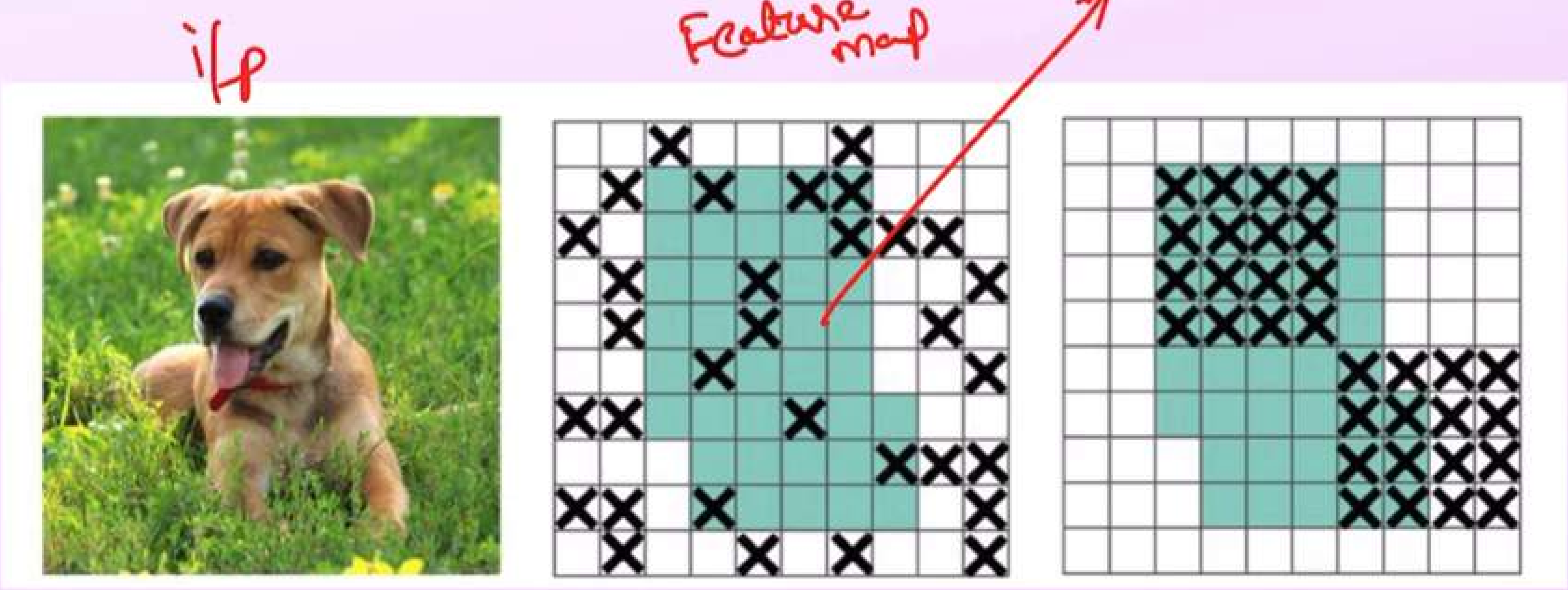
Activations



Drop Block Regularization

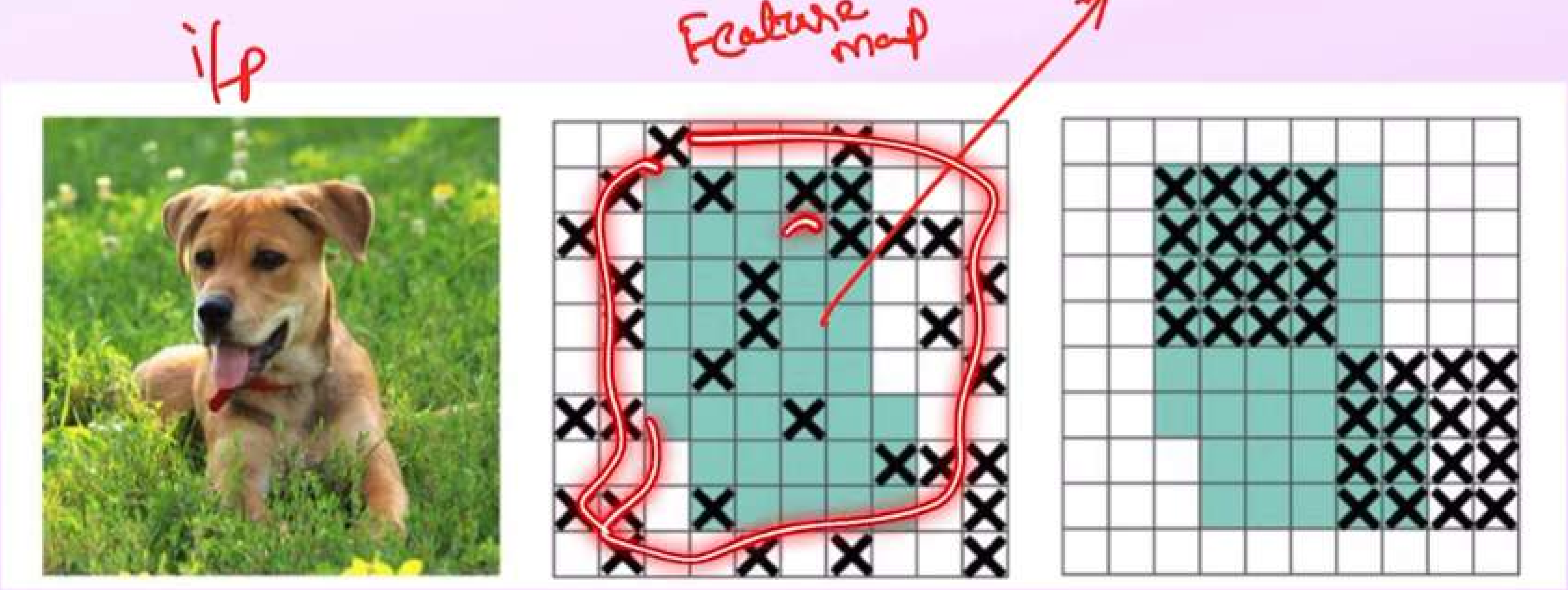
- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

0.5



Drop Block Regularization

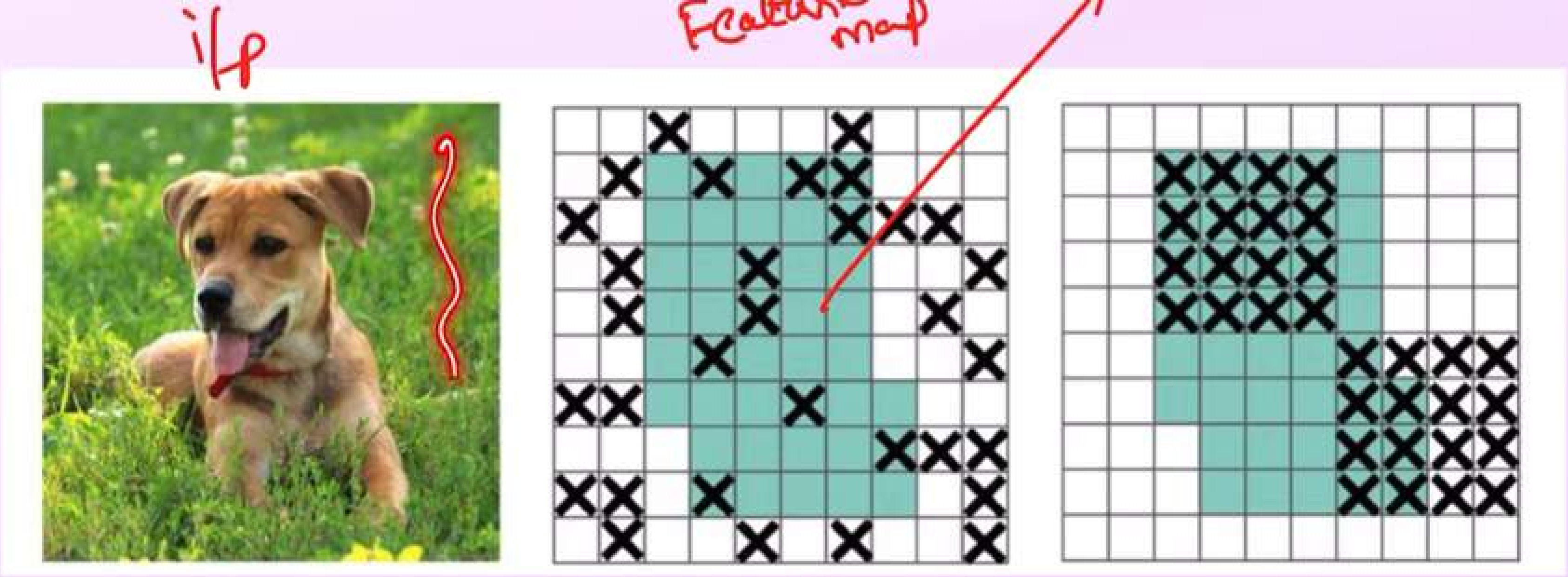
- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers



Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

Activations



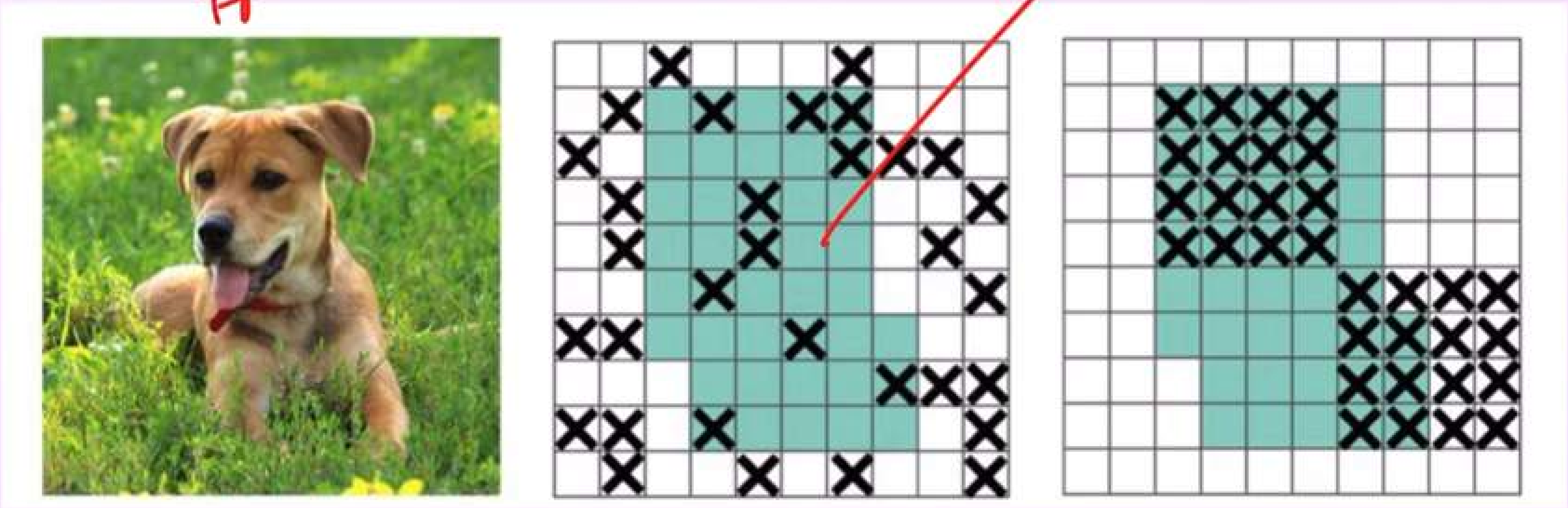
Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

Activations

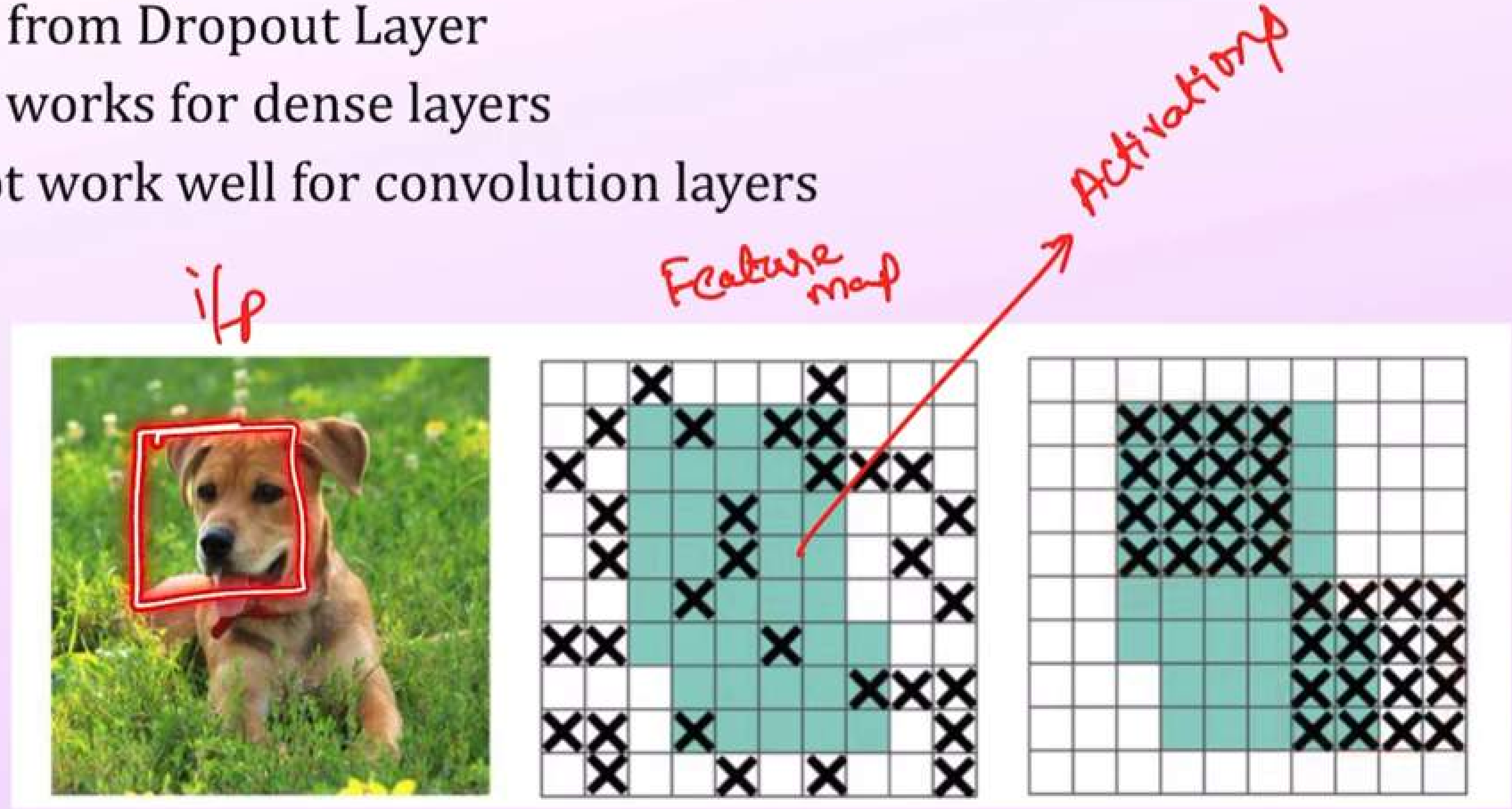
i/p

Feature map



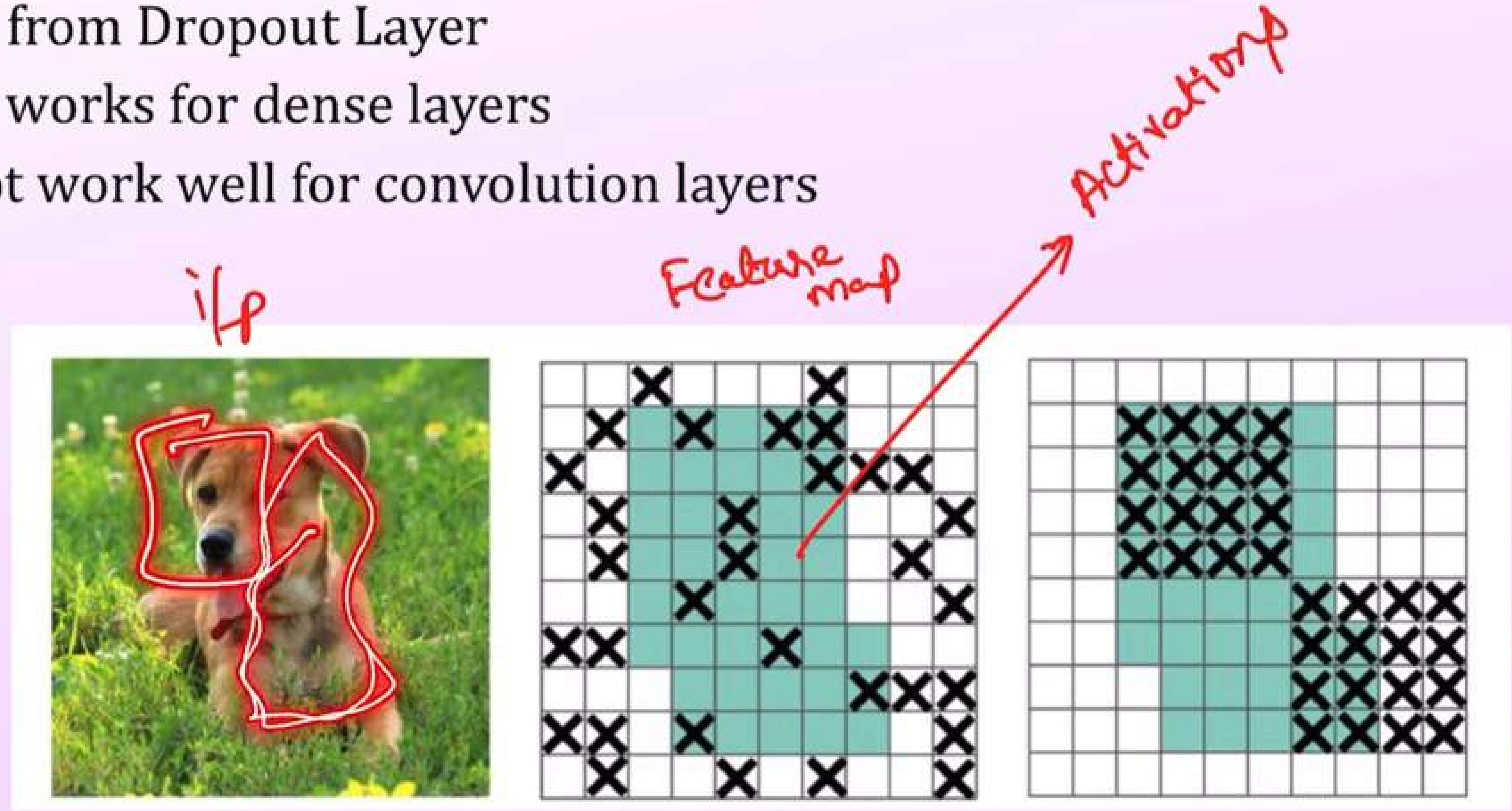
Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers



Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers



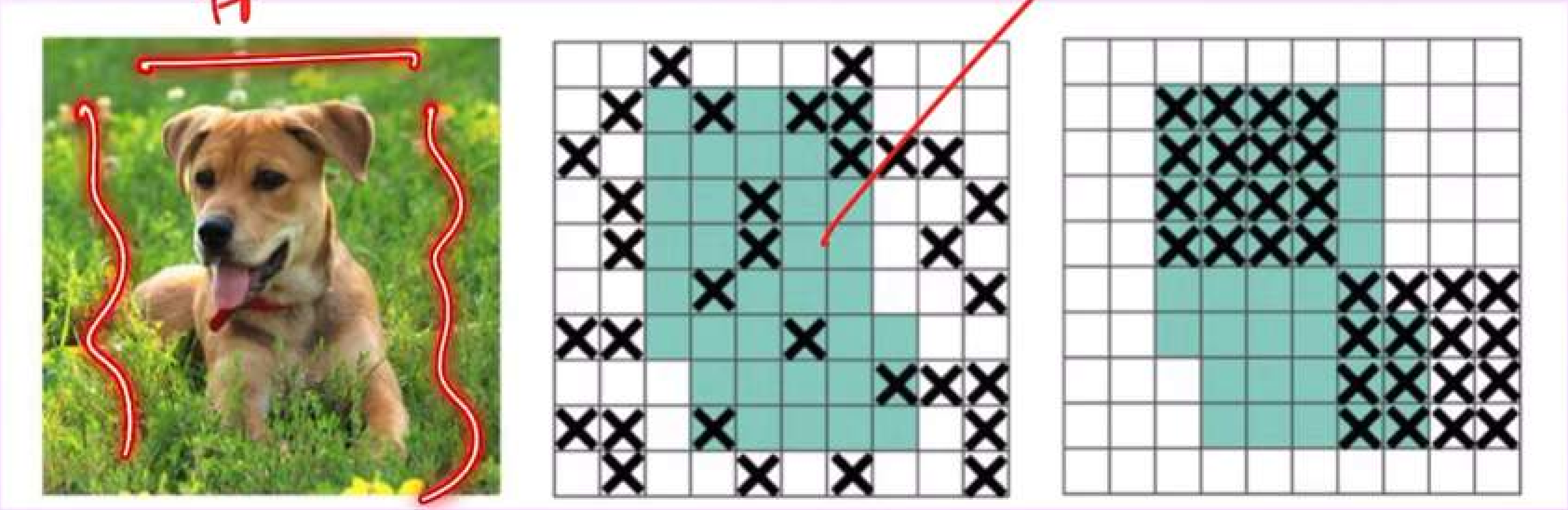
Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

Activations

i/p

Feature map

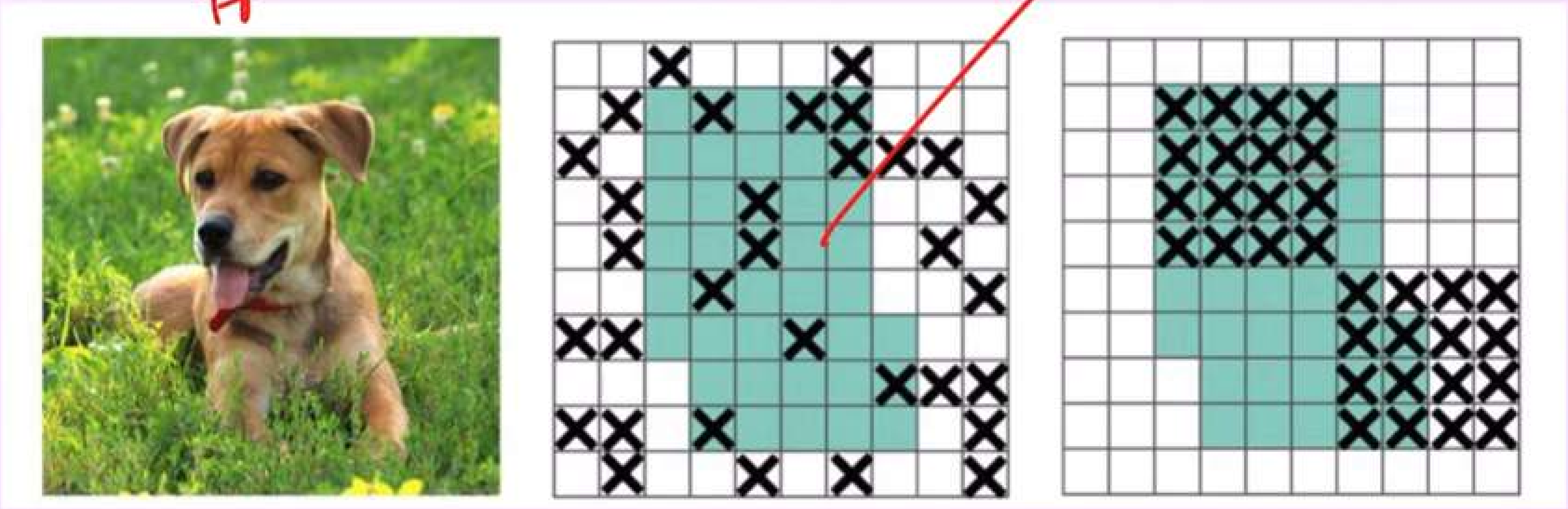


Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

Activations

i/p Feature map

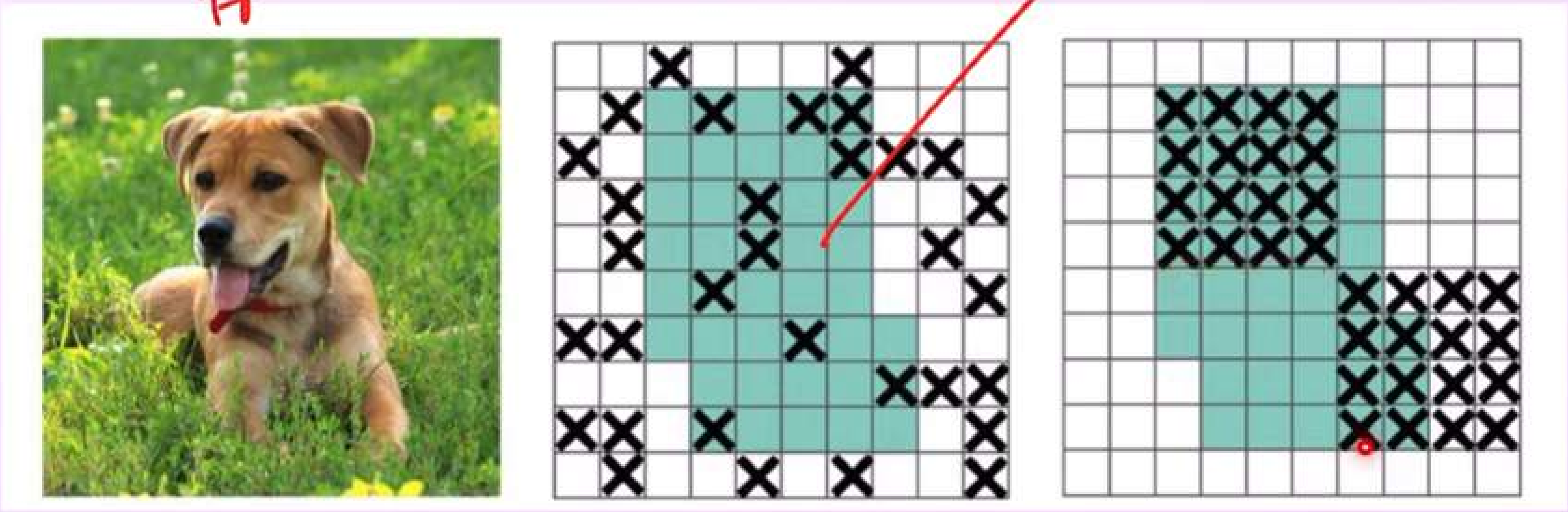


Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

Activations

i/p Feature map

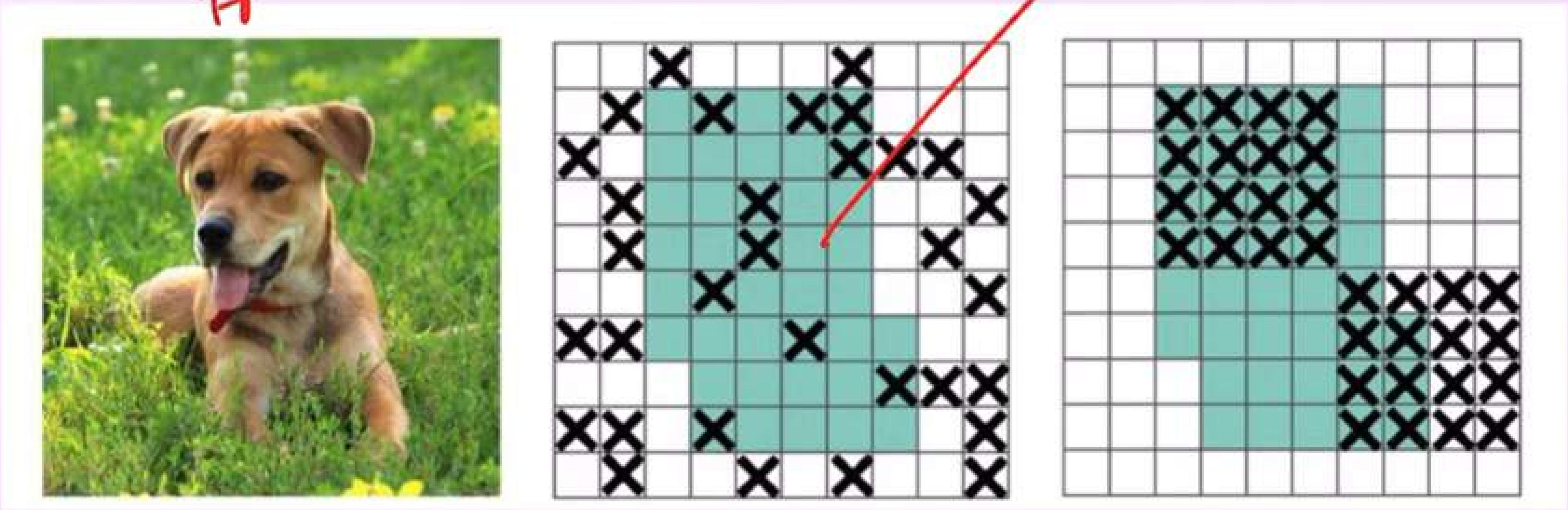


Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

Activations

i/p Feature map



Drop Block Regularization

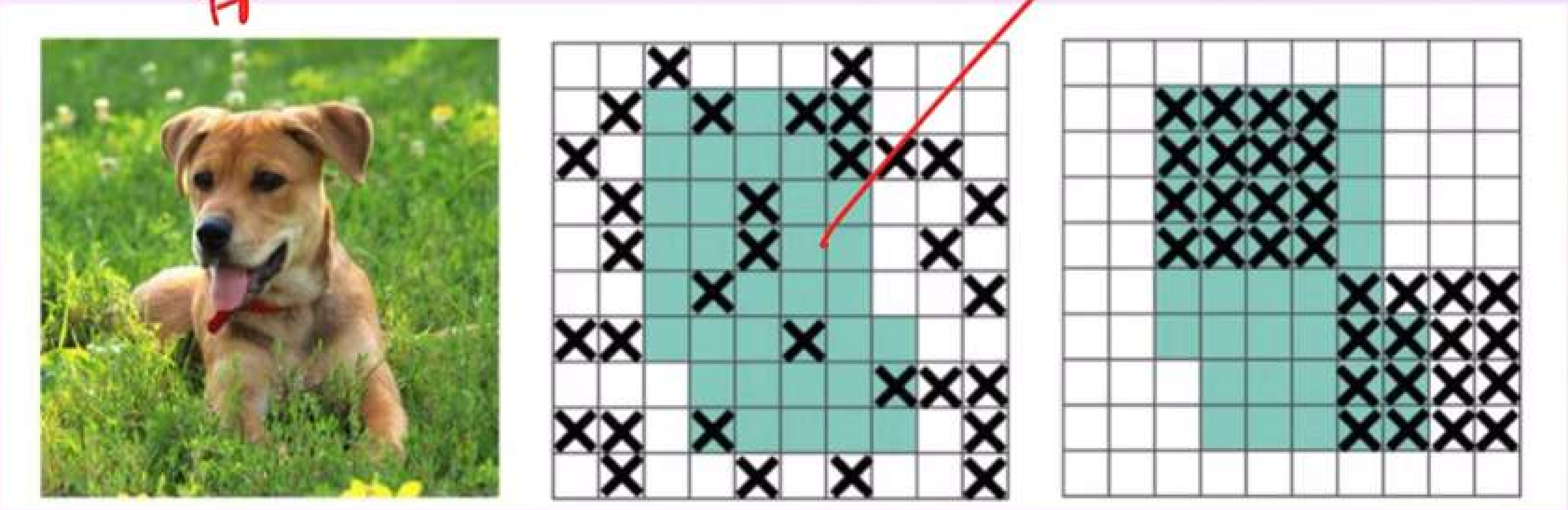
- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

Activations

P=0.1 → 10%

i/p

Feature map



Drop Block Regularization

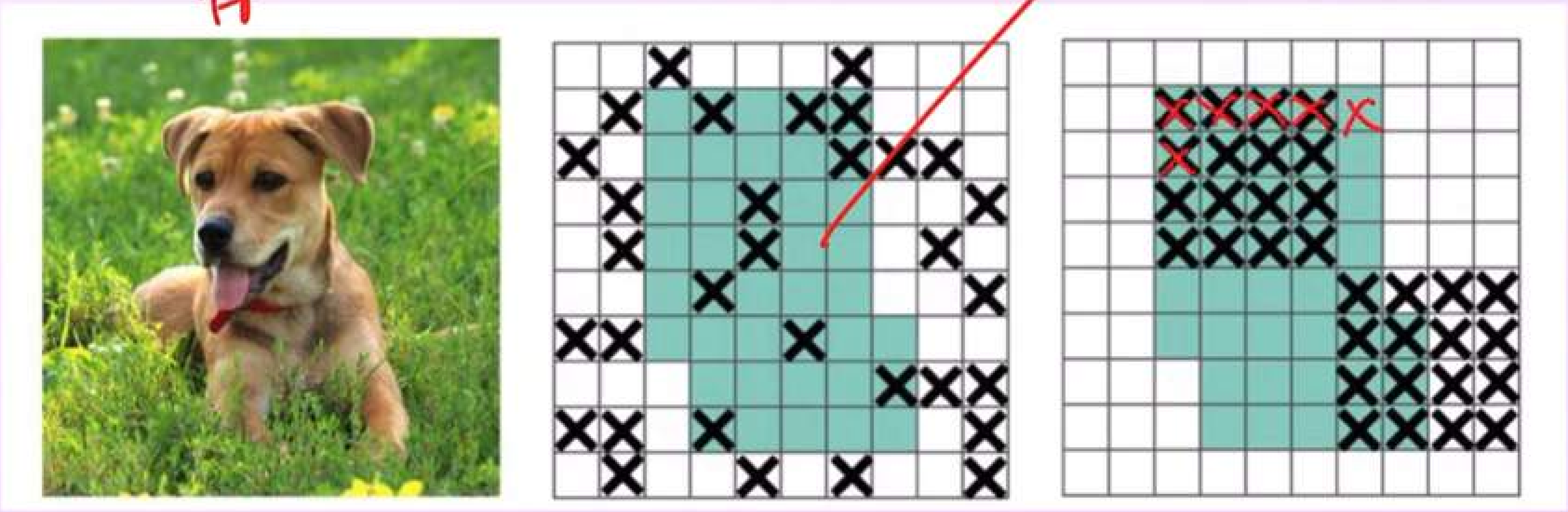
- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

Activations

P=0.1 → 10%

i/p

Feature map



Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

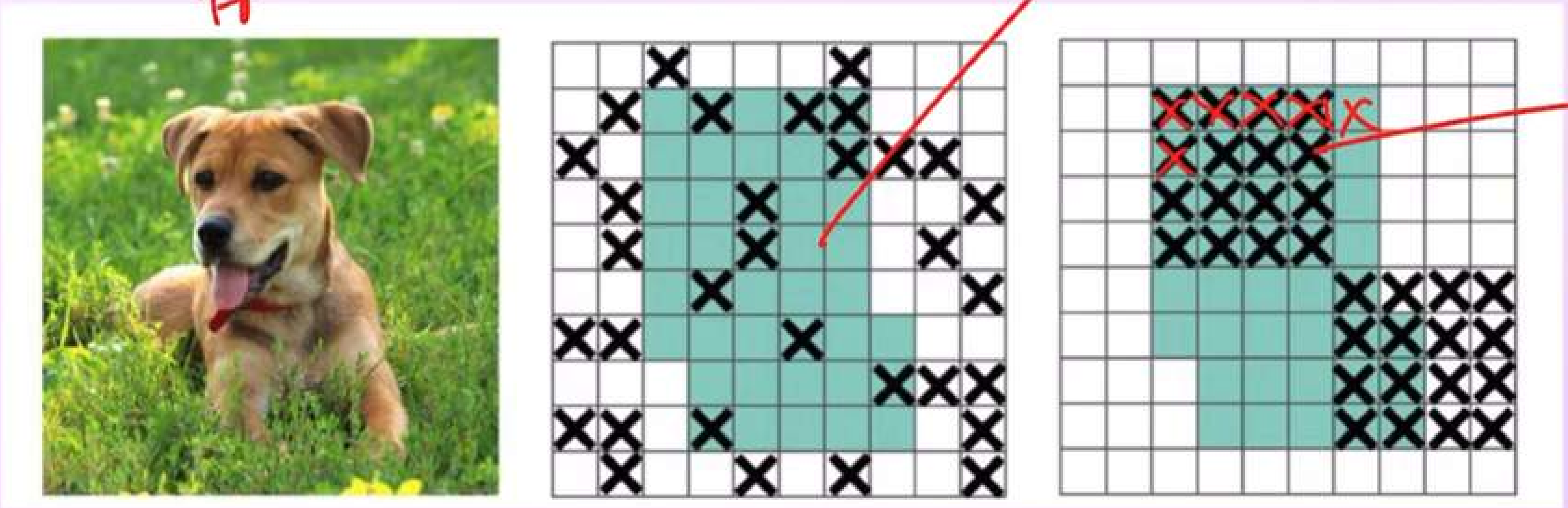
Activations

P=0.1

10%

i/p

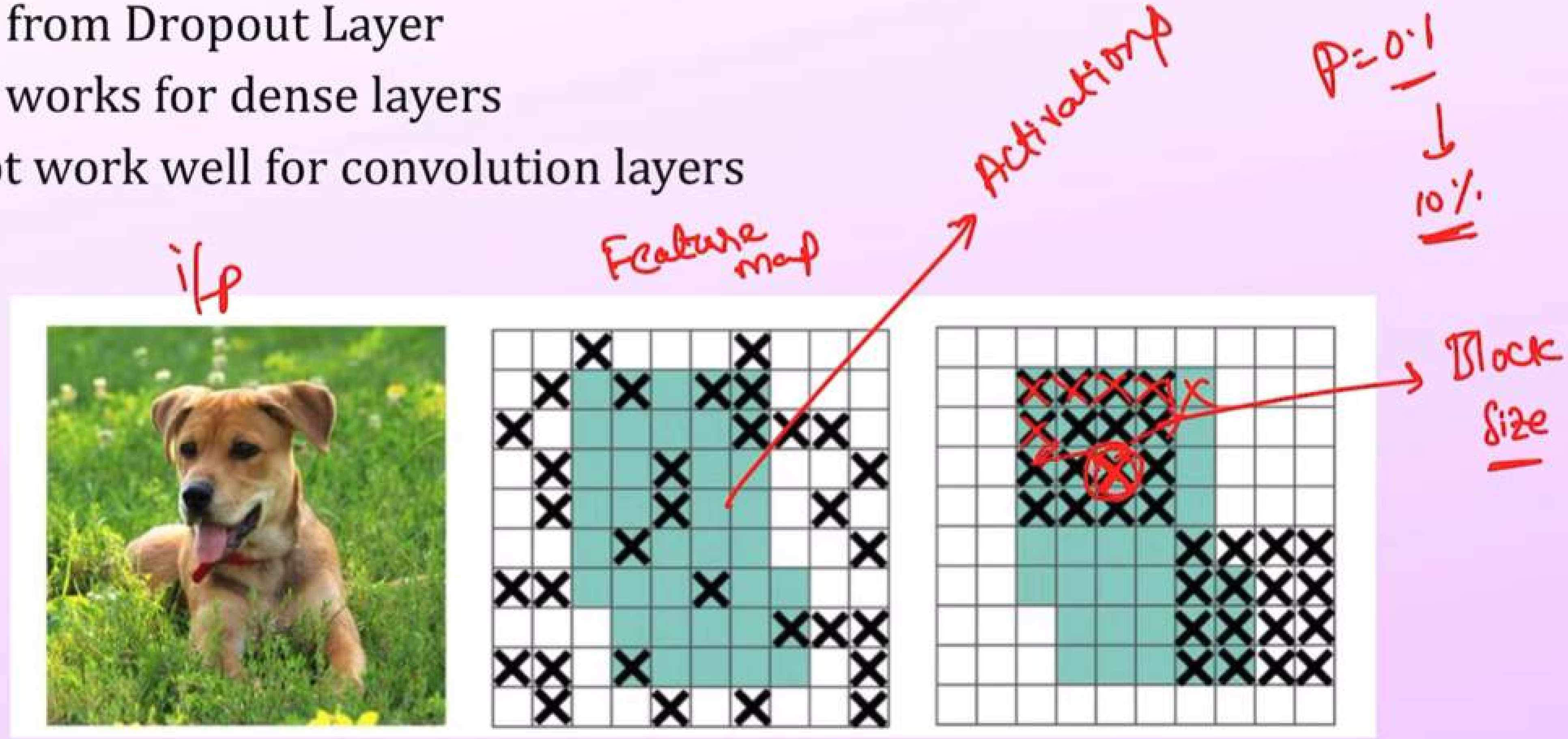
Feature map



Block size

Drop Block Regularization

- Inspired from Dropout Layer
- Dropout works for dense layers
- Might not work well for convolution layers

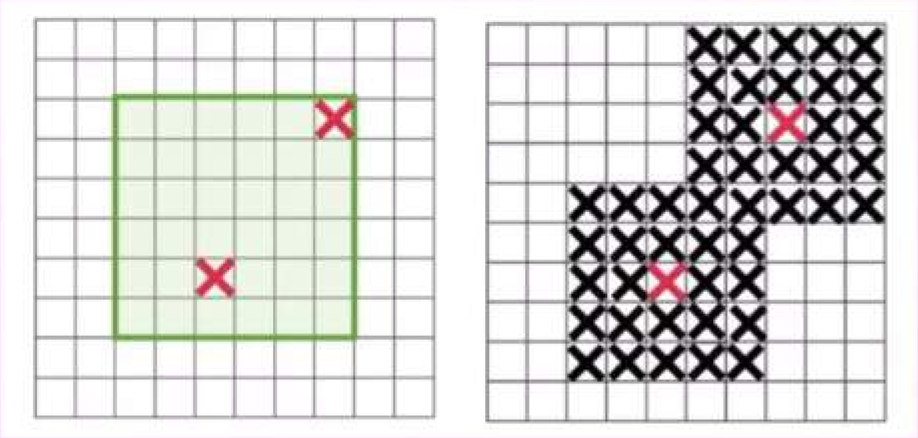


Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

Input:output activations of a layer (A), *block_size*, γ, *mode*
if *mode* == *Inference* **then**
  return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times \mathbf{count}(M)/\mathbf{count\_ones}(M)$

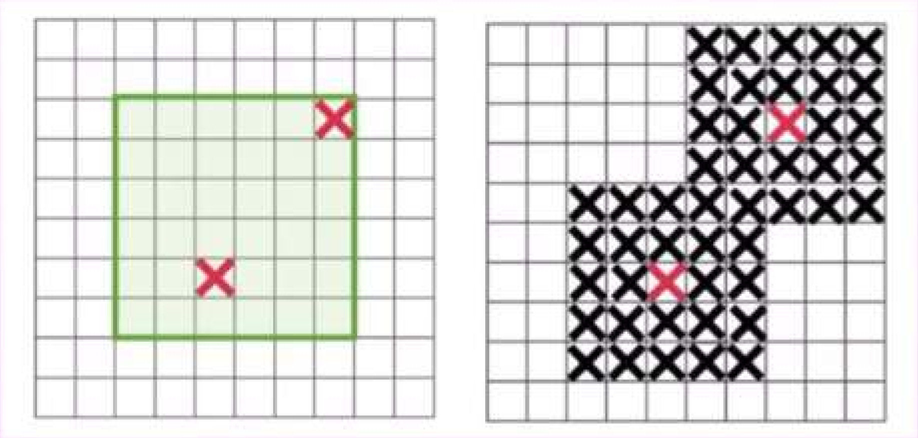


Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

```
Input:output activations of a layer (A), block_size, γ, mode
if mode == Inference then
    return A
end if
Randomly sample mask M: M_{i,j} ~ Bernoulli(γ)
For each zero position M_{i,j}, create a spatial square mask with the center being M_{i,j}, the width,
height being block_size and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: A = A × M
Normalize the features: A = A × count(M)/count_ones(M)
```

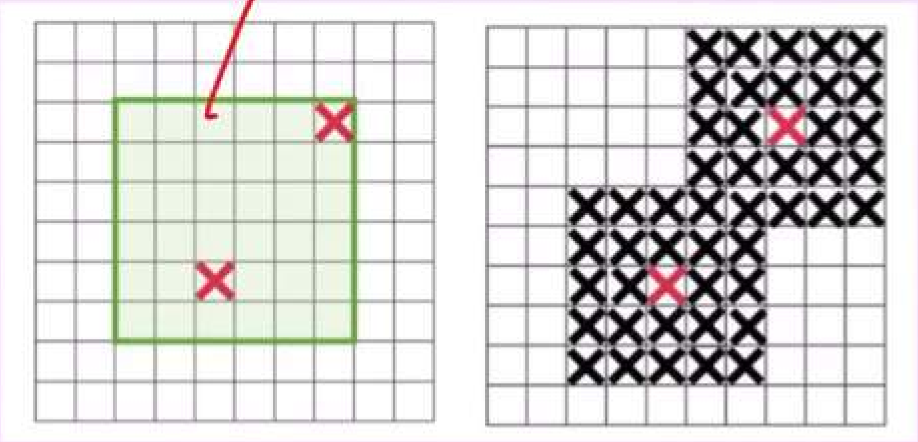


Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

Input:output activations of a layer (A), *block_size*, γ, *mode*
if $mode == Inference$ **then**
    return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times \mathbf{count}(M)/\mathbf{count\_ones}(M)$

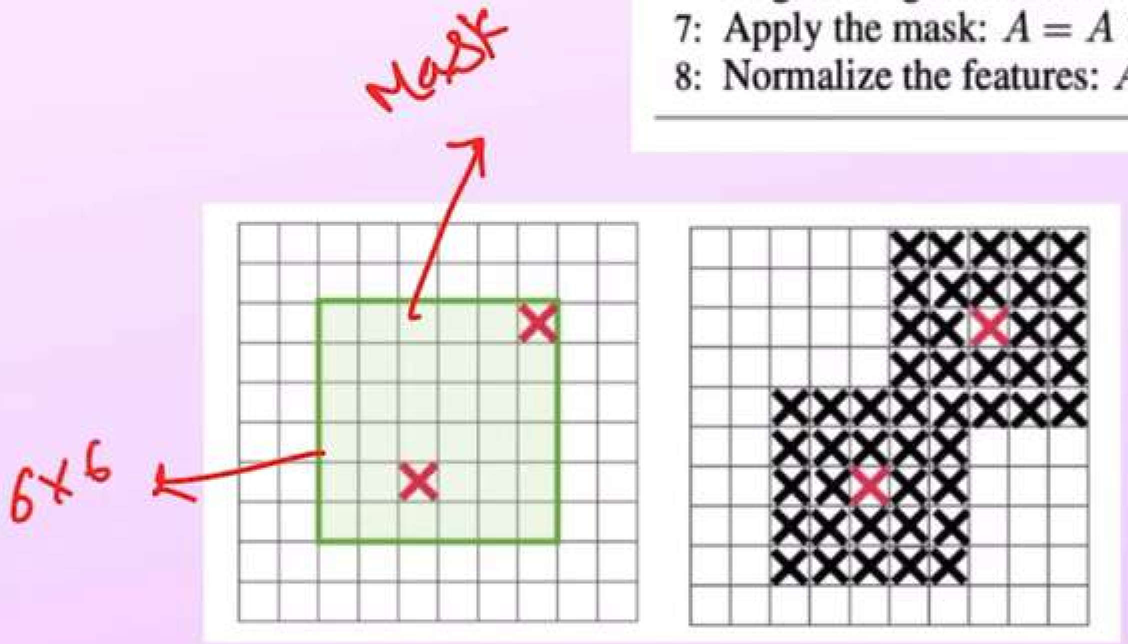


Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

Input:output activations of a layer (A), *block_size*, γ, *mode*
if *mode* == *Inference* **then**
    return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times \mathbf{count}(M)/\mathbf{count\_ones}(M)$



Drop Block Regularization

- Block_size
- γ - Probability of Keeping

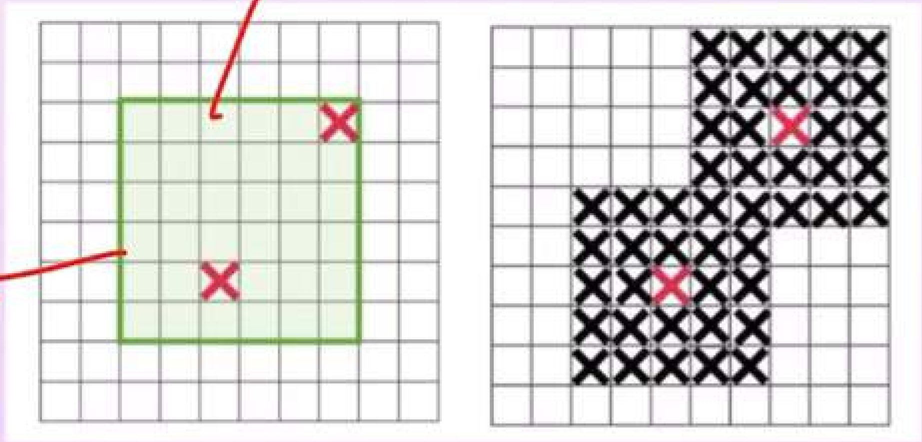
Algorithm 1 DropBlock

Input:output activations of a layer (A), *block_size*, γ, *mode*
if $mode$ == *Inference* **then**
    return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times \textbf{count}(M)/\textbf{count\_ones}(M)$

P
0.25

Mask

6x6



Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

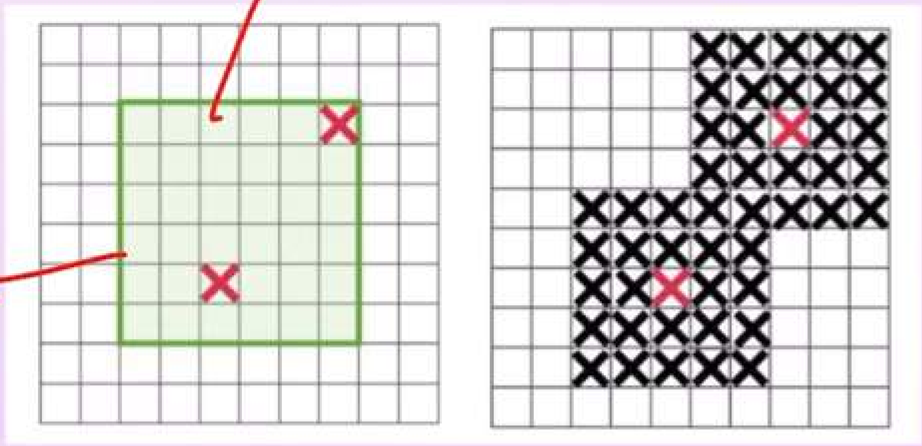
Input:output activations of a layer (A), *block_size*, γ, *mode*
if *mode* == *Inference* **then**
    return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times \mathbf{count}(M)/\mathbf{count\_ones}(M)$

P
0.25

Mask

.9

6x6



Drop Block Regularization

- Block_size
- γ - Probability

  of Keeping

Algorithm 1 DropBlock

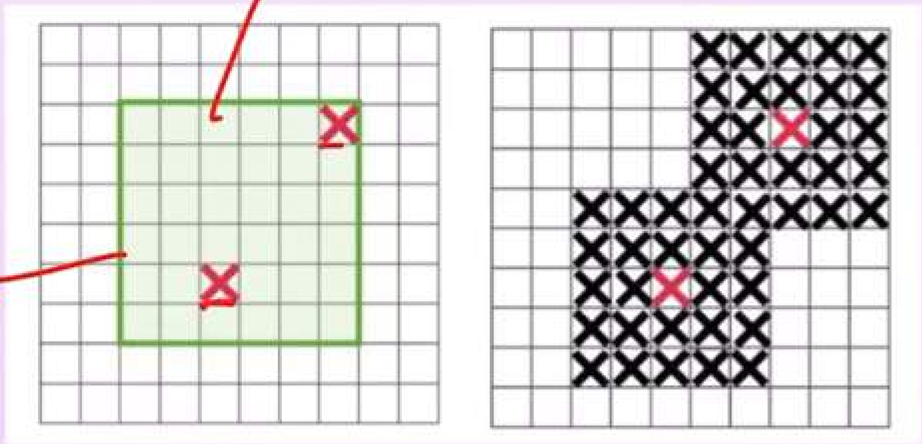
Input:output activations of a layer (A), *block_size*, γ, *mode*
if *mode* == *Inference* **then**
  return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times \mathbf{count}(M)/\mathbf{count\_ones}(M)$

p
0.25

Mask

9

6x6



Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

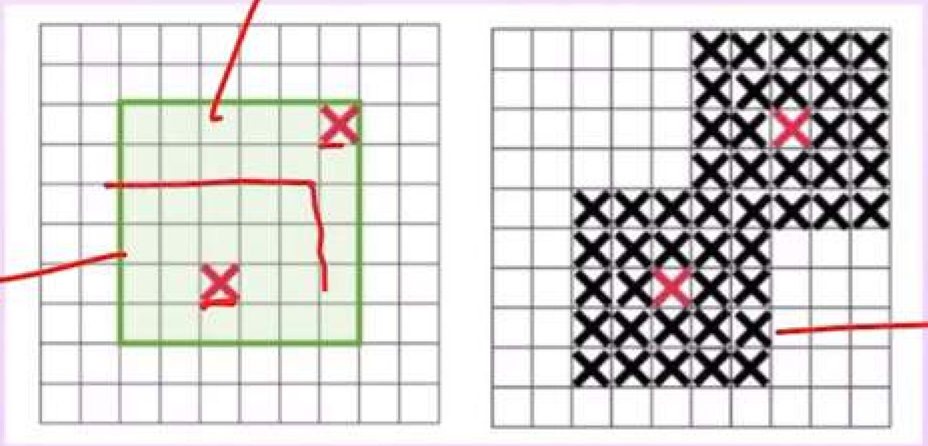
Input:output activations of a layer (A), *block_size*, γ, *mode*
if *mode* == *Inference* **then**
    return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times \textbf{count}(M)/\textbf{count\_ones}(M)$

P
0.25

.9

6x6

Mask



5x5

Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

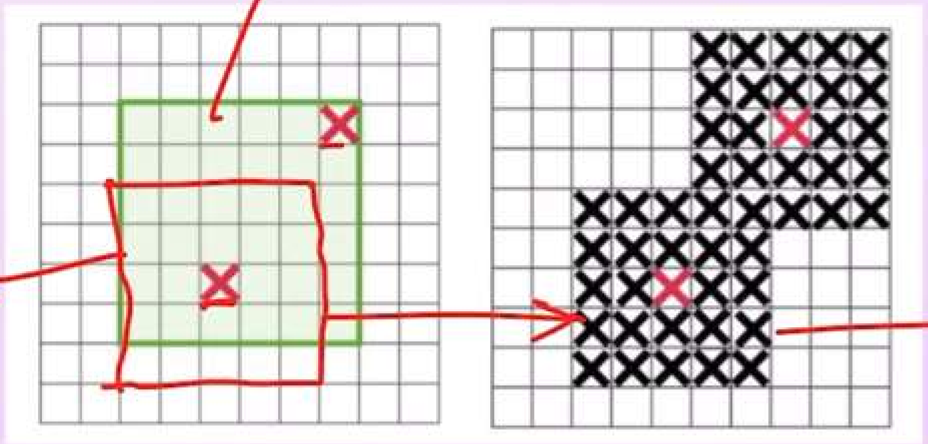
Input:output activations of a layer (A), *block_size*, γ, *mode*
if $mode$ == *Inference* **then**
  return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times \mathbf{count}(M)/\mathbf{count\_ones}(M)$

P
0.25

Mask

9

6x6



5x5

Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

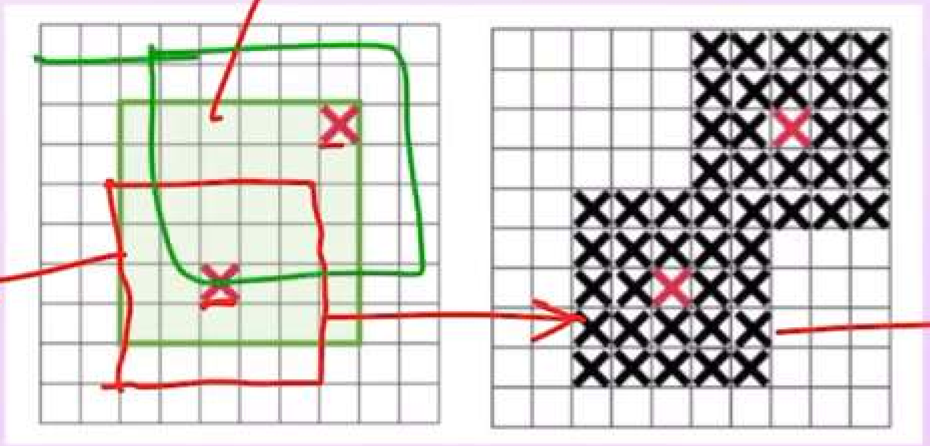
Input: output activations of a layer (A), *block_size*, γ, *mode*
if $mode == Inference$ **then**
    return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times \textbf{count}(M)/\textbf{count\_ones}(M)$

P
0.25

9

6x6

Mask



5x5

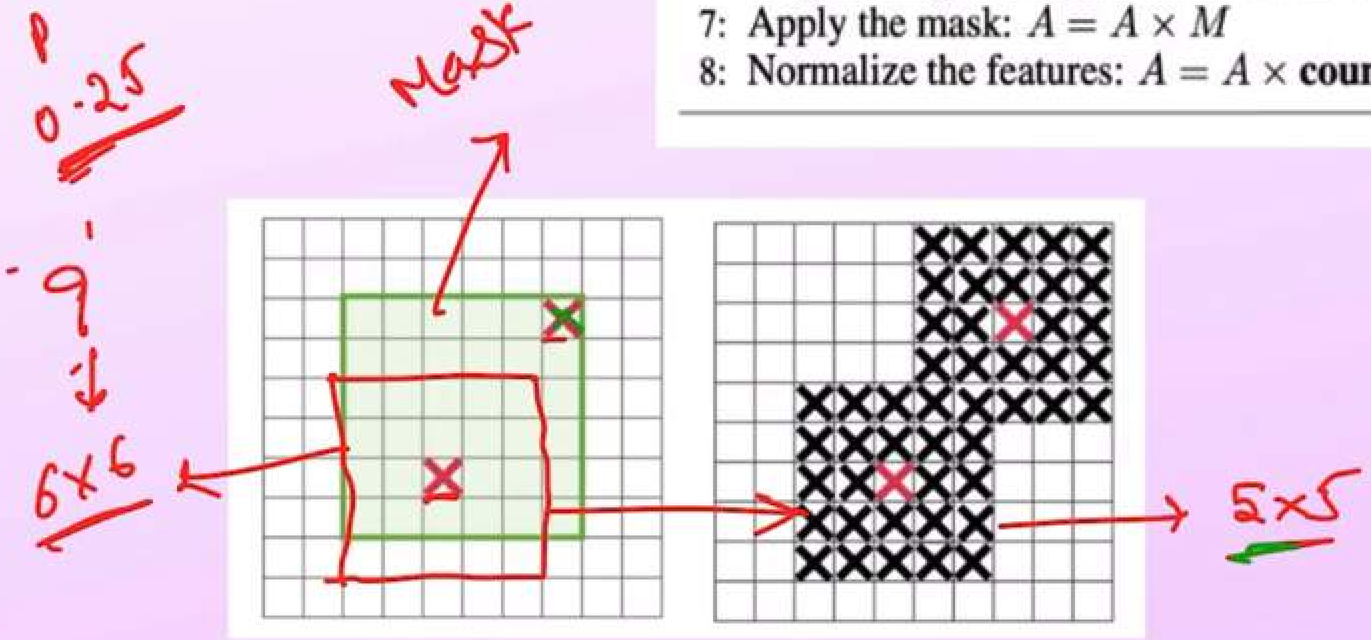
Drop Block Regularization

- Block_size
- γ - Probability

  of Keeping

Algorithm 1 DropBlock

Input: output activations of a layer (A), *block_size*, γ, *mode*
if *mode* == *Inference* **then**
    return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times$ **count**(M)/**count_ones**(M)



Drop Block Regularization

- Block_size
- γ - Probability

  of Keeping

Algorithm 1 DropBlock

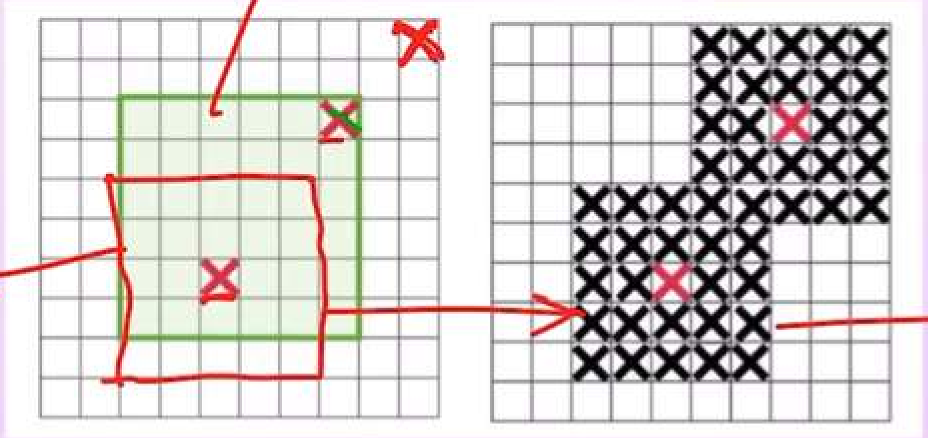
Input:output activations of a layer (A), *block_size*, γ, *mode*
if *mode* == *Inference* **then**
    return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times$ **count**(M)/**count_ones**(M)

P
0.25

Mask

.9

6x6



5x5

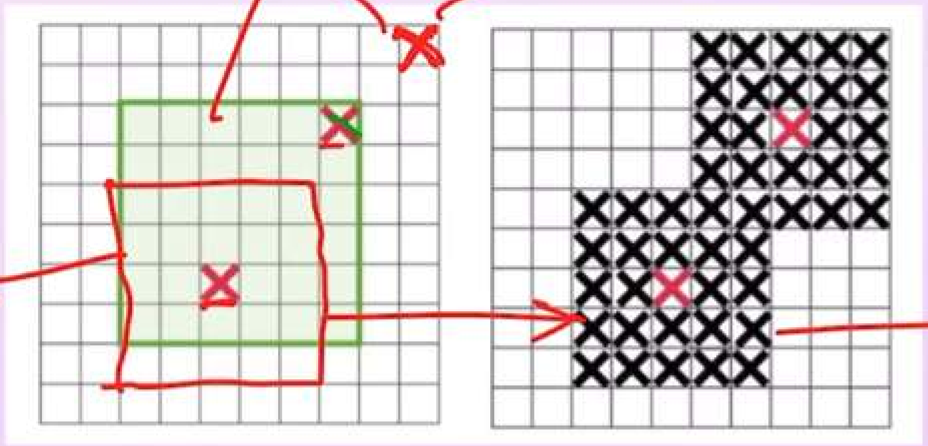
Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

1: **Input:**output activations of a layer (A), *block_size*, γ, *mode*
2: **if** *mode* == *Inference* **then**
3: return A
4: **end if**
5: Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
6: For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
7: Apply the mask: $A = A \times M$
8: Normalize the features: $A = A \times$ **count**(M)/**count_ones**(M)

P
0.25
.9
Mask
6x6



5x5

Drop Block Regularization

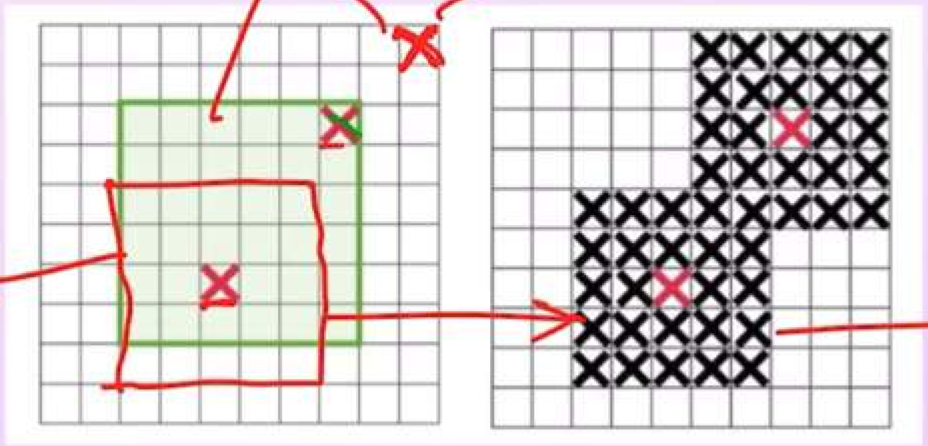
- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

```
Input: output activations of a layer (A), block_size, γ, mode
if mode == Inference then
    return A
end if
Randomly sample mask M: M_{i,j} ~ Bernoulli(γ)
For each zero position M_{i,j}, create a spatial square mask with the center being M_{i,j}, the width,
    height being block_size and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: A = A × M
Normalize the features: A = A × count(M)/count_ones(M)
```

P 0.25

9

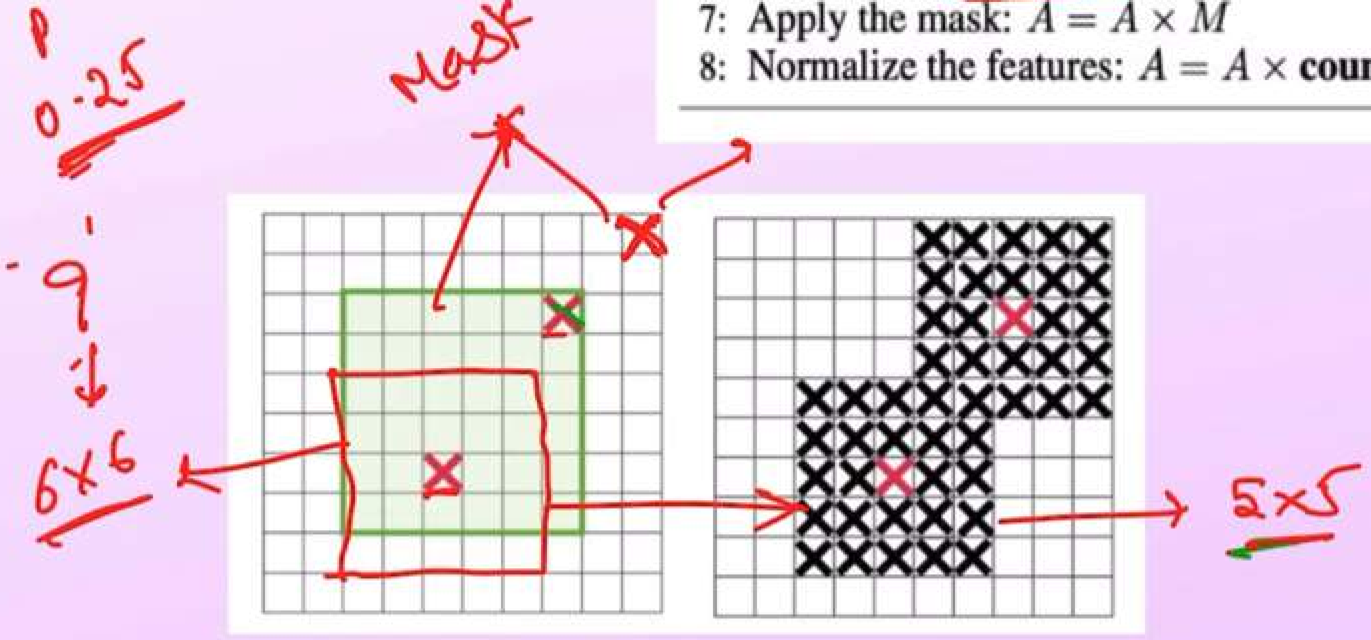


Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

Input:output activations of a layer (A), *block_size*, γ, *mode*
if *mode* == *Inference* **then**
  return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times$ **count**(M)/**count_ones**(M)



Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

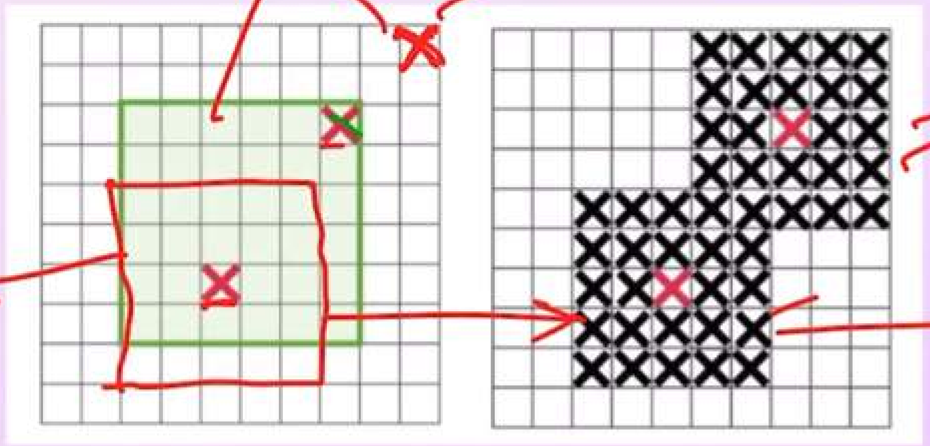
```
Input: output activations of a layer (A), block_size, γ, mode
if mode == Inference then
    return A
end if
Randomly sample mask M: M_{i,j} ~ Bernoulli(γ)
For each zero position M_{i,j}, create a spatial square mask with the center being M_{i,j}, the width,
    height being block_size and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: A = A × M
Normalize the features: A = A × count(M)/count_ones(M)
```

p
0.25

Mask

9

6x6



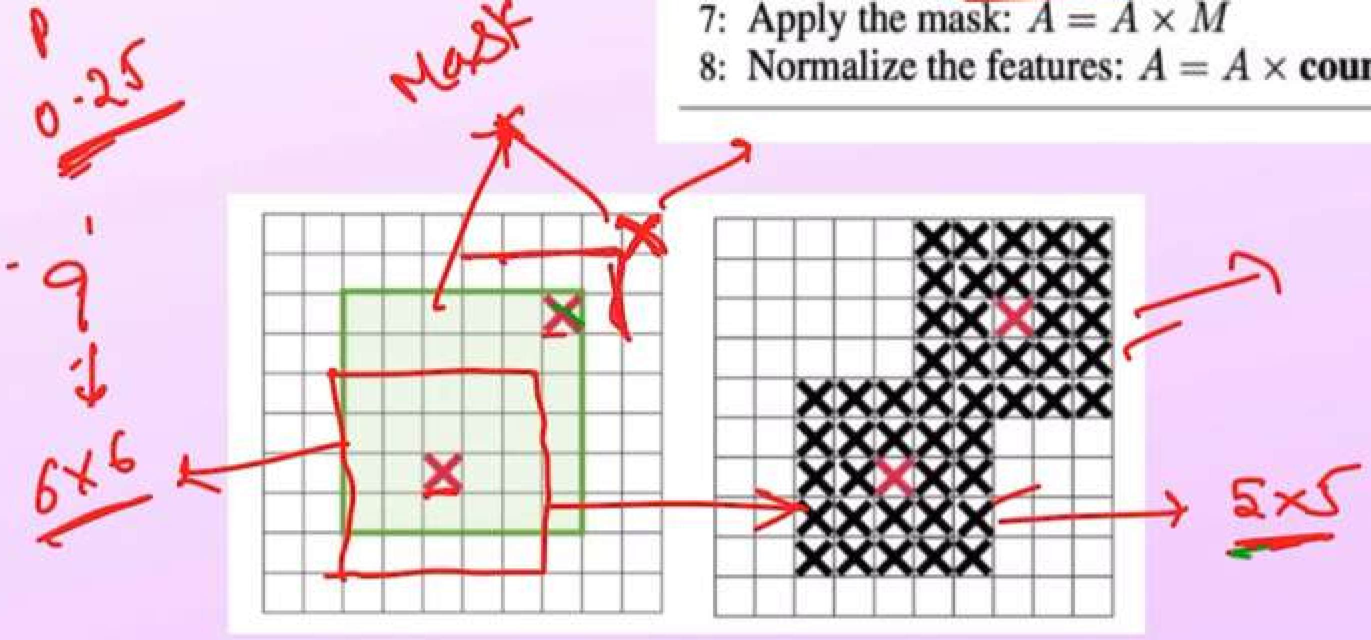
5x5

Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

Input: output activations of a layer (A), *block_size*, γ, *mode*
if *mode* == *Inference* **then**
  return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times \mathbf{count}(M)/\mathbf{count\_ones}(M)$



Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

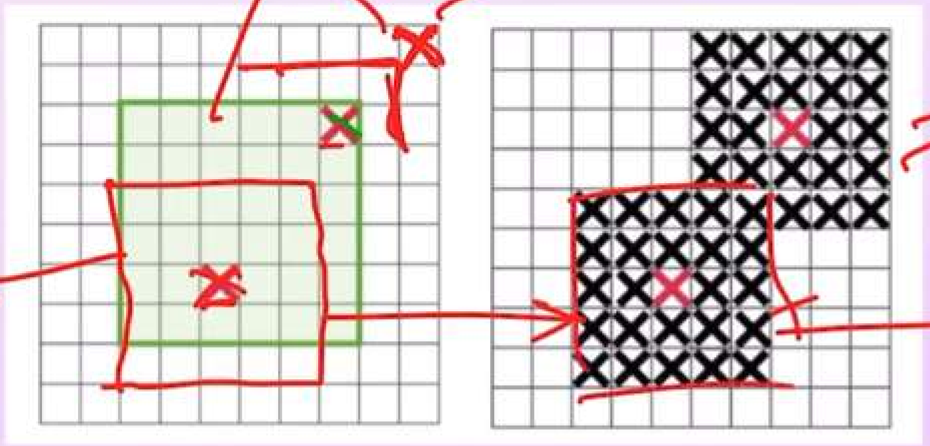
```
Input:output activations of a layer (A), block_size, γ, mode
if mode == Inference then
    return A
end if
Randomly sample mask M: M_{i,j} ~ Bernoulli(γ)
For each zero position M_{i,j}, create a spatial square mask with the center being M_{i,j}, the width,
height being block_size and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: A = A × M
Normalize the features: A = A × count(M)/count_ones(M)
```

P
0.25

9

6x6

Mask



5x5

Drop Block Regularization

- Block_size
- γ - Probability of Keeping

Algorithm 1 DropBlock

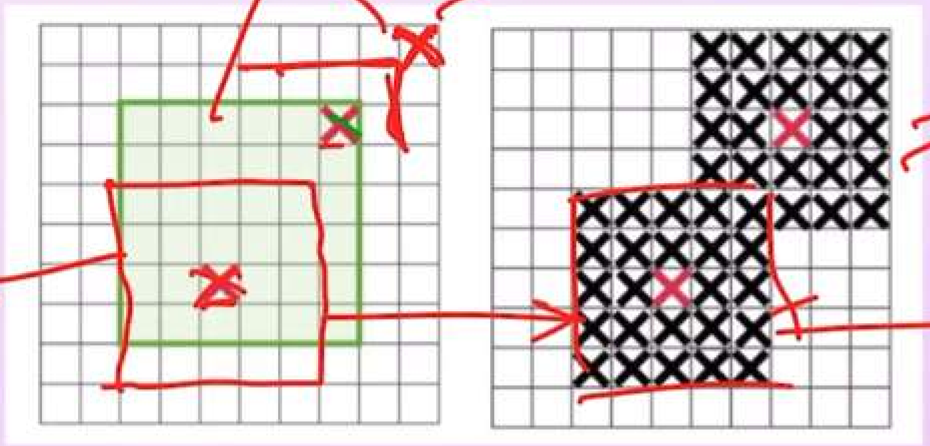
Input: output activations of a layer (A), *block_size*, γ, *mode*
if $mode$ == *Inference* **then**
    return A
end if
Randomly sample mask M: $M_{i,j} \sim Bernoulli(\gamma)$
For each zero position $M_{i,j}$, create a spatial square mask with the center being $M_{i,j}$, the width, height being *block_size* and set all the values of M in the square to be zero (see Figure 2).
Apply the mask: $A = A \times M$
Normalize the features: $A = A \times \mathbf{count}(M)/\mathbf{count\_ones}(M)$

p 0.25

Mask

9

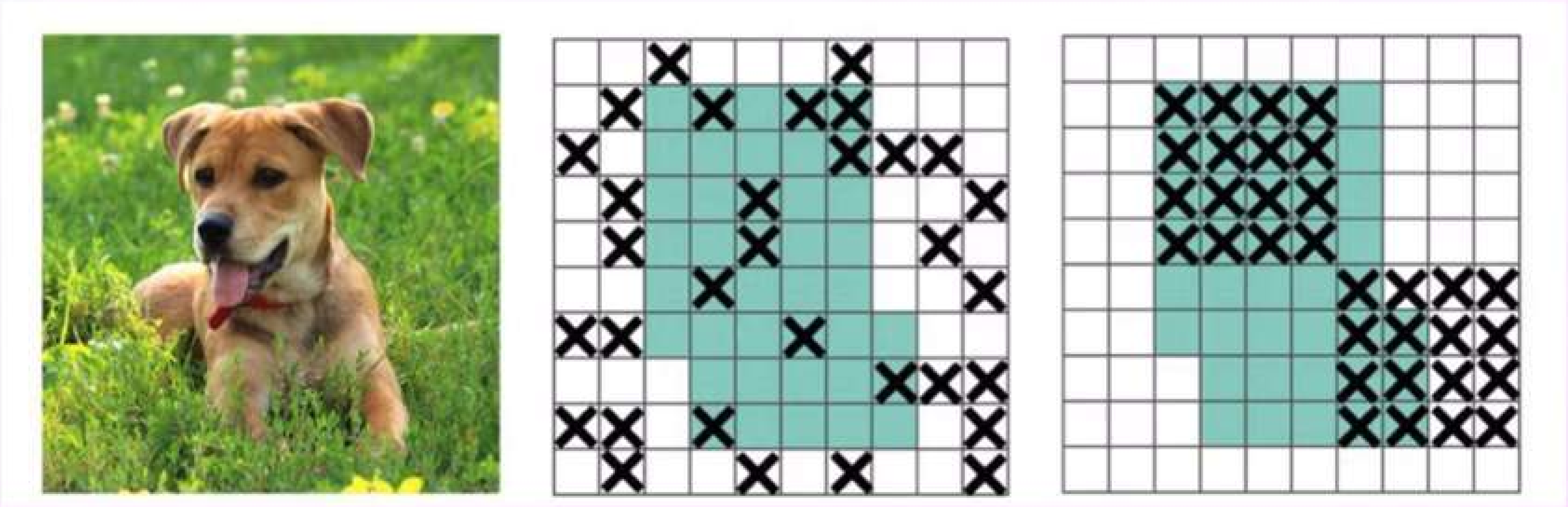
6x6



5x5

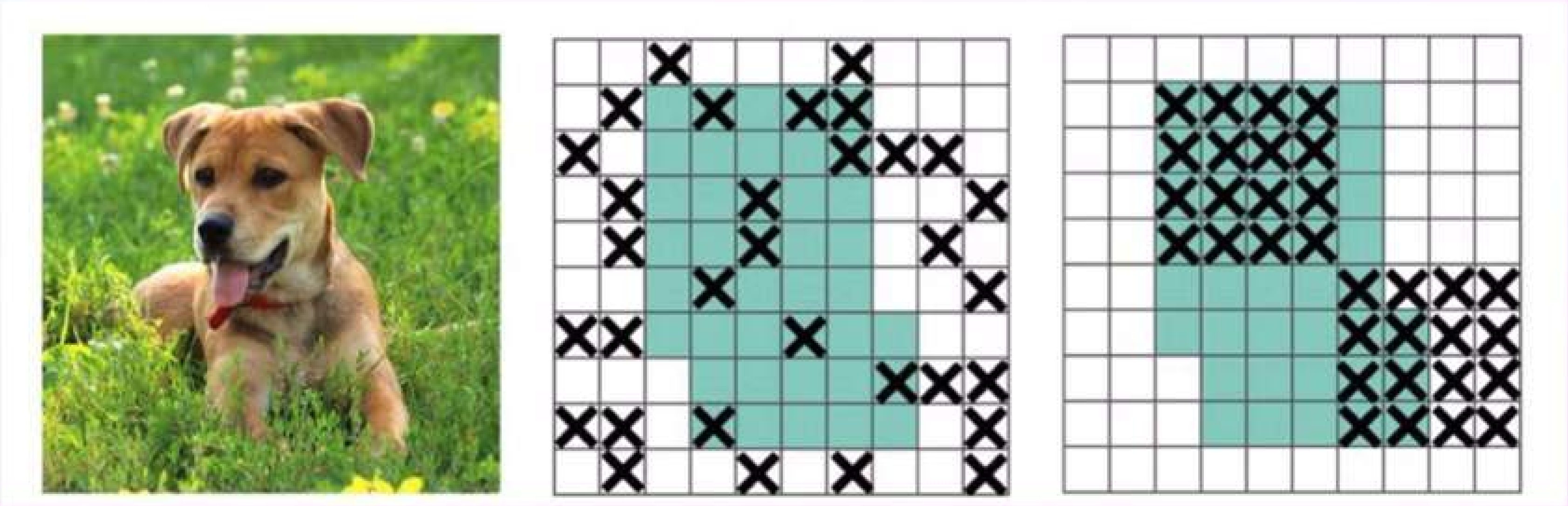
Drop Block Regularization

- ResNet-50 (Imagenet) - Improved accuracy from 76.51% to 78.13%
- RetinaNet (COCO) - 36.8% to 38.4%



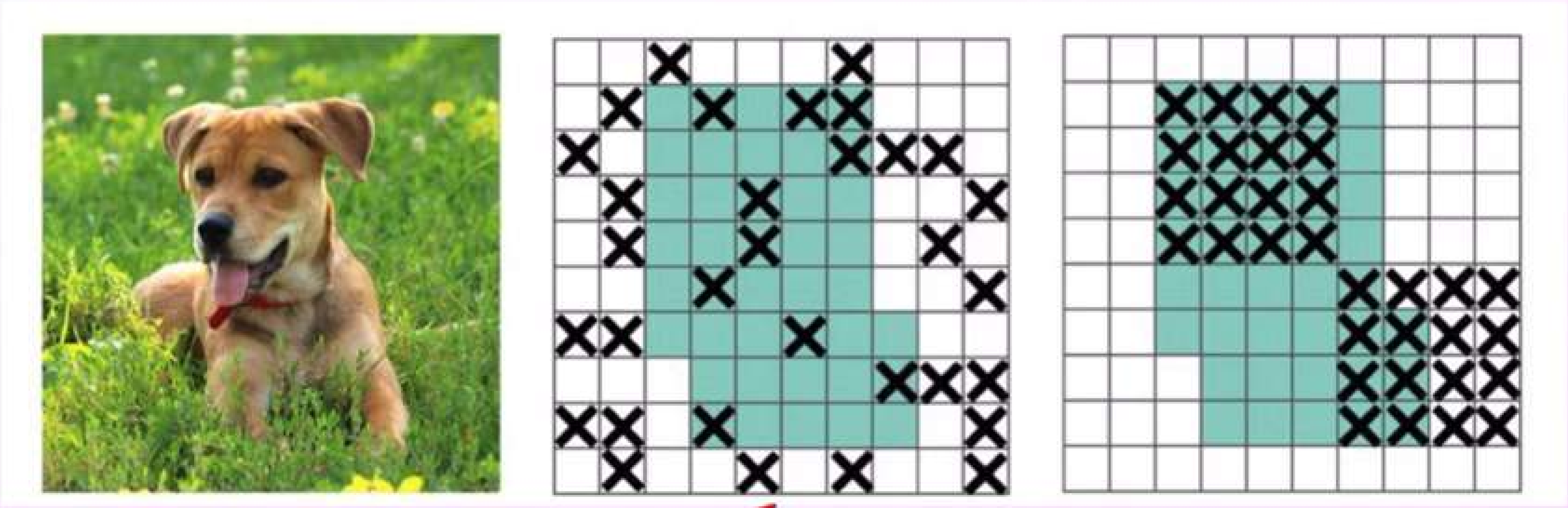
Drop Block Regularization

- ResNet-50 (Imagenet) - Improved accuracy from 76.51% to 78.13%
- RetinaNet (COCO) - 36.8% to 38.4%



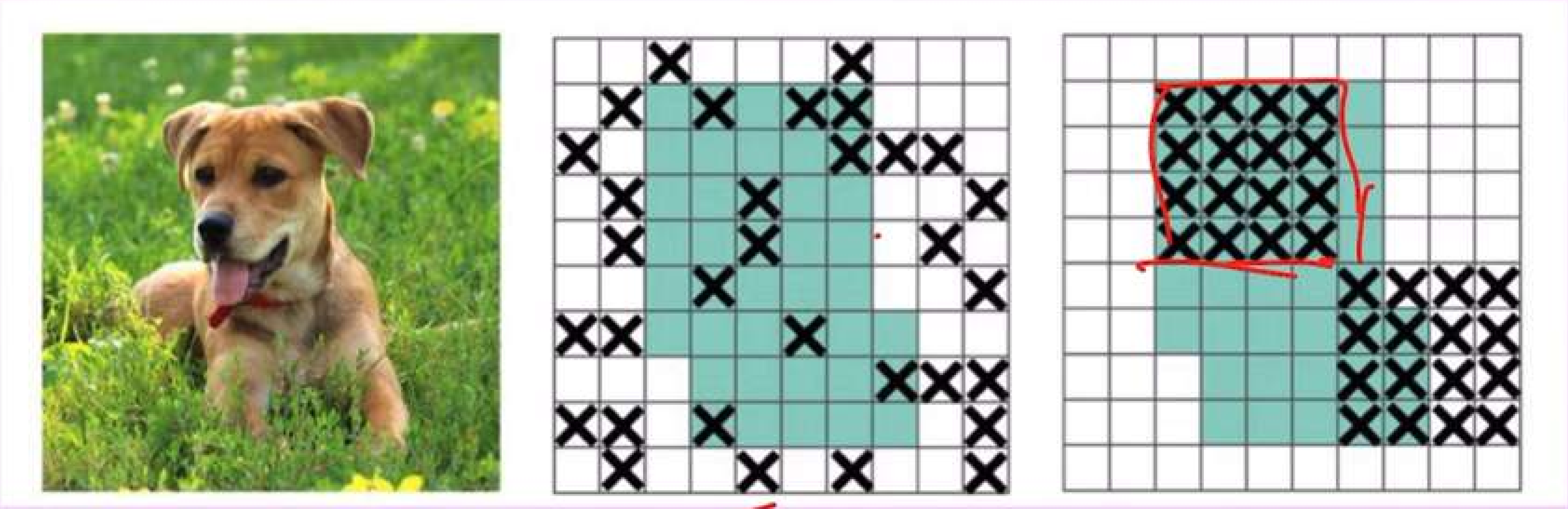
Drop Block Regularization

- ResNet-50 (Imagenet) - Improved accuracy from 76.51% to 78.13%
- RetinaNet (COCO) - 36.8% to 38.4%



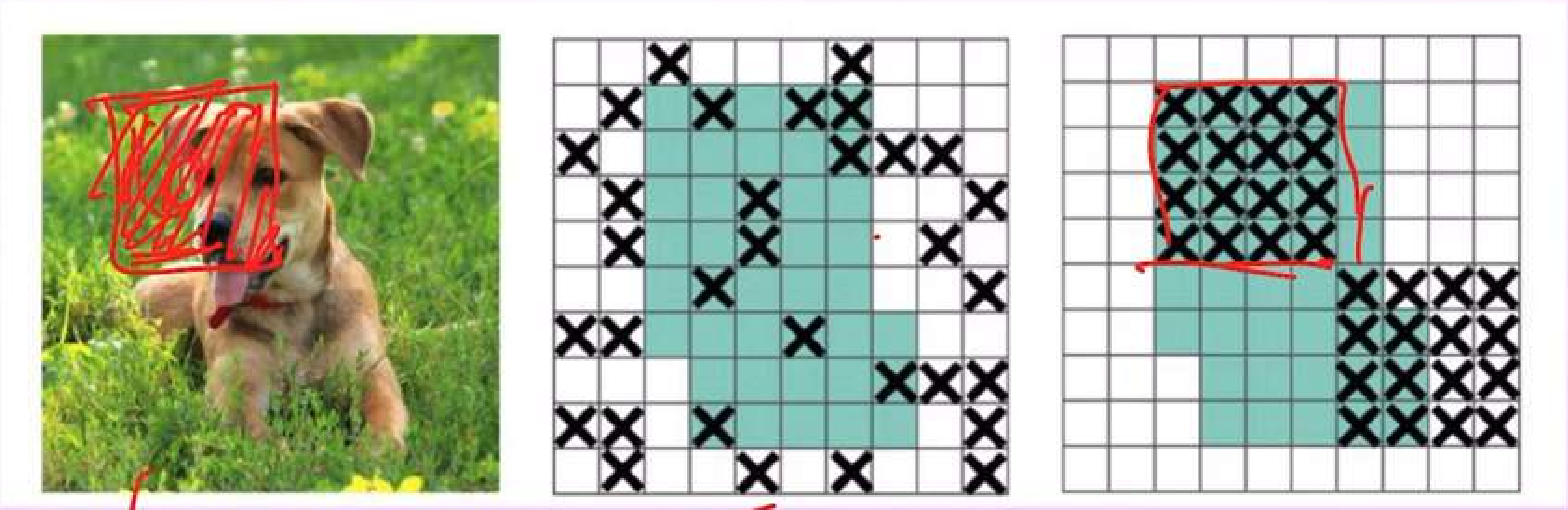
Drop Block Regularization

- ResNet-50 (Imagenet) - Improved accuracy from 76.51% to 78.13%
- RetinaNet (COCO) - 36.8% to 38.4%



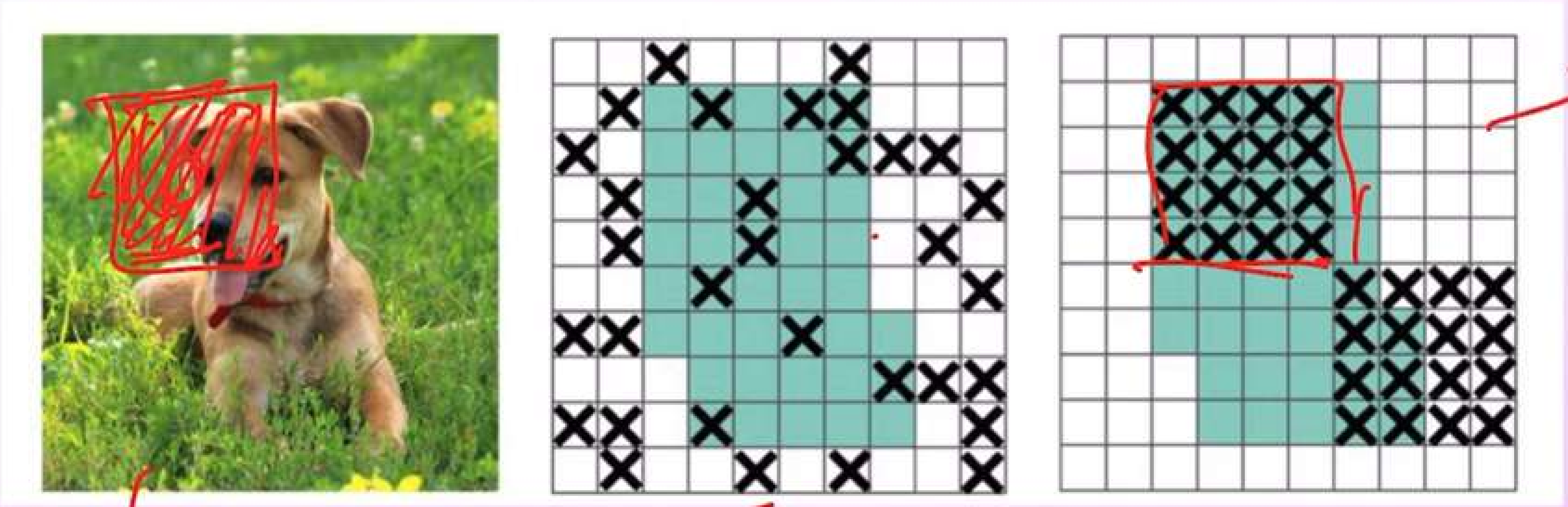
Drop Block Regularization

- ResNet-50 (Imagenet) - Improved accuracy from 76.51% to 78.13%
- RetinaNet (COCO) - 36.8% to 38.4%

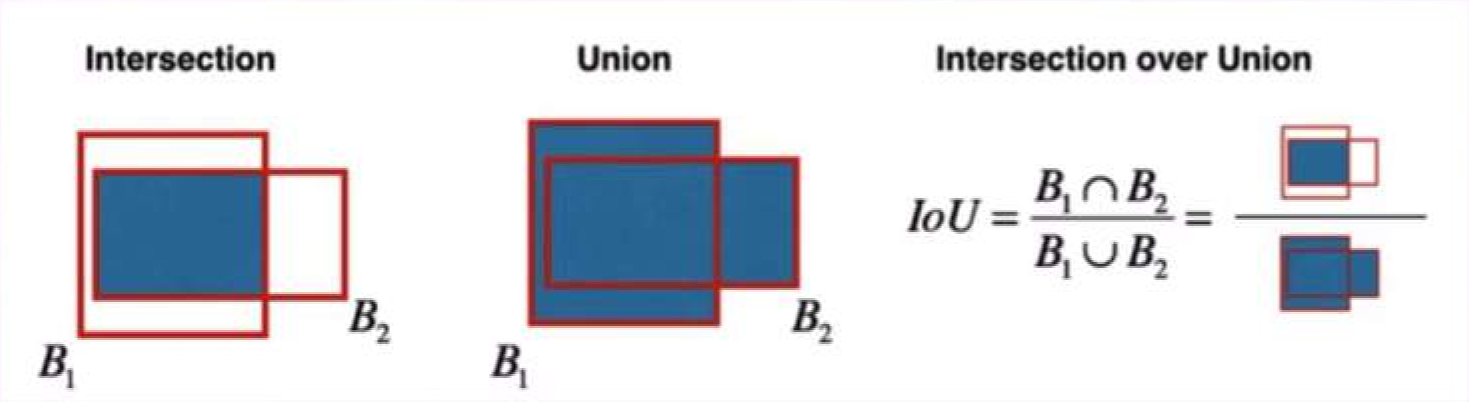


Drop Block Regularization

- ResNet-50 (Imagenet) - Improved accuracy from 76.51% to 78.13%
- RetinaNet (COCO) - 36.8% to 38.4%



IOU Loss



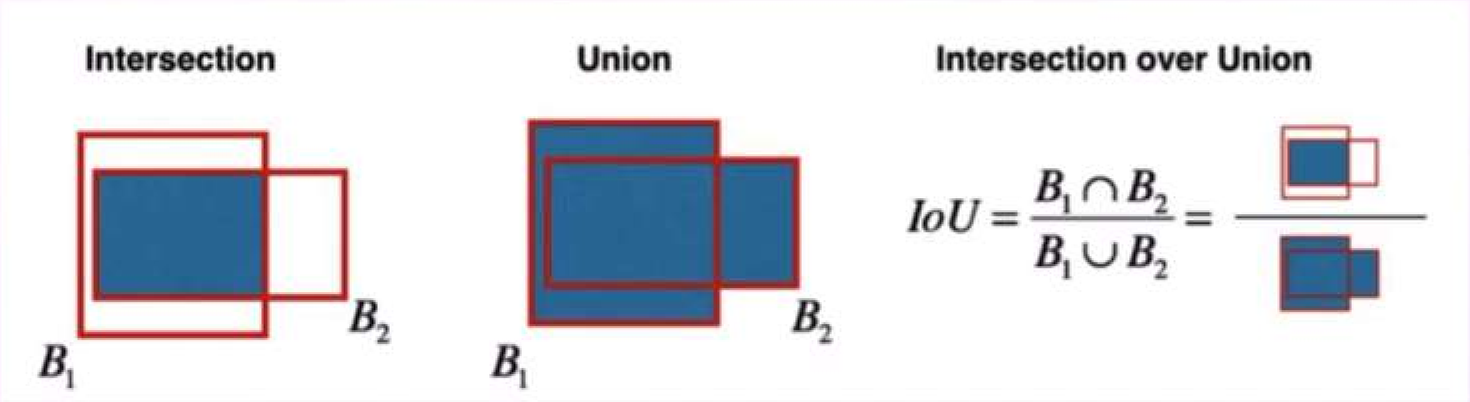
$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

IOU Loss



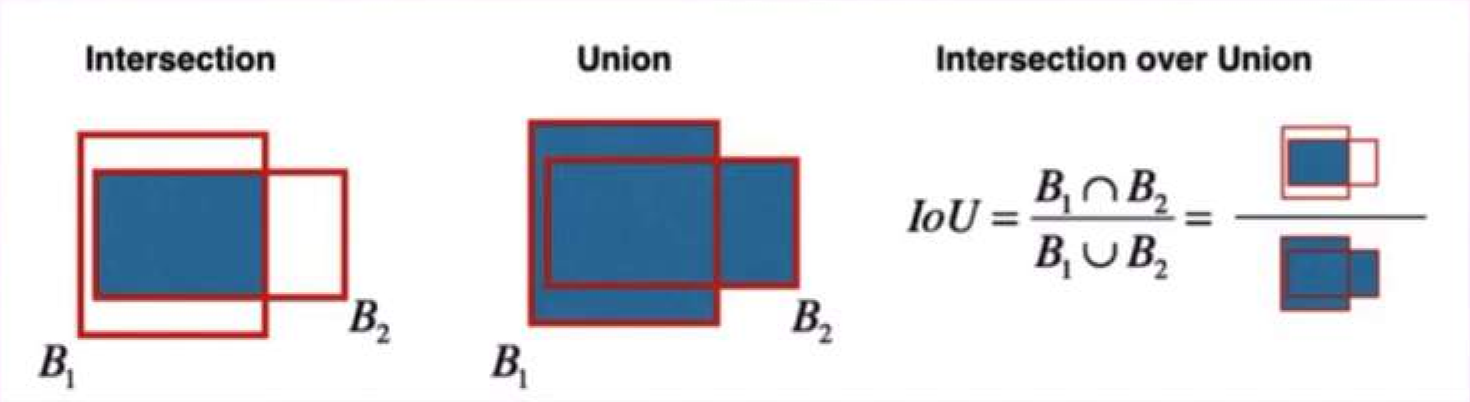
$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt}) \text{ is the ground-truth}$$

$$B = (x, y, w, h) \text{ is the predicted box}$$

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

IOU Loss



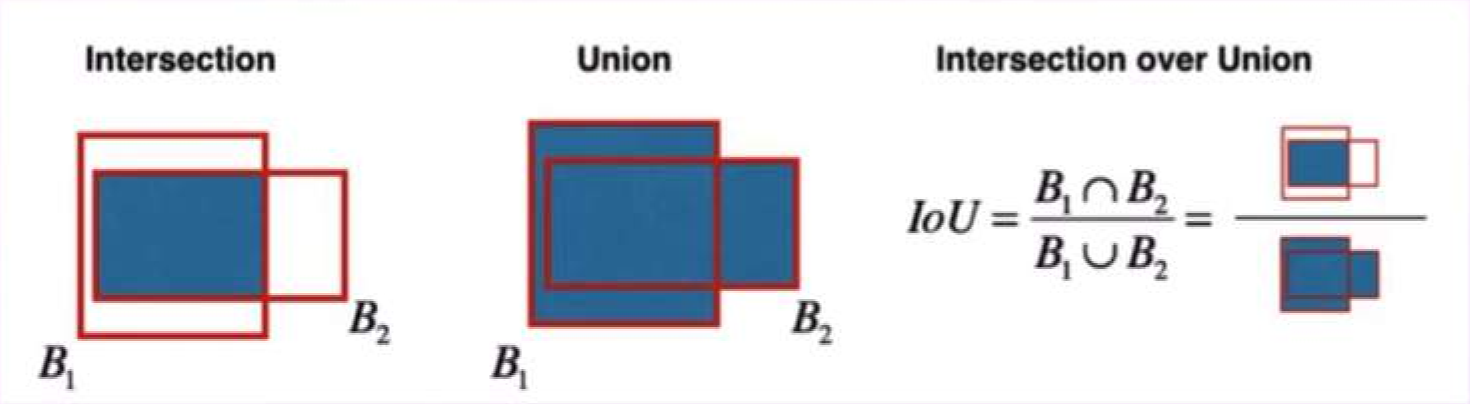
$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

IOU Loss



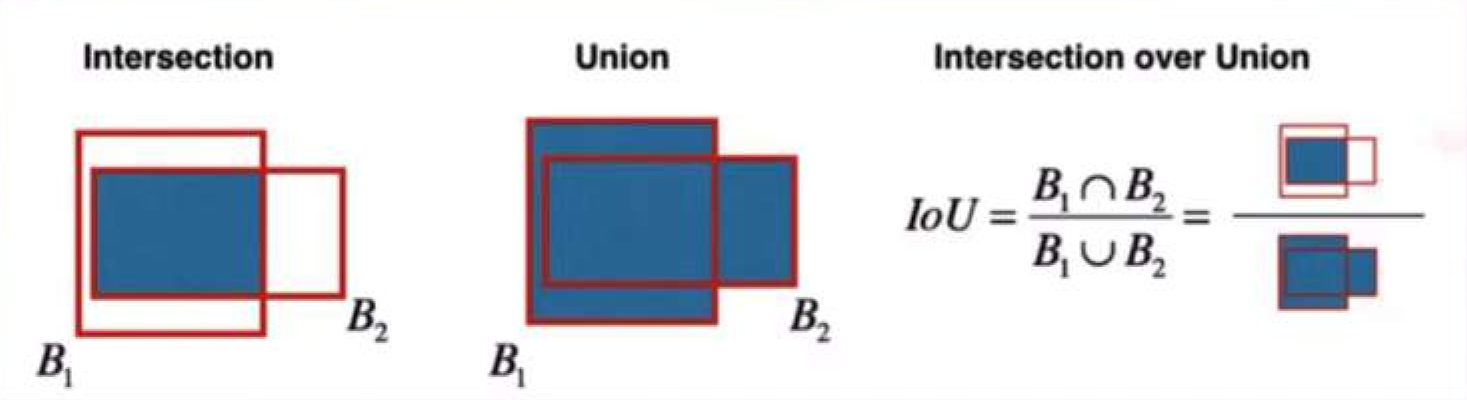
$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

IOU Loss



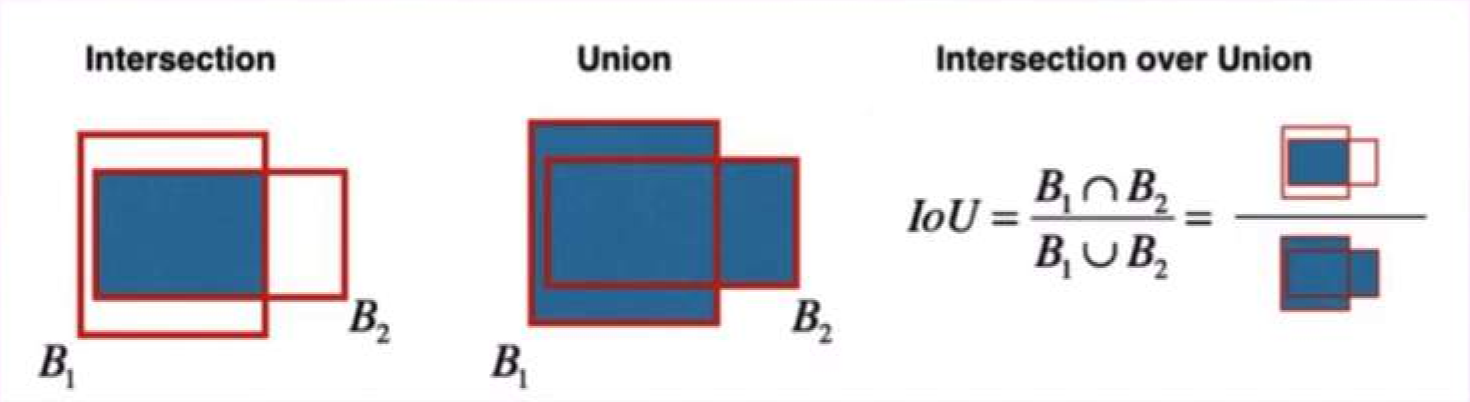
$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

IOU Loss



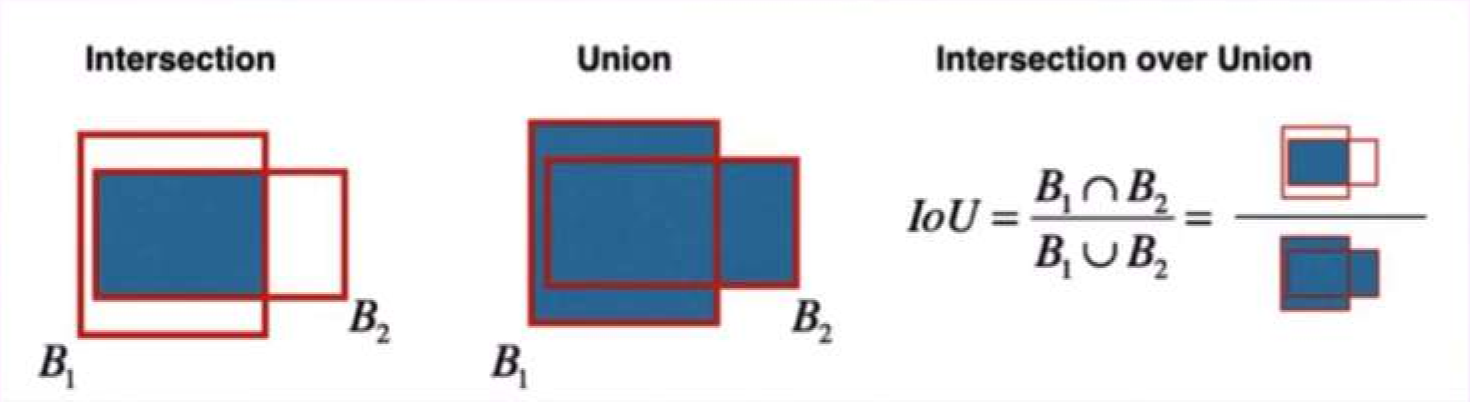
$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

IOU Loss



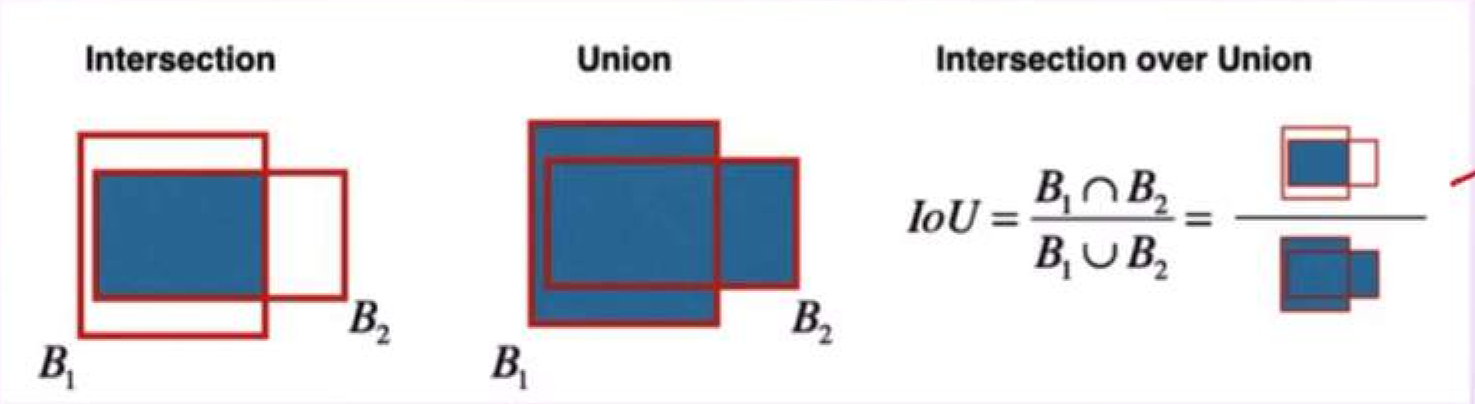
$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

IOU Loss



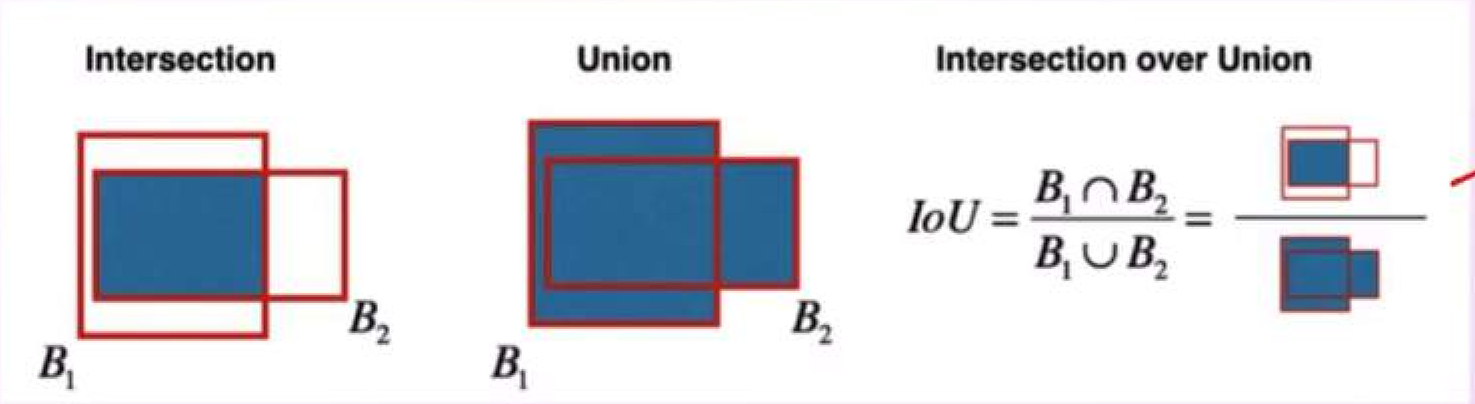
$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{\underline{B \cup B^{gt}}} \rightarrow 1 - \underline{IOU}$$

IOU Loss



1 → perfect overlap

0 → No overlap

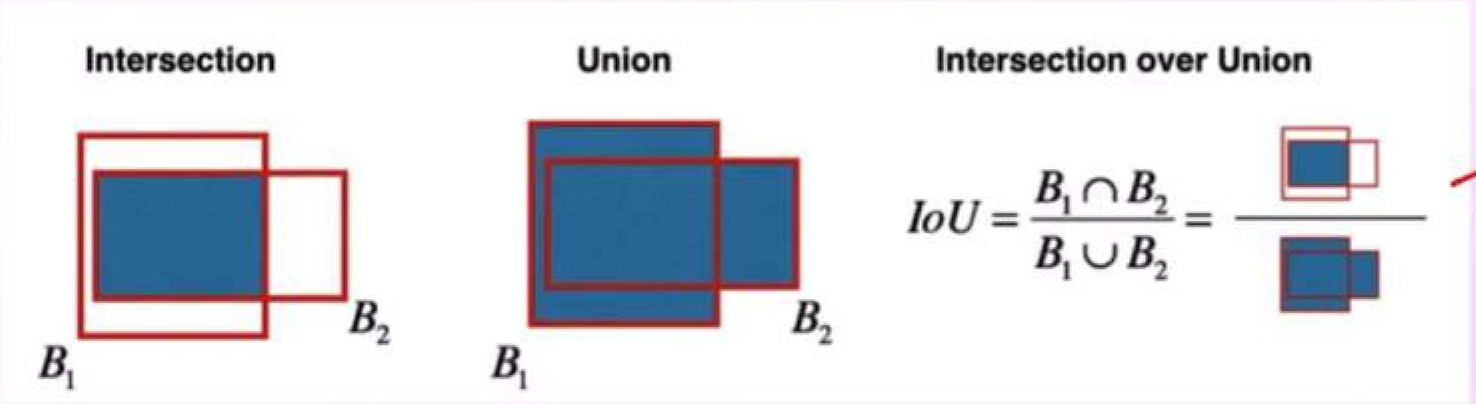
$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}} \rightarrow 1 - IOU$$

IOU Loss



1 → perfect overlap

0 → No overlap

$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt}) \text{ is the ground-truth}$$

$$B = (x, y, w, h) \text{ is the predicted box}$$

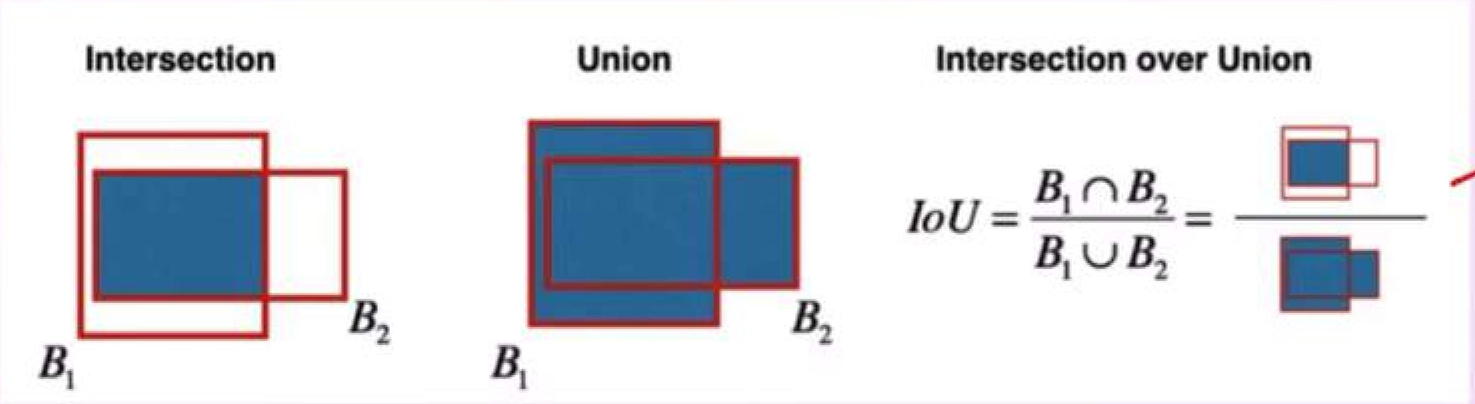
$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$ → 1 − IOU

IOU Loss

IOU ↑

1 → perfect overlap

0 → No overlap



$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

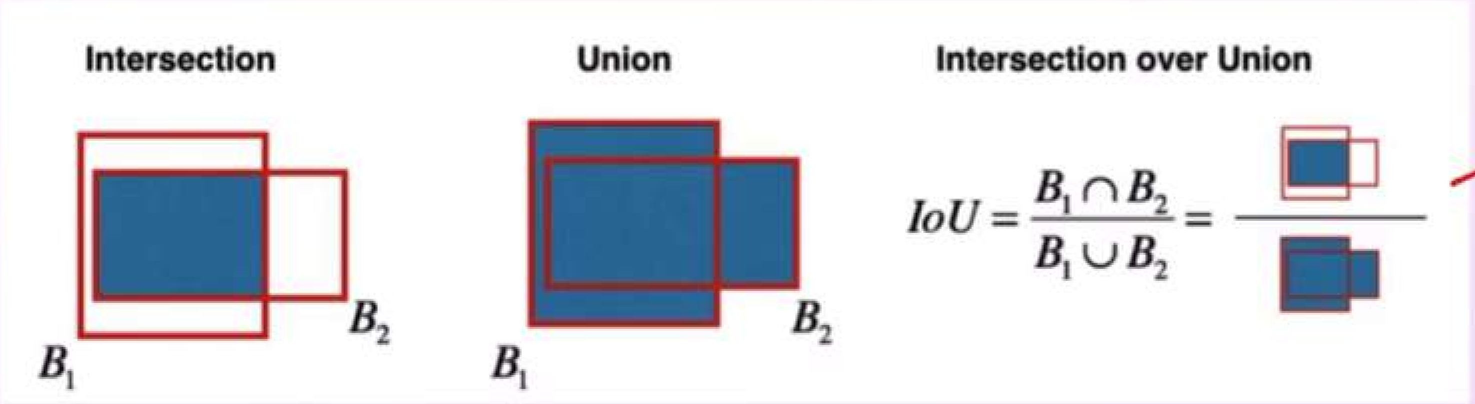
$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$ → 1 − IOU

IOU Loss

IOU ↑

1 → perfect overlap

0 → No overlap



Loss ↓

$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

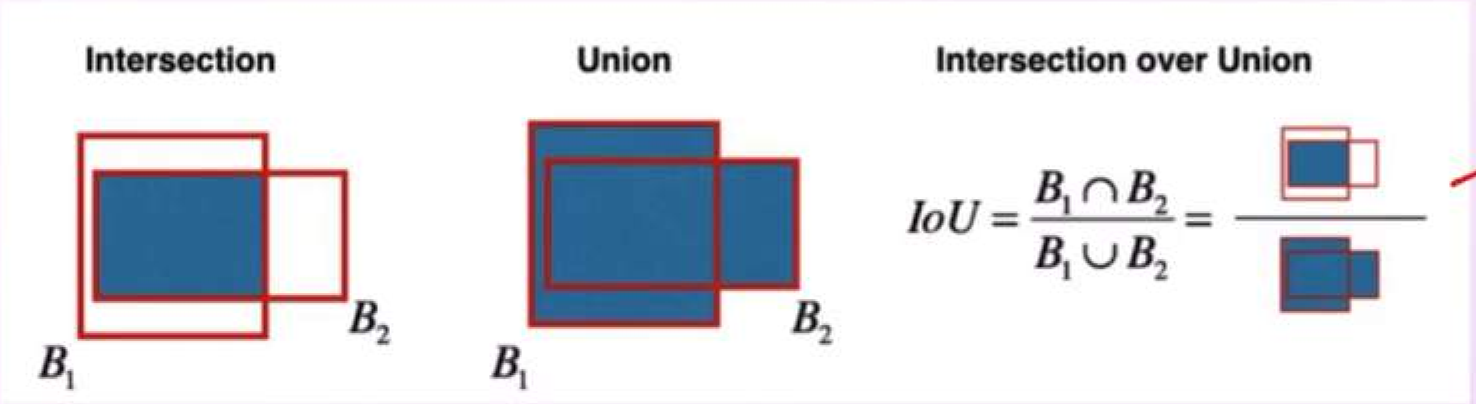
→ 1 − IOU → ↓

IOU Loss

IOU ↑

$1 \rightarrow$ perfect overlap

$0 \rightarrow$ No overlap



$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

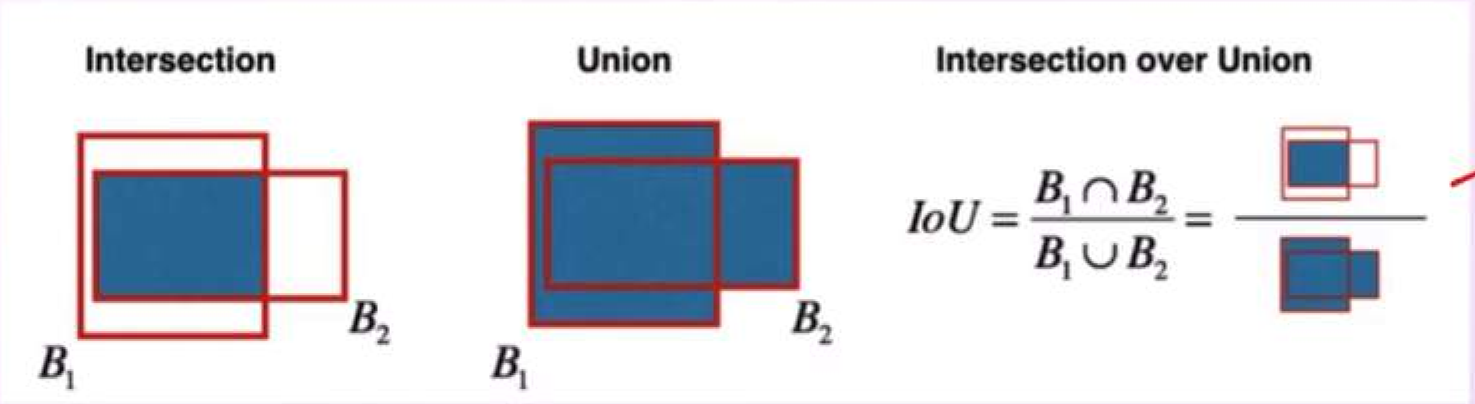
$\rightarrow 1 - \text{IOU} \rightarrow \Downarrow$

Loss ↓

IOU

0.7

IOU Loss



$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}} \rightarrow 1 - \text{IOU} \rightarrow \Downarrow$$

IOU ↑

1 → perfect overlap

0 → No overlap

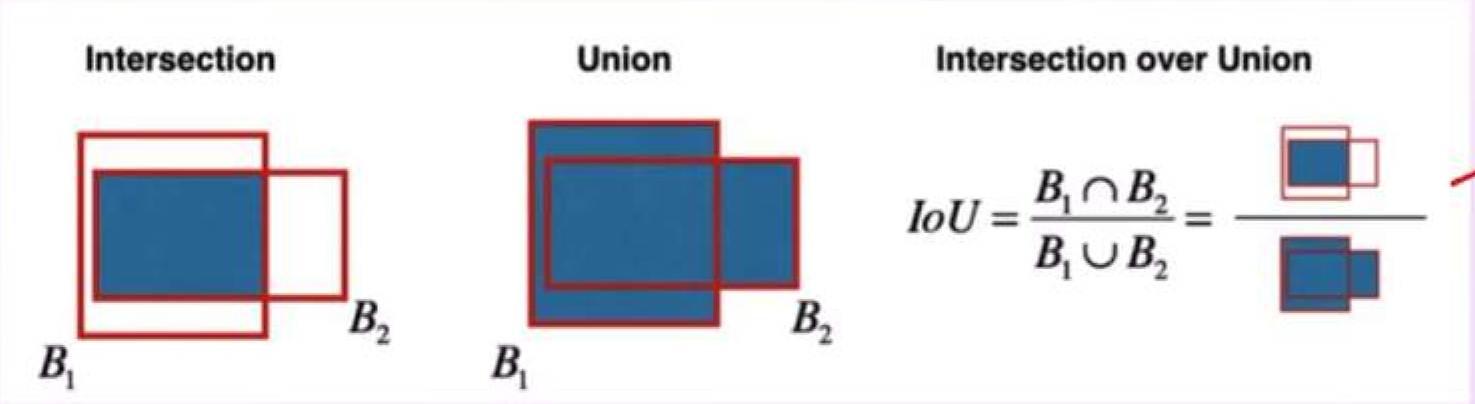
Loss ↓

IOU

0.7

→ 0.3 ↓

IOU Loss



$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}} \rightarrow 1 - IOU \rightarrow$$

IOU ↑

1 → perfect overlap

0 → No overlap

Loss ↓

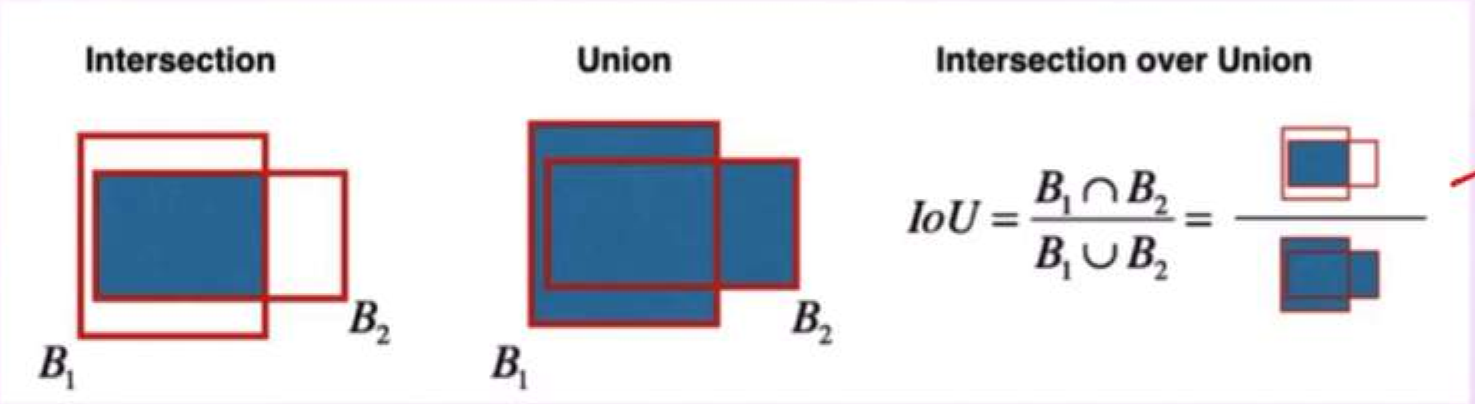
IOU

0.7

0.3 ↓ non overlapping

IOU Loss

IOU ↑



1 → perfect overlap

0 → No overlap

$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

Loss ↓

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

IOU

0.7

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

→ 1 − IOU →

0.3 ↓ Non overlapping region

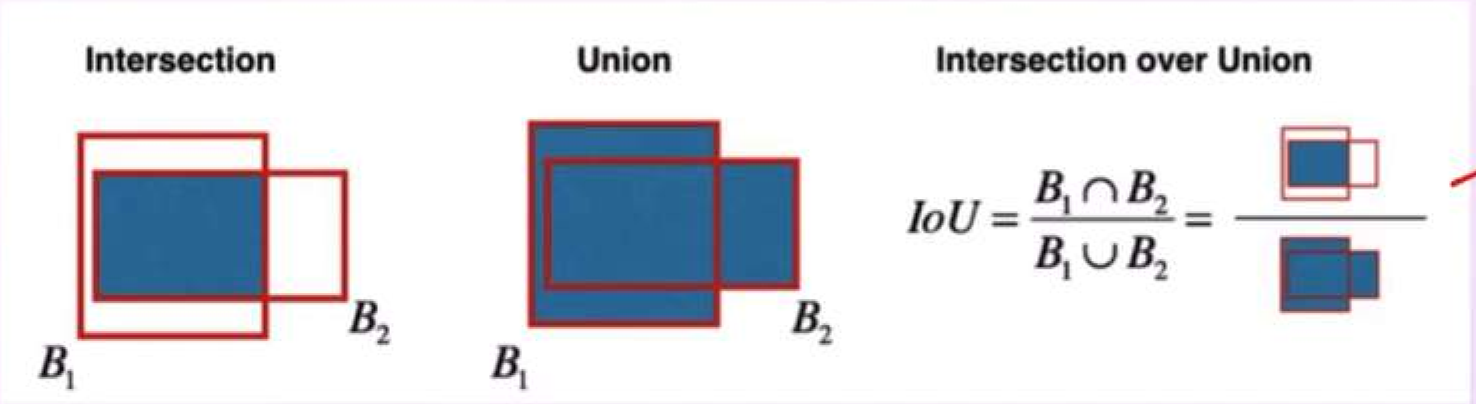
0.

IOU Loss

IOU ↑

1 → perfect overlap

0 → No overlap



$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

Loss ↓

IOU

0.7

1 − IOU →

0.3 ↓

Non overlapping region

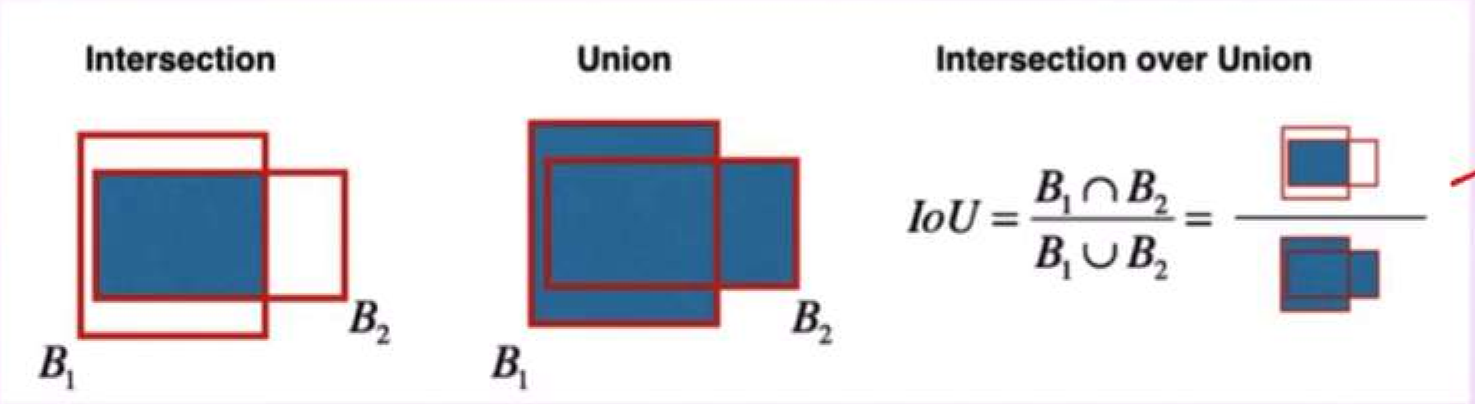
0.

IOU Loss

IOU ↑

1 → perfect overlap

0 → No overlap



$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

→ 1 − IOU →

Loss ↓

IOU

0.7

0.3 ↓

Non overlapping region

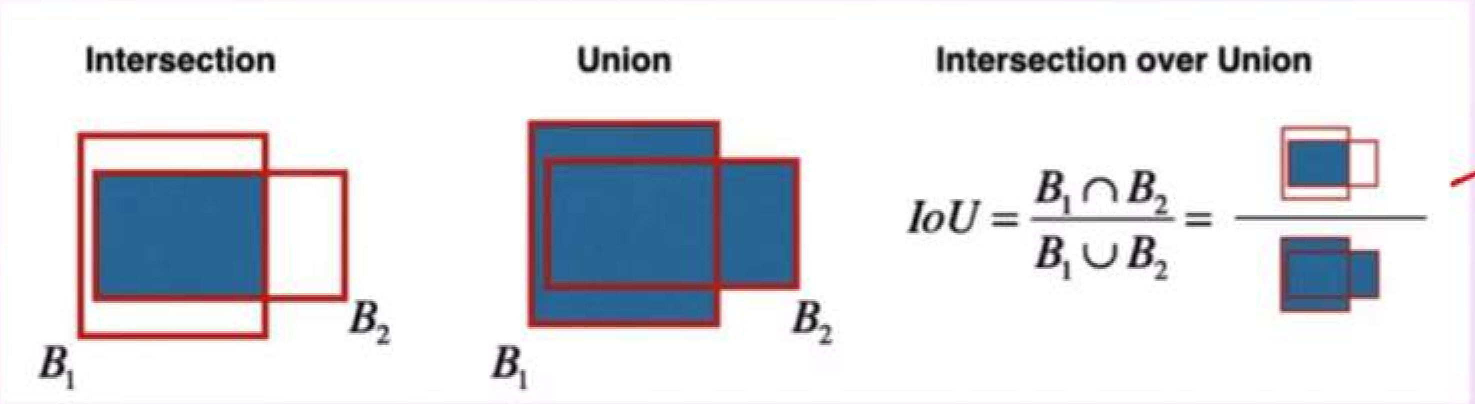
0.

IOU Loss

IOU ↑

1 → perfect overlap

0 → No overlap



$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}} \rightarrow 1 - IOU \rightarrow \Downarrow$$

Loss ↓

IOU

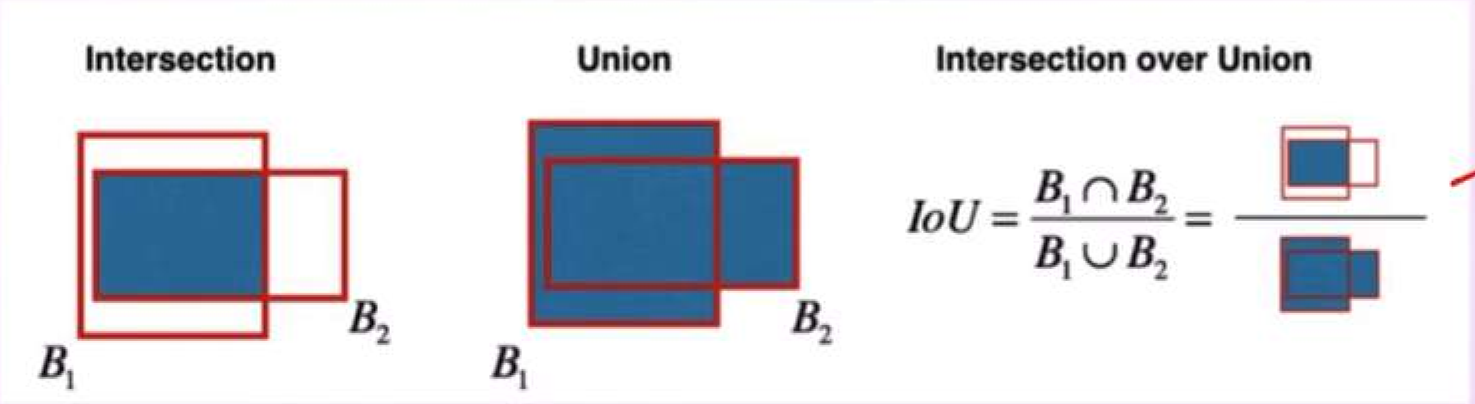
0.7

0.3 ↓

Non overlapping region

0.

IOU Loss



$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}} \rightarrow 1 - IOU \rightarrow$$

IOU ↑

1 → perfect overlap

0 → No overlap

Loss ↓

IOU 0.7 ↓

→ 0.3 ↓ Non overlapping region 0.

IOU Loss

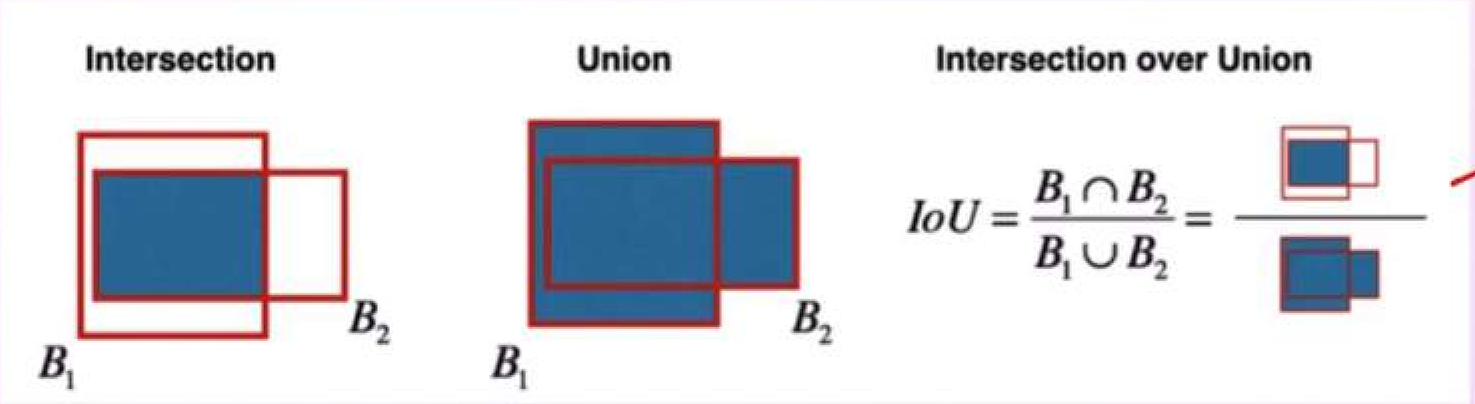
GT

1

IOU ↑

1 → perfect overlap

0 → No overlap



$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}} \rightarrow 1 - \text{IOU} \rightarrow$$

Logs ↓

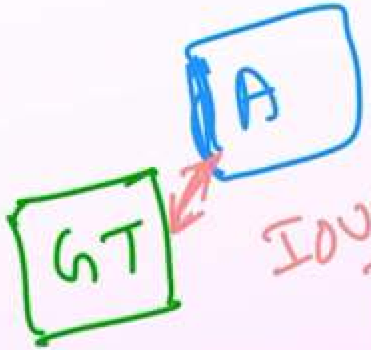
IOU

0.7

→ 0.3 ↓ Non overlapping region

0.

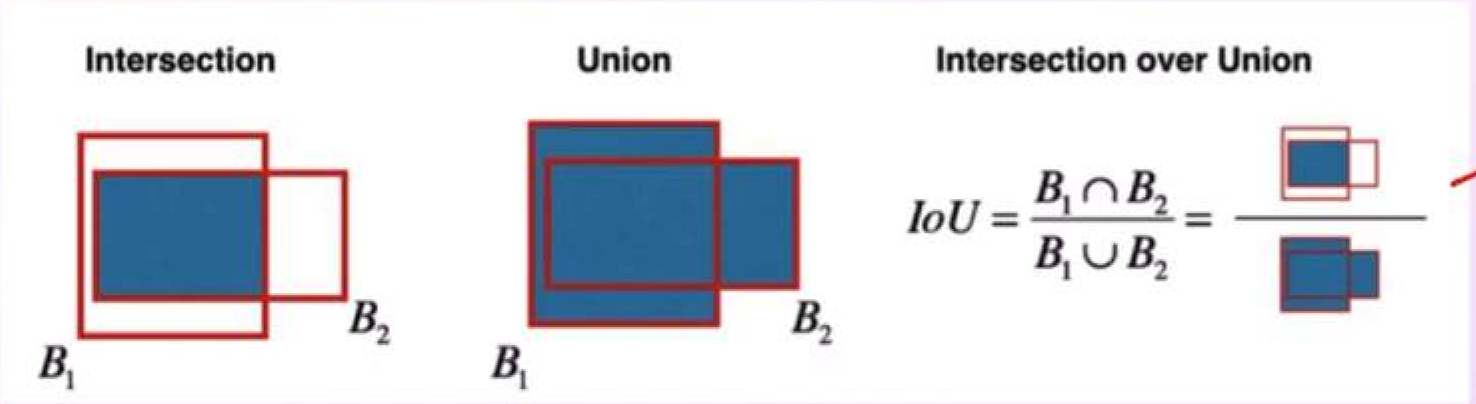
IOU Loss



IOU=0

IOU ↑

1 → perfect overlap



0 → No overlap

Loss ↓

$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt}) \text{ is the ground-truth}$$

$$B = (x, y, w, h) \text{ is the predicted box}$$

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

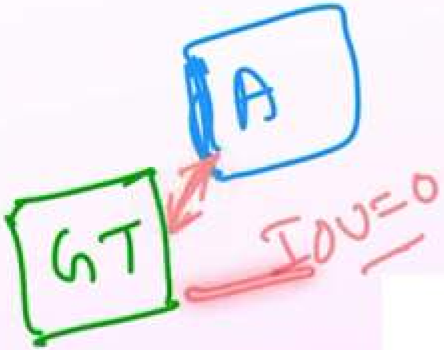
→ 1 − IOU →

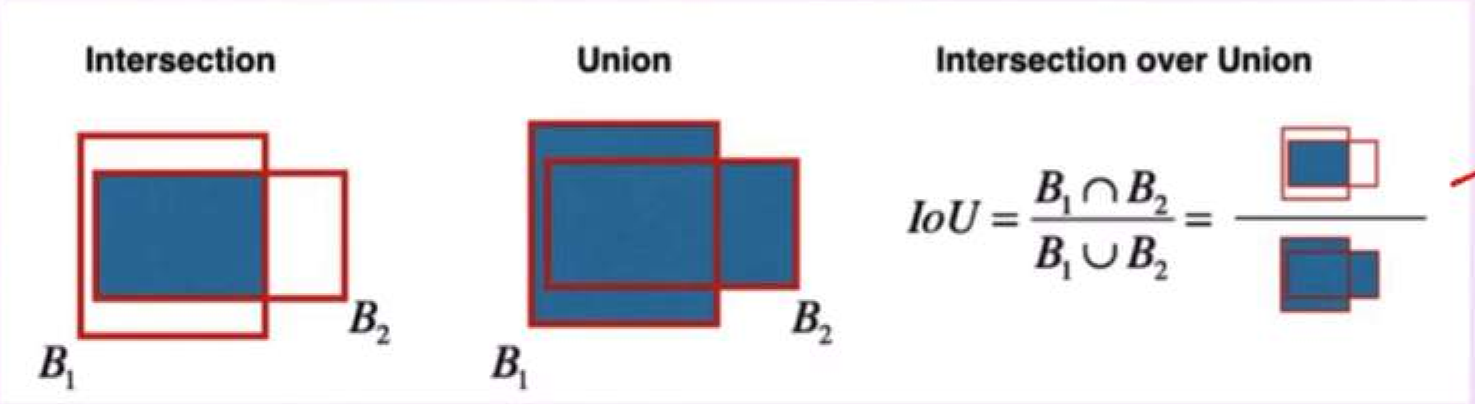
IOU 0.7

0.3 ↓ Non overlapping region

0.

IOU Loss





IOU ↑

1 → perfect overlap

0 → No overlap

$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

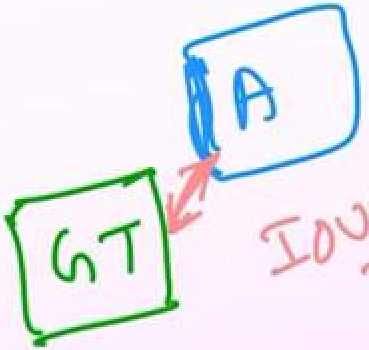
→ 1 − IOU →

Loss ↓

IOU 0.7

→ 0.3 ↓ Non overlapping region 0.

IOU Loss

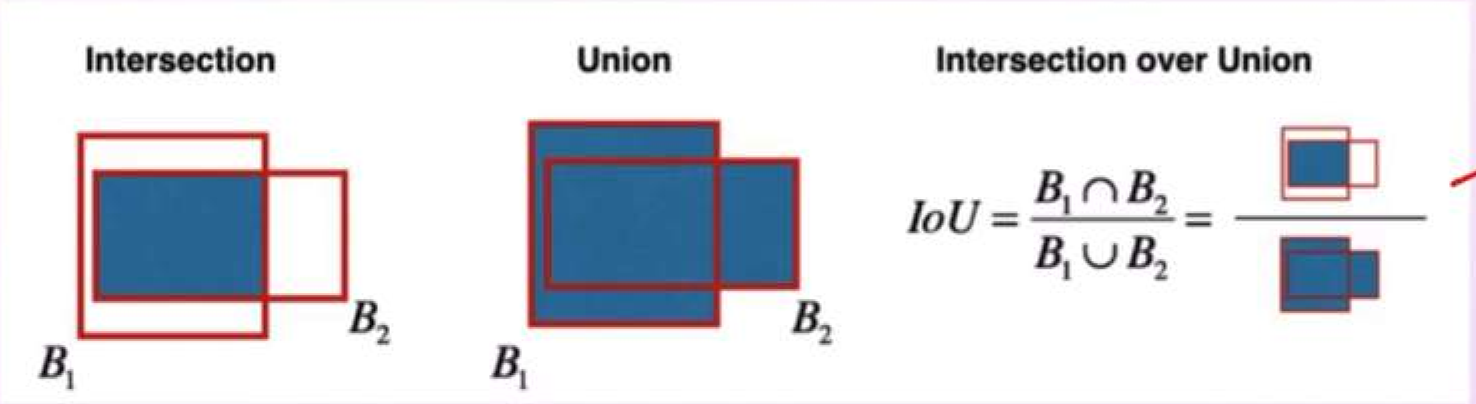


IOU=0

IOU ↑

1 → perfect overlap

0 → No overlap



$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}} \rightarrow 1 - \mathrm{IOU} \rightarrow$$

Loss ↓

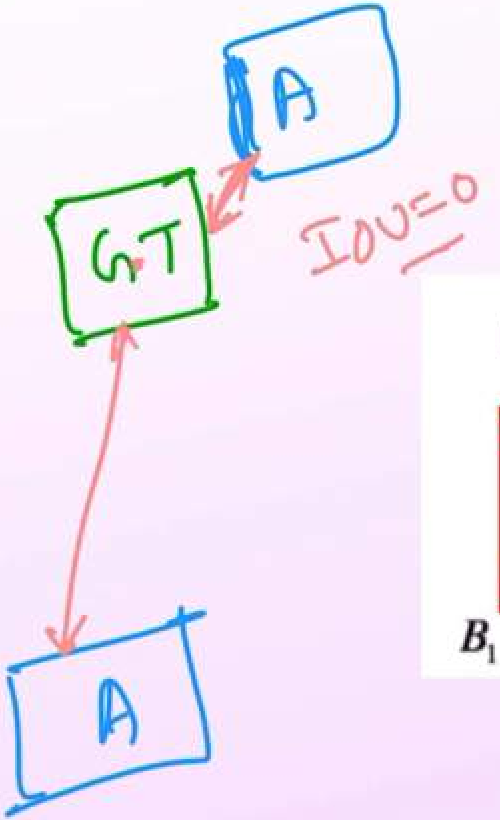
IOU 0.7 ⇓

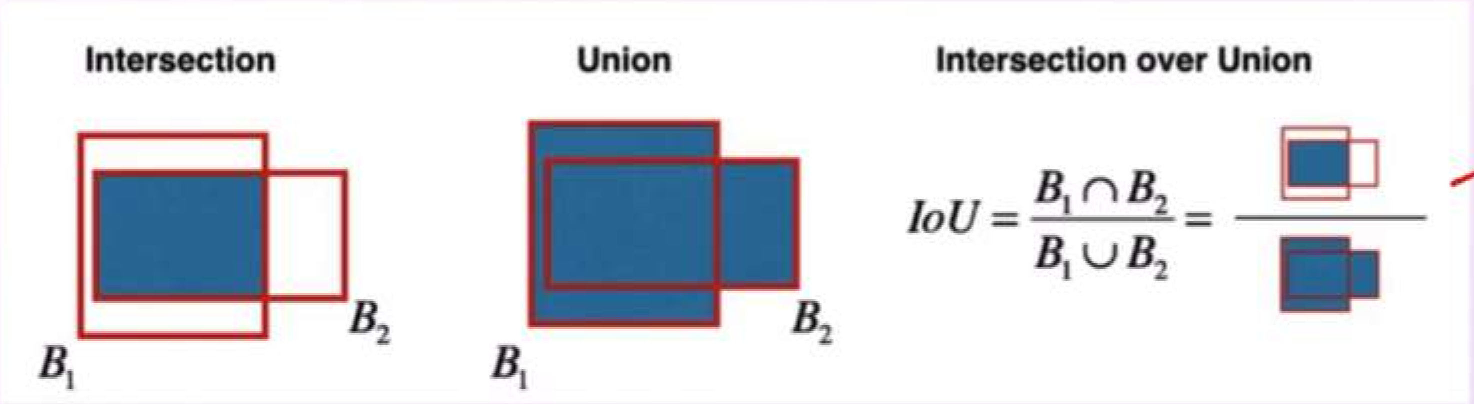
0.3 ↓

Non overlapping region

0.

IOU Loss





IOU ↑

1 → perfect overlap

0 → No overlap

Loss ↓

$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

→ 1 − IOU →

IOU

0.7

0.3 ↓ non overlapping region

0.

IOU Loss

A

GT

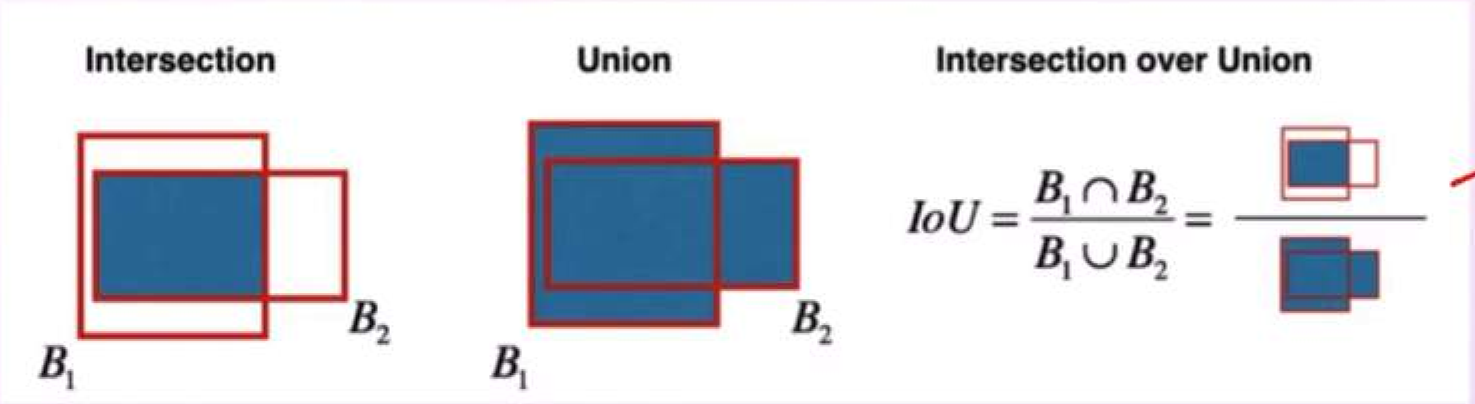
IOU=0

Loss = 1

A

IOU ↑

1 → perfect overlap



0 → No overlap

Loss ↓

$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

→ 1 − IOU →

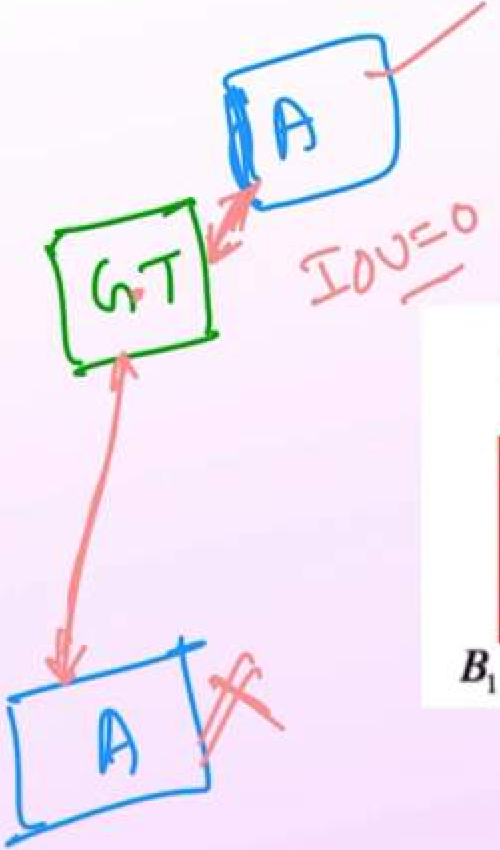
IOU

0.7

0.3 ↓ non overlapping region

0.

IOU Loss

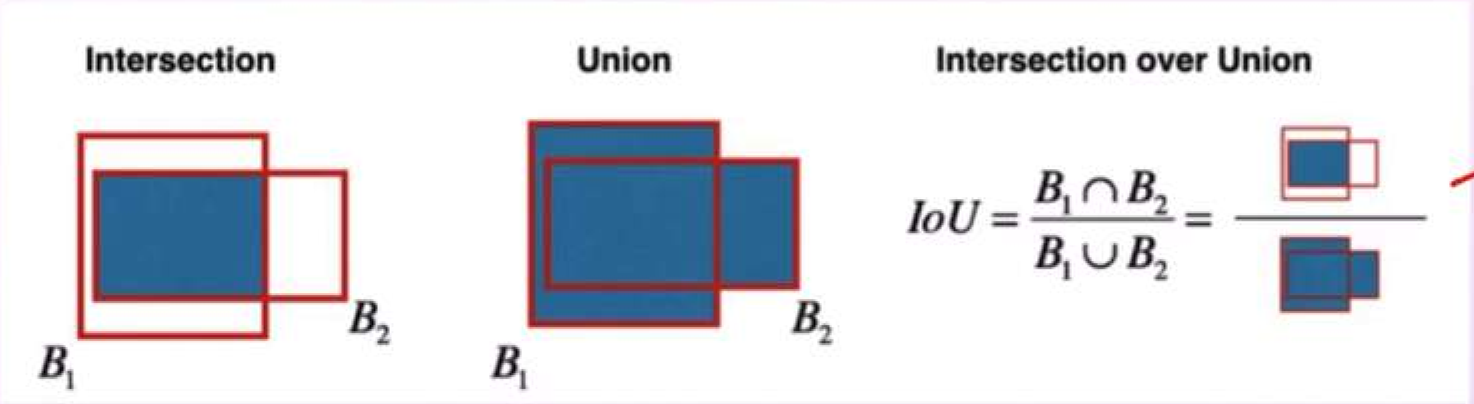


Loss = 0

IOU=0

IOU ↑

1 → perfect overlap



0 → No overlap

$$IoU = \frac{B \cap B^{gt}}{B \cup B^{gt}}$$

$B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground-truth

$B = (x, y, w, h)$ is the predicted box

$$\mathcal{L}_{IoU} = 1 - \frac{B \cap B^{gt}}{B \cup B^{gt}} \rightarrow 1 - \text{IOU} \rightarrow$$

Loss ↓

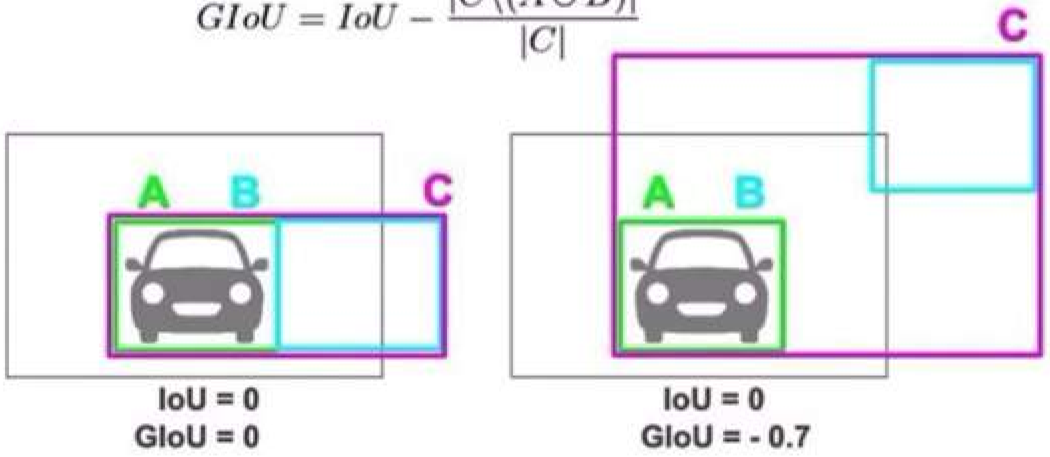
IOU 0.7

→ 0.3 ↓ Non overlapping region

0.

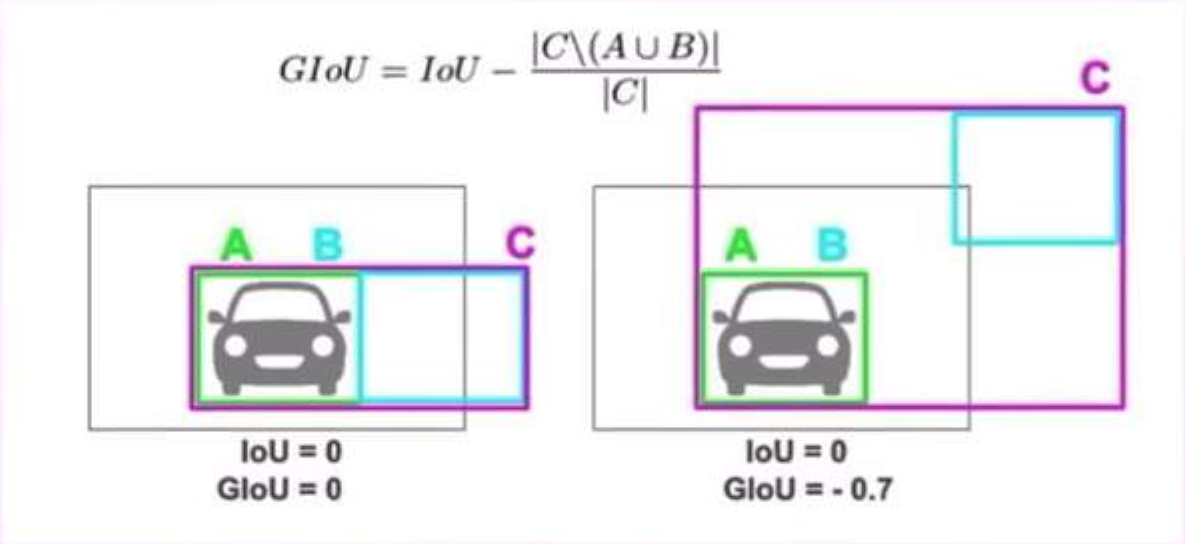
Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$



$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

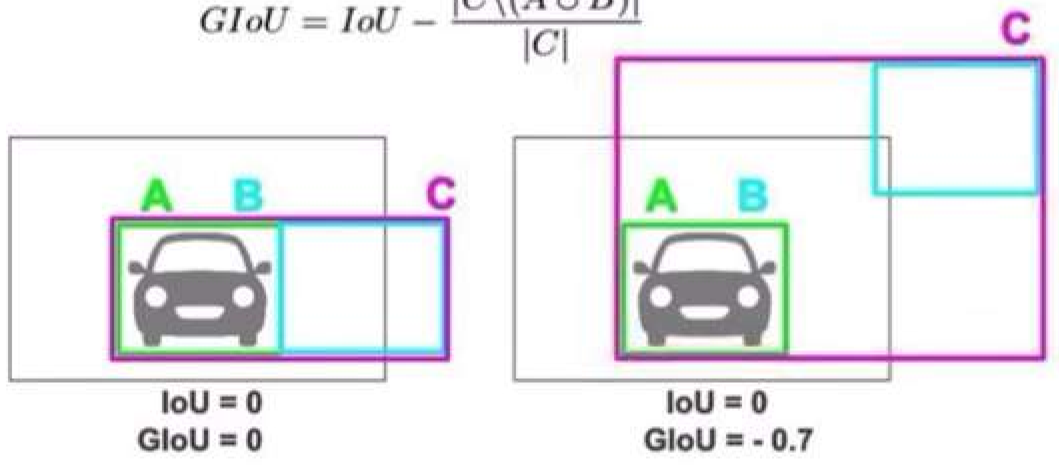
Generalized IOU Loss



$$\mathcal{L}_{GIoU} = \boxed{1 - IoU} + \frac{|C - B \cup B^{gt}|}{|C|}$$

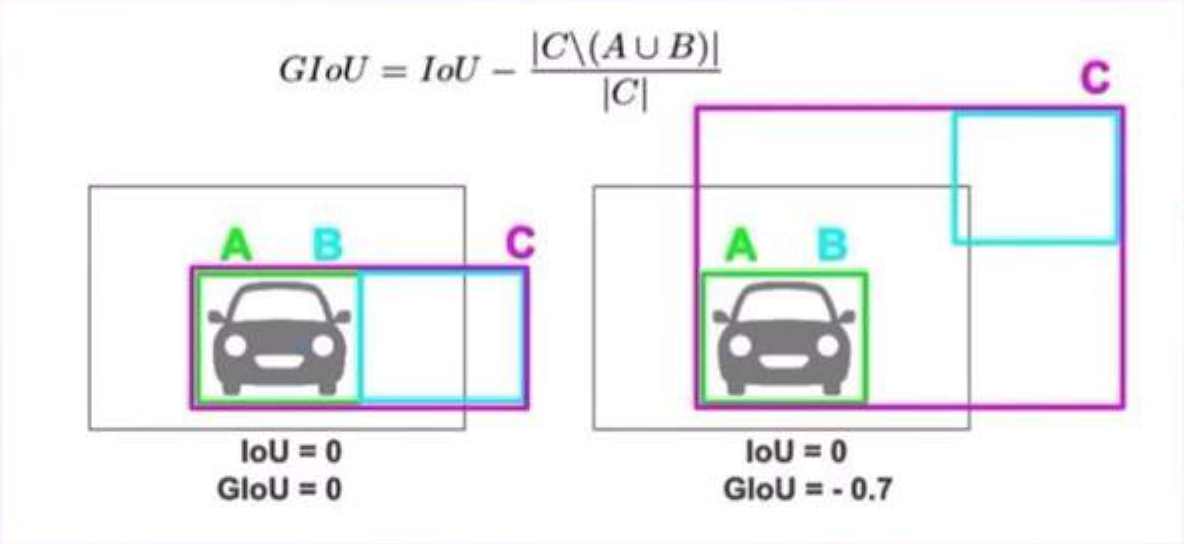
Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$



$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Generalized IOU Loss

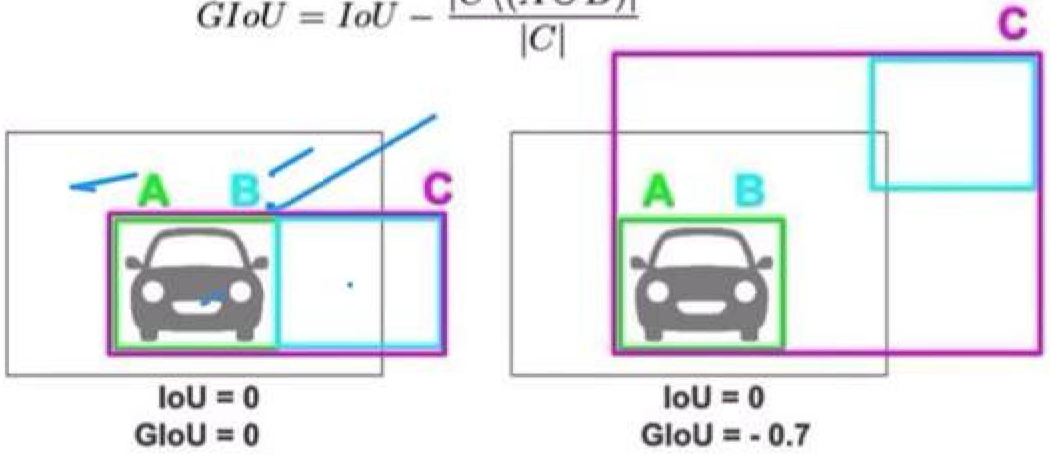


Minimum Enclosed Rectangle

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$

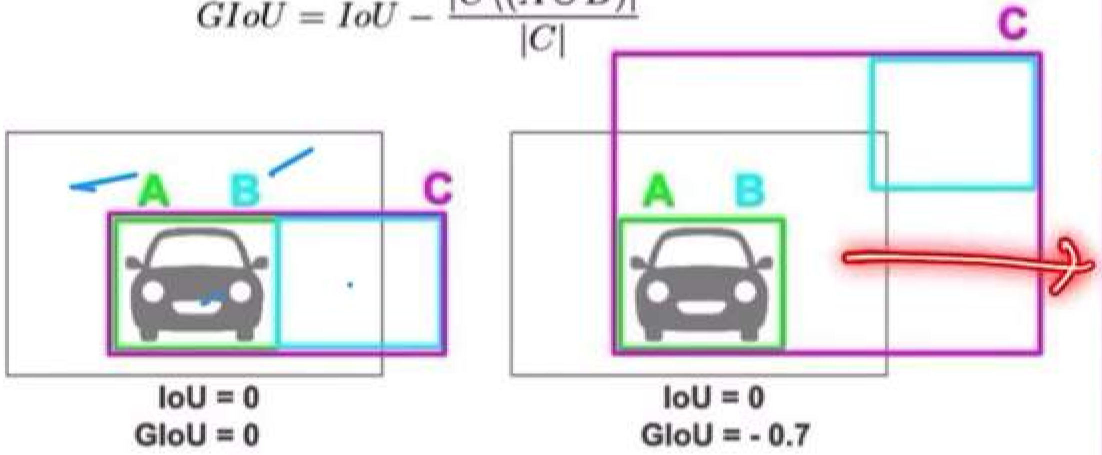


Minimum Enclosed Rectangle

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$

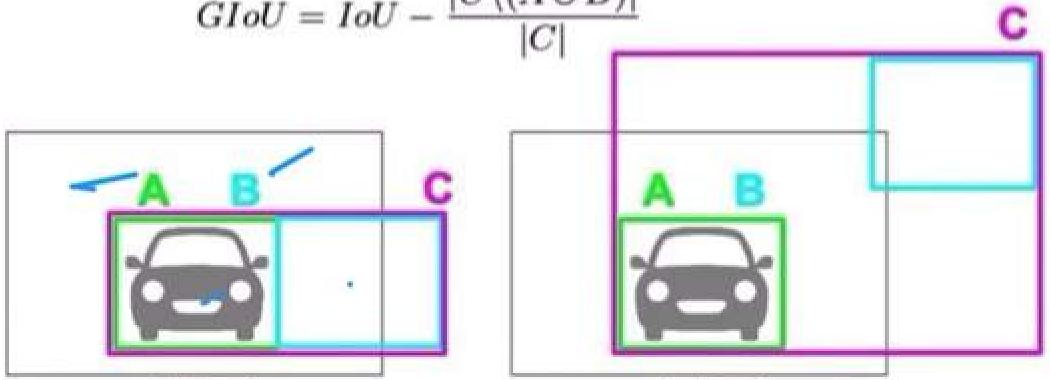


Minimum Enclosed Rectangle

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$



IoU = 0
GIoU = 0

IoU = 0
GIoU = - 0.7

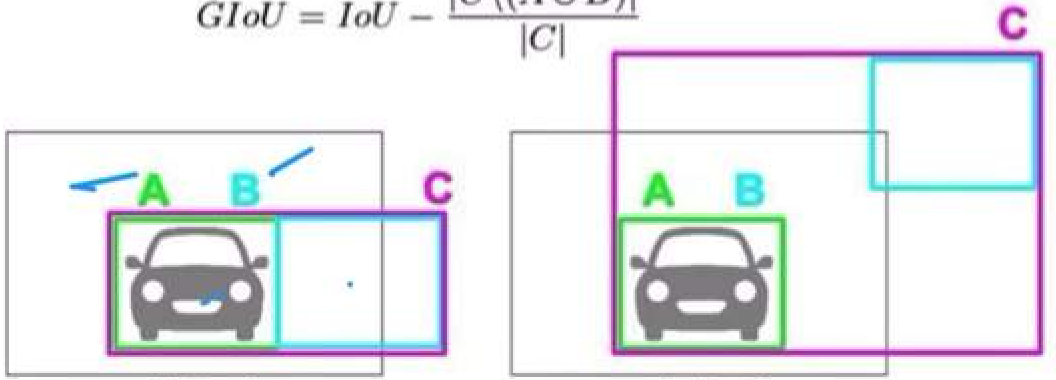
Minimum
Enclosed
Rectangle

Area

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Generalized IOU Loss

$$GIoU = IoU - \frac{|C\backslash(A \cup B)|}{|C|}$$



Minimum Enclosed Rectangle

Area(C) > far

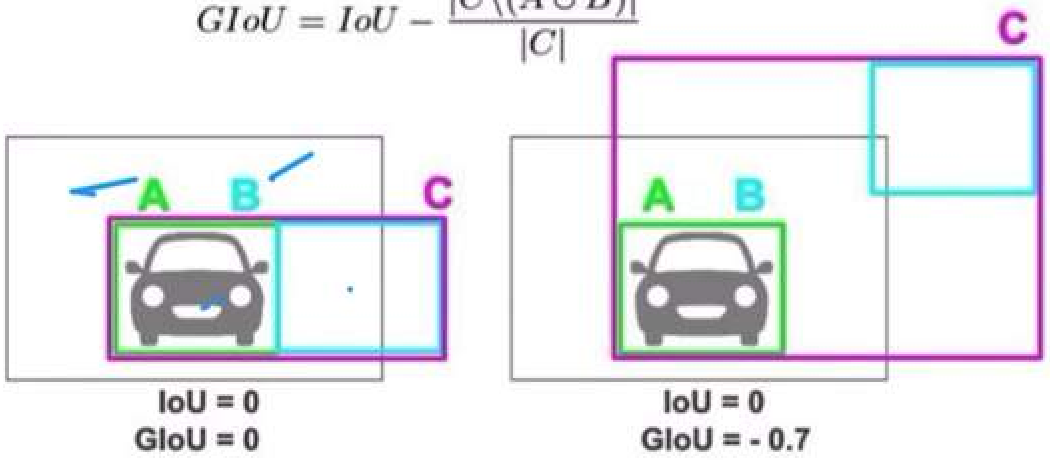
↓

↓

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Generalized IOU Loss

$$GIoU = IoU - \frac{|C\backslash(A \cup B)|}{|C|}$$



Minimum Enclosed Rectangle

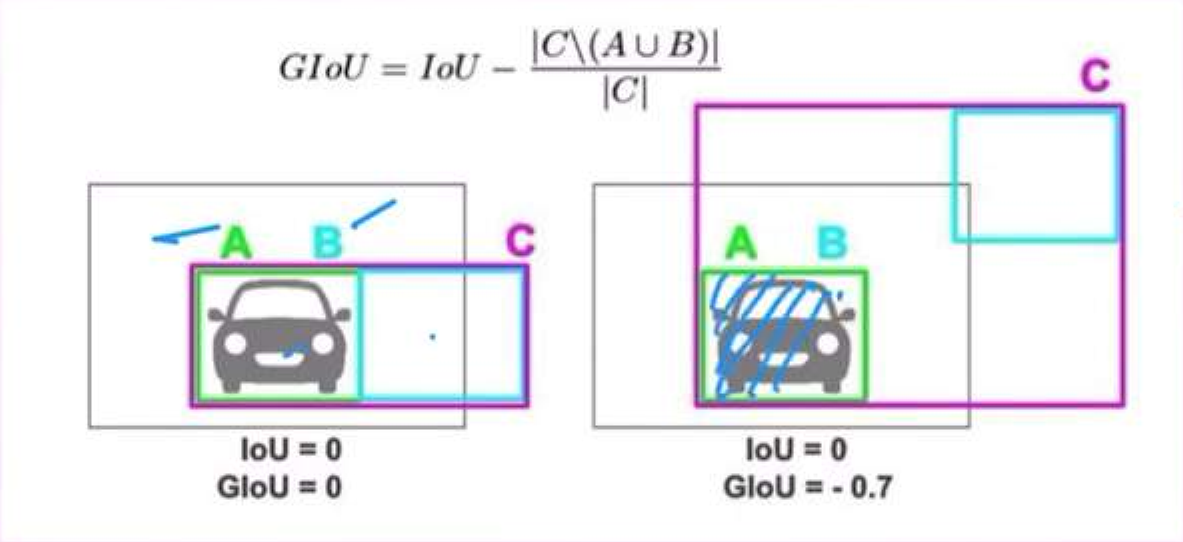
Area(C) > far

↓

Iou

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Generalized IOU Loss



Minimum Enclosed Rectangle

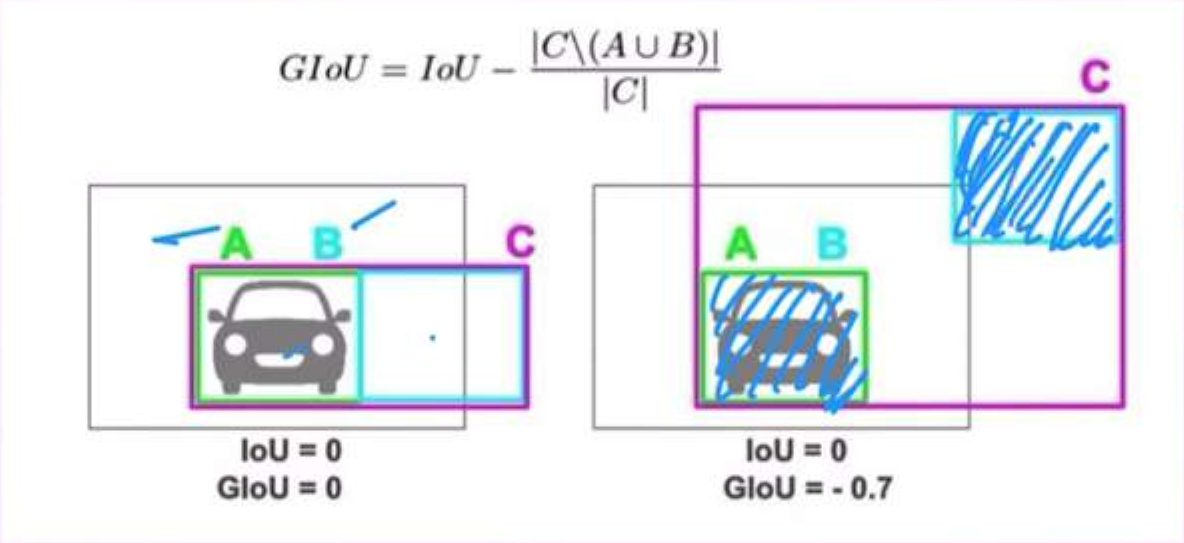
Area(C) > far

↓

loss

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Generalized IOU Loss



Minimum Enclosed Rectangle

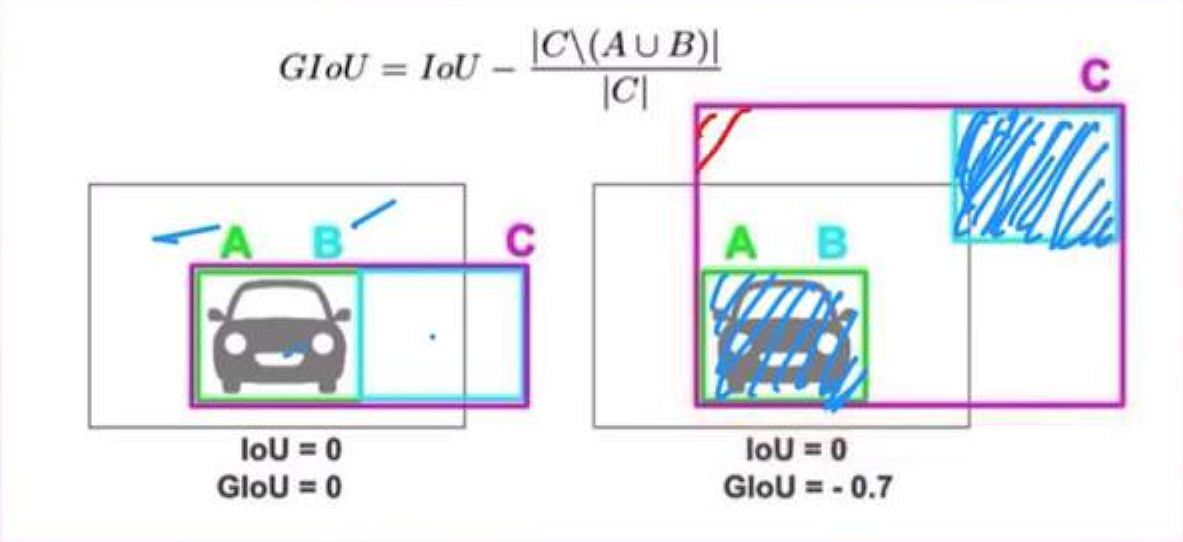
Area(C) > far

↓

Iou

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Generalized IOU Loss



Minimum Enclosed Rectangle

Area(C) > far

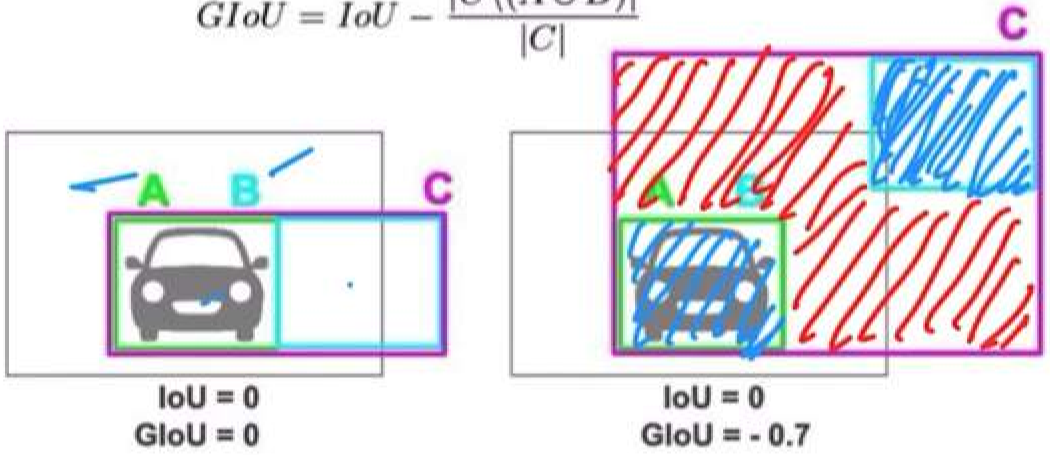
↓

low

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$



Minimum Enclosed Rectangle

Area(C) > far

↓

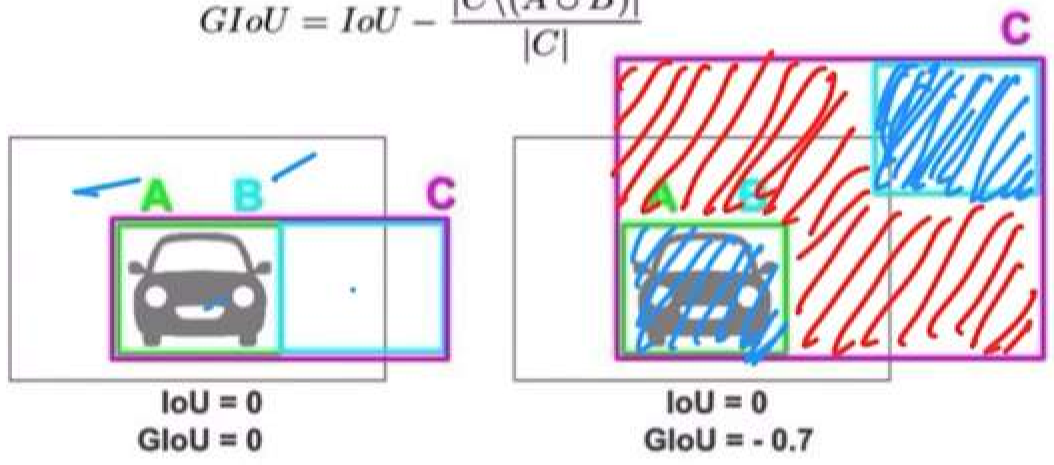
low

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

0, 1

Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$



Minimum Enclosed Rectangle

Area(C) > far

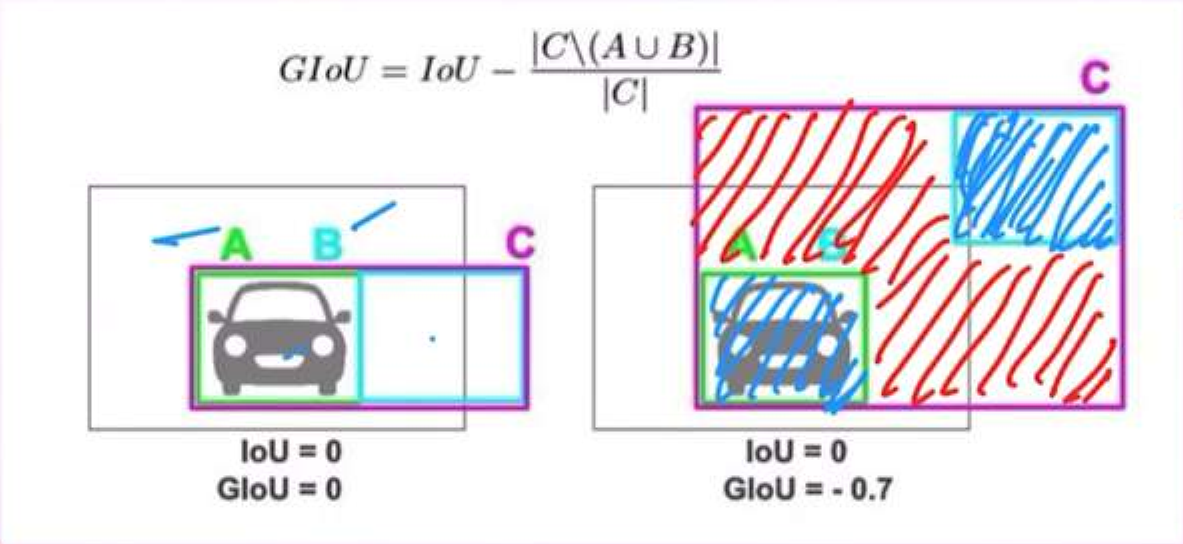
↓

low

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

(0, 1)

Generalized IOU Loss



Minimum Enclosed Rectangle

Area(C) > far

↓

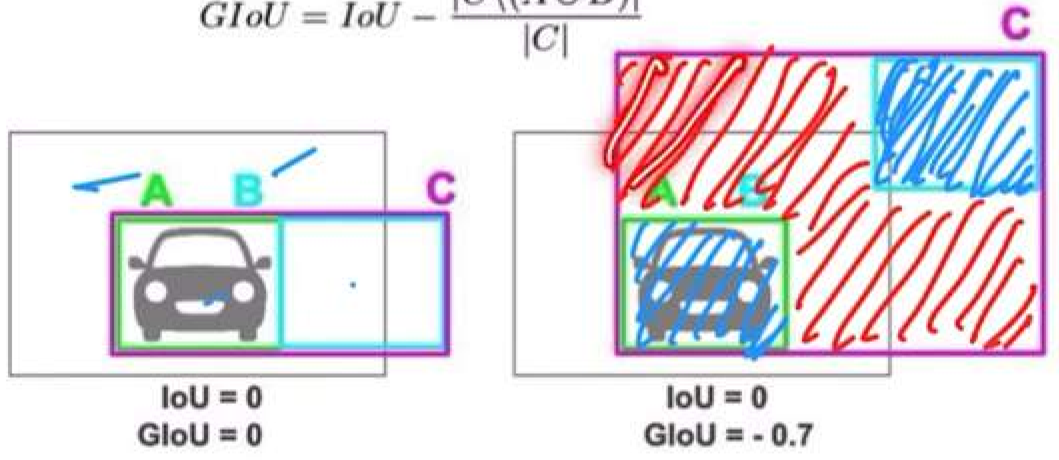
IoU

$C - B \cup B^{gt}$

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|} \rightarrow (0, 1)$$

Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$



Minimum Enclosed Rectangle

Area(C) > far

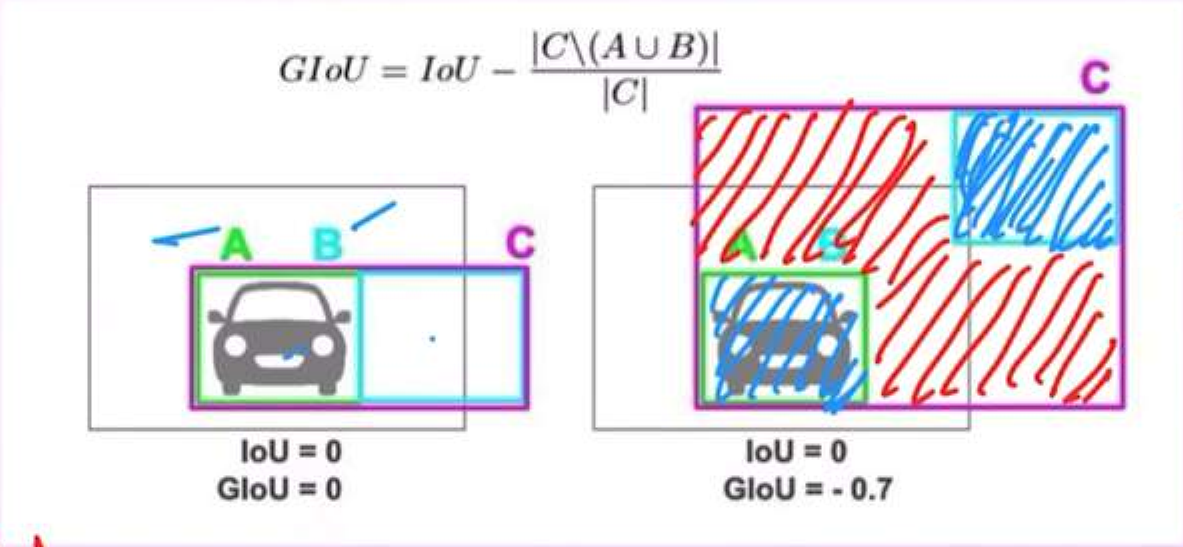
low

$C - B \cup B^{gt}$

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

(0, 1)

Generalized IOU Loss



Minimum Enclosed Rectangle

Area(C) > far

↓

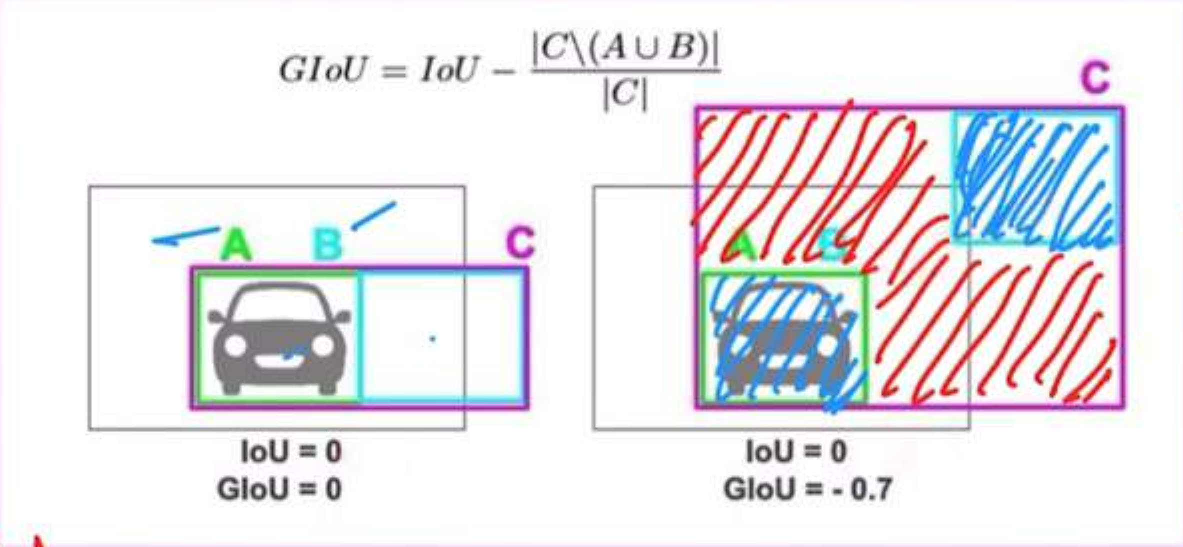
low

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

(0, 1)

C − B ∪ B^gt

Generalized IOU Loss



Minimum Enclosed Rectangle

Area(C) > far

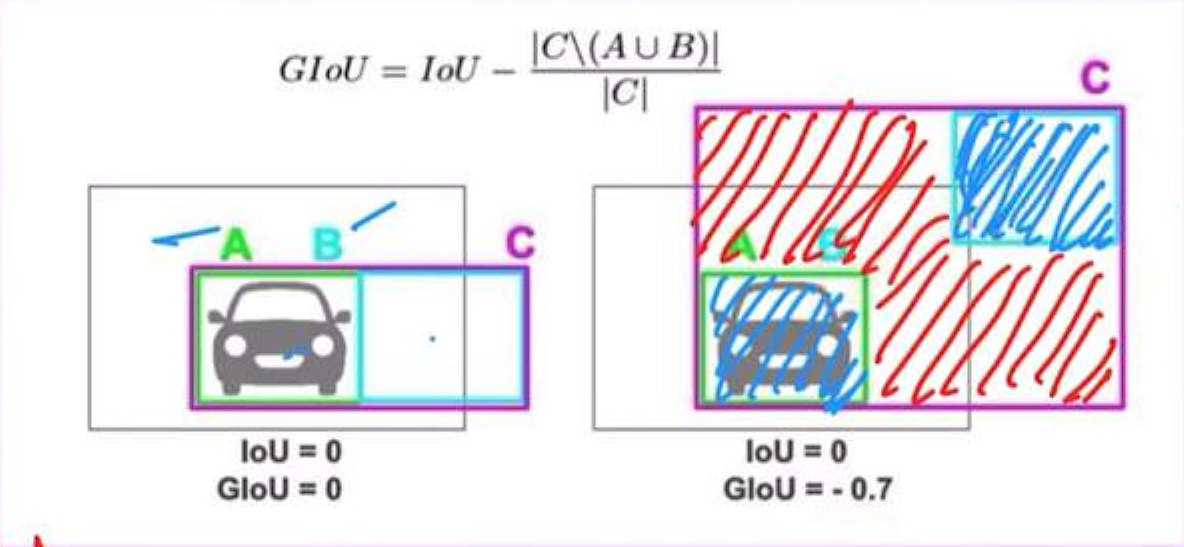
low

$C - B \cup B^{gt}$

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

(0, 1)

Generalized IOU Loss



Minimum Enclosed Rectangle

Area(C) > far

↓

Iou

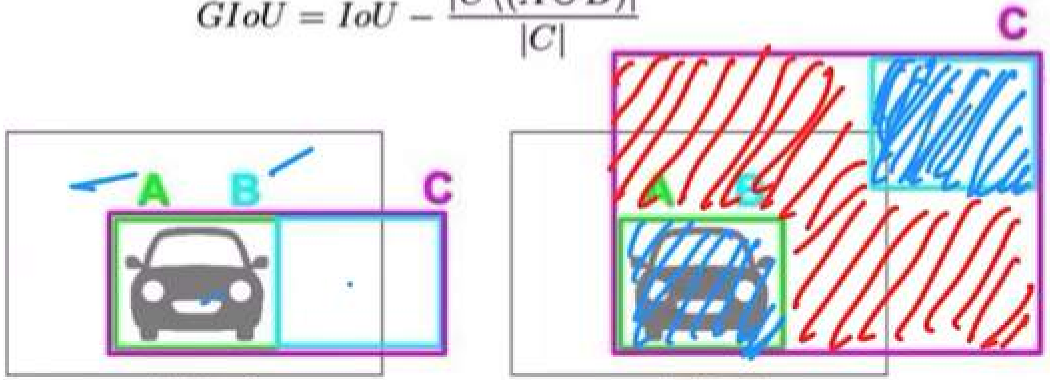
C − B ∪ B^gt

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

(0, 1)

Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$



IoU = 0
GIoU = 0

IoU = 0
GIoU = - 0.7

Minimum Enclosed Rectangle

Area(C) > far

↓

loss

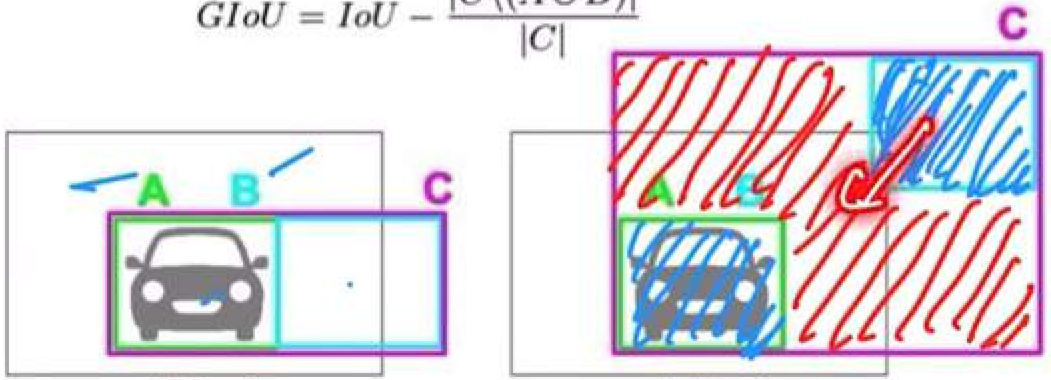
$C - B \cup B^{gt}$

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

(0, 1)

Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$



Minimum Enclosed Rectangle

Area(C) > far

↓

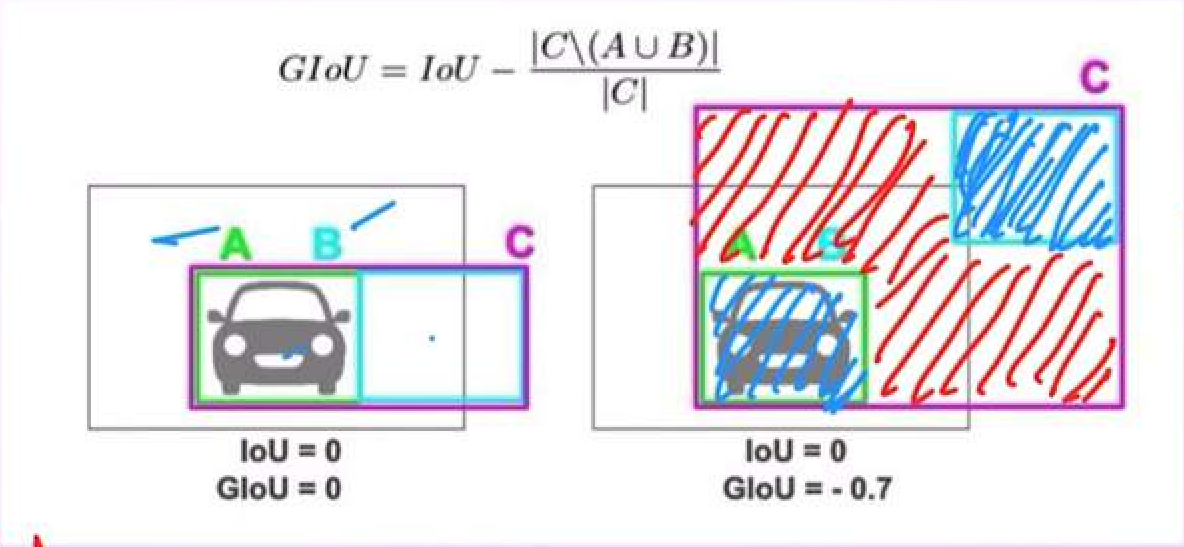
low

$C - B \cup B^{gt}$

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

(0, 1)

Generalized IOU Loss



Minimum Enclosed Rectangle

Area(C) > far

↓

IoU

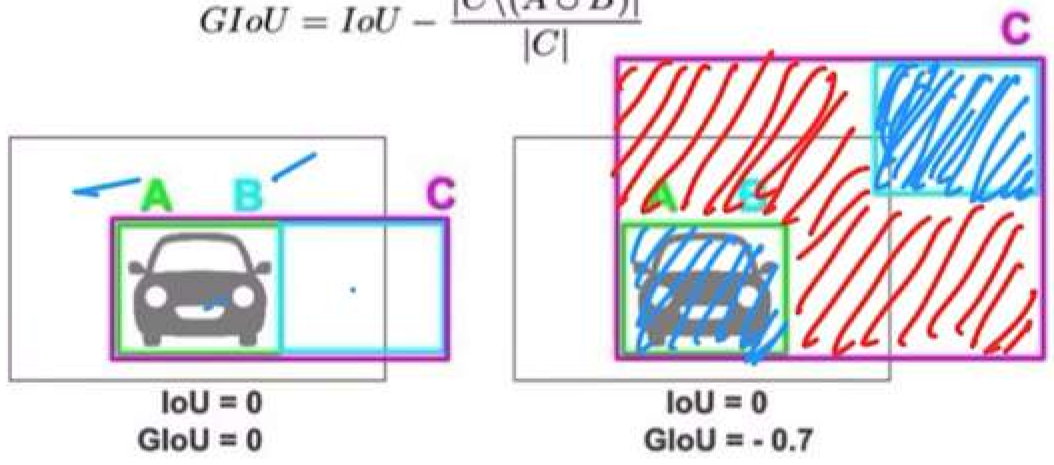
$C - B \cup B^{gt}$

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

(0, 1)

Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$



Minimum Enclosed Rectangle

Area(C) > far

↓

IoU

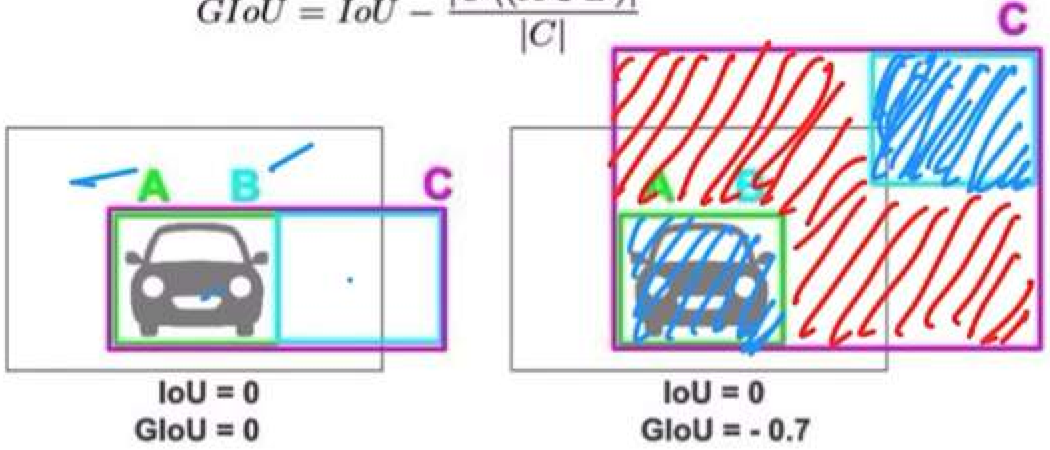
$C - B \cup B^{gt}$

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

(0, 1)

Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$



Minimum Enclosed Rectangle

Area(C) > far

↓

low

$C - B \cup B^{gt}$

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

$(0,1)$

Distance IOU Loss

$$\mathcal{L}_{DIoU} = 1 - IoU + \underline{\frac{\rho^2(b, b^{gt})}{c^2}}$$

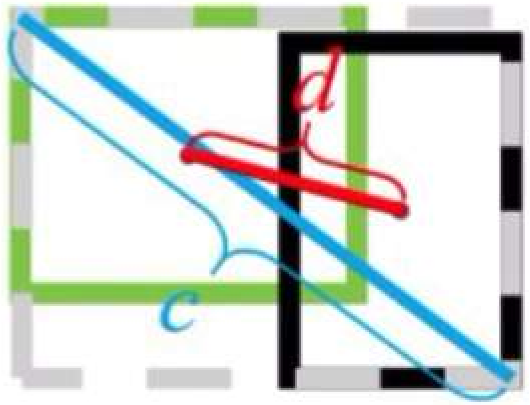


Figure 5: DIoU loss for bounding box regression, where the normalized distance between central points can be directly minimized. c is the diagonal length of the smallest enclosing box covering two boxes, and $d = \rho(\mathbf{b}, \mathbf{b}^{gt})$ is the distance of central points of two boxes.

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Distance IOU Loss

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

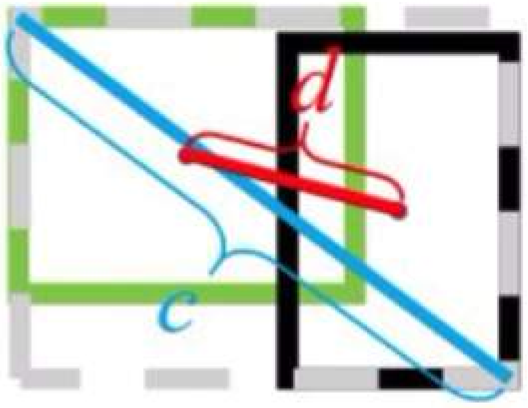


Figure 5: DIoU loss for bounding box regression, where the normalized distance between central points can be directly minimized. c is the diagonal length of the smallest enclosing box covering two boxes, and $d = \rho(\mathbf{b}, \mathbf{b}^{gt})$ is the distance of central points of two boxes.

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Distance IOU Loss

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

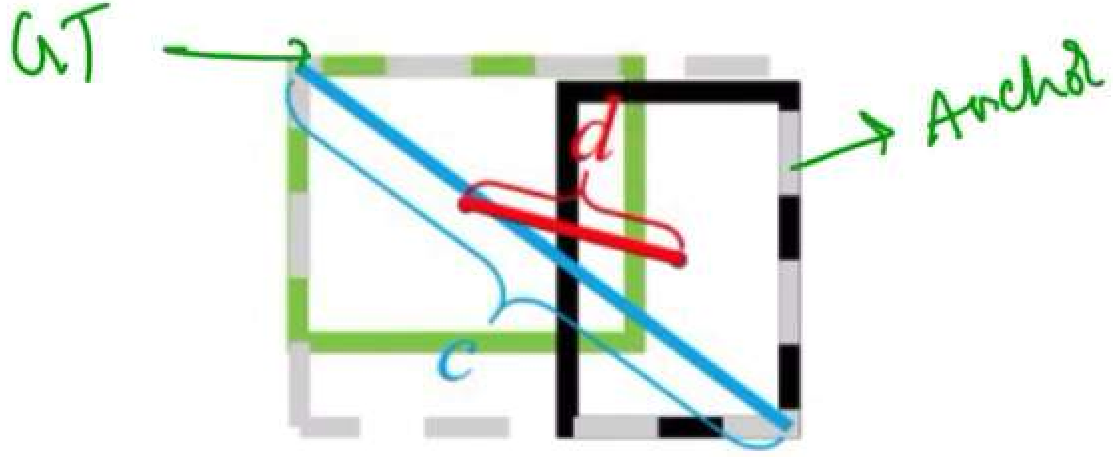


Figure 5: DIoU loss for bounding box regression, where the normalized distance between central points can be directly minimized. c is the diagonal length of the smallest enclosing box covering two boxes, and $d = \rho(\mathbf{b}, \mathbf{b}^{gt})$ is the distance of central points of two boxes.

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Distance IOU Loss

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

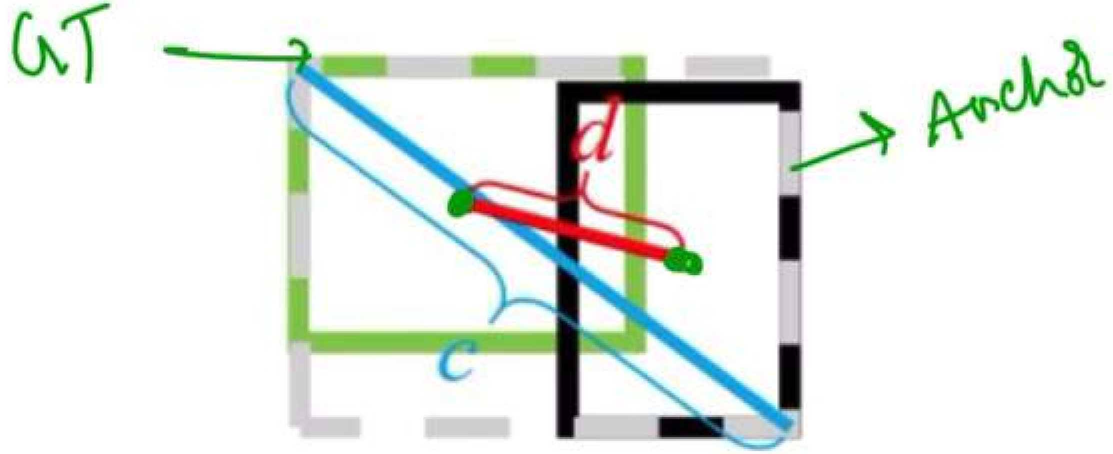


Figure 5: DIoU loss for bounding box regression, where the normalized distance between central points can be directly minimized. c is the diagonal length of the smallest enclosing box covering two boxes, and $d = \rho(\mathbf{b}, \mathbf{b}^{gt})$ is the distance of central points of two boxes.

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Distance IOU Loss

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$ → d

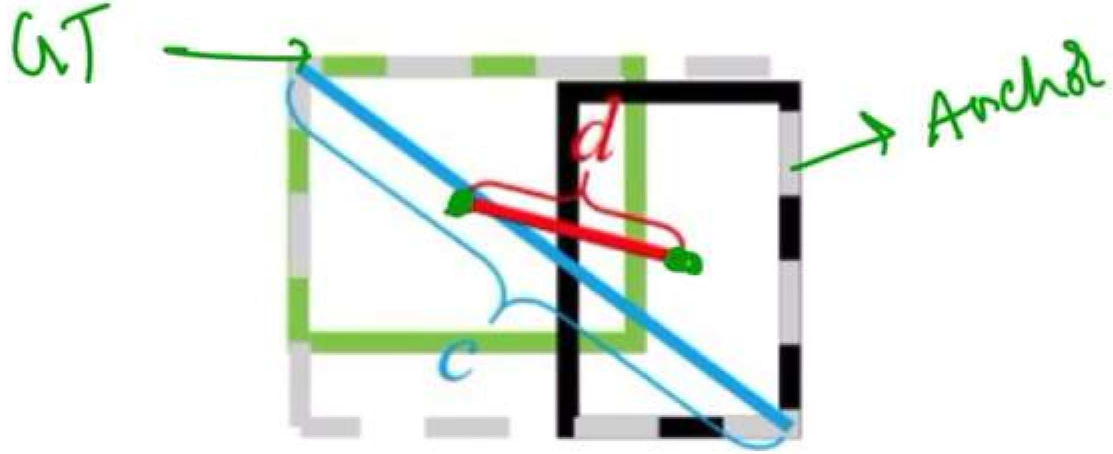
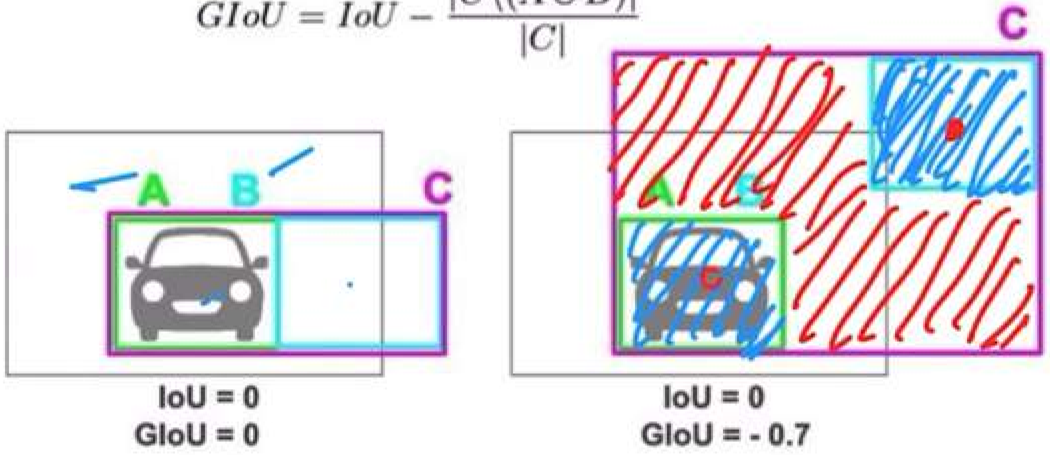


Figure 5: DIoU loss for bounding box regression, where the normalized distance between central points can be directly minimized. c is the diagonal length of the smallest enclosing box covering two boxes, and $d = \rho(\mathbf{b}, \mathbf{b}^{gt})$ is the distance of central points of two boxes.

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$



Minimum Enclosed Rectangle

Area(C) > far

↓

loss

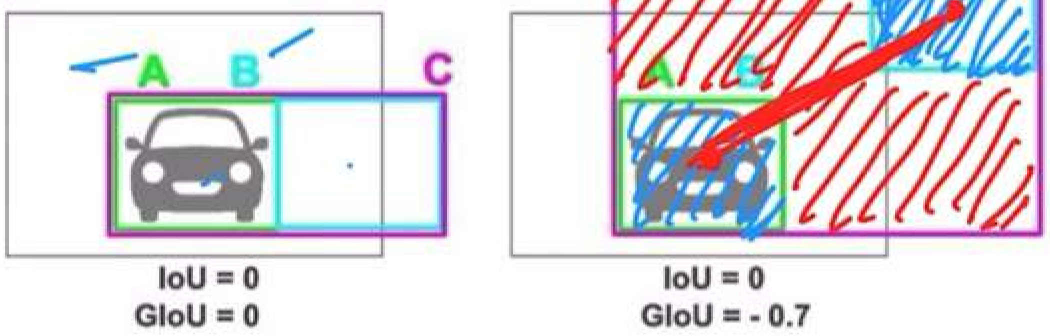
$C - B \cup B^{gt}$

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

(0, 1)

Generalized IOU Loss

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$$



Minimum Enclosed Rectangle

Area(C) > far

↓

low

$C - B \cup B^{gt}$

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

$(0, 1)$

Distance IOU Loss

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} \rightarrow d$$

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

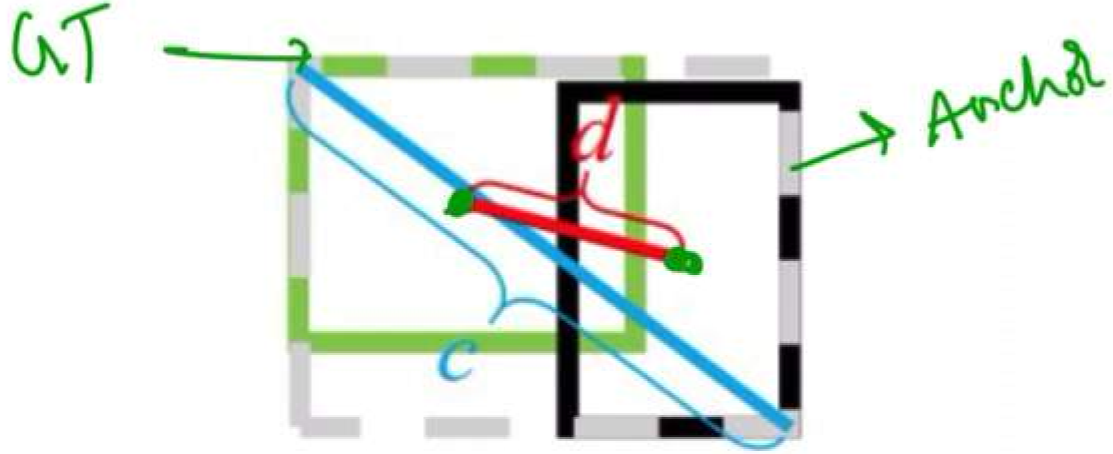


Figure 5: DIoU loss for bounding box regression, where the normalized distance between central points can be directly minimized. c is the diagonal length of the smallest enclosing box covering two boxes, and $d = \rho(\mathbf{b}, \mathbf{b}^{gt})$ is the distance of central points of two boxes.

Distance IOU Loss

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} \longrightarrow d$$

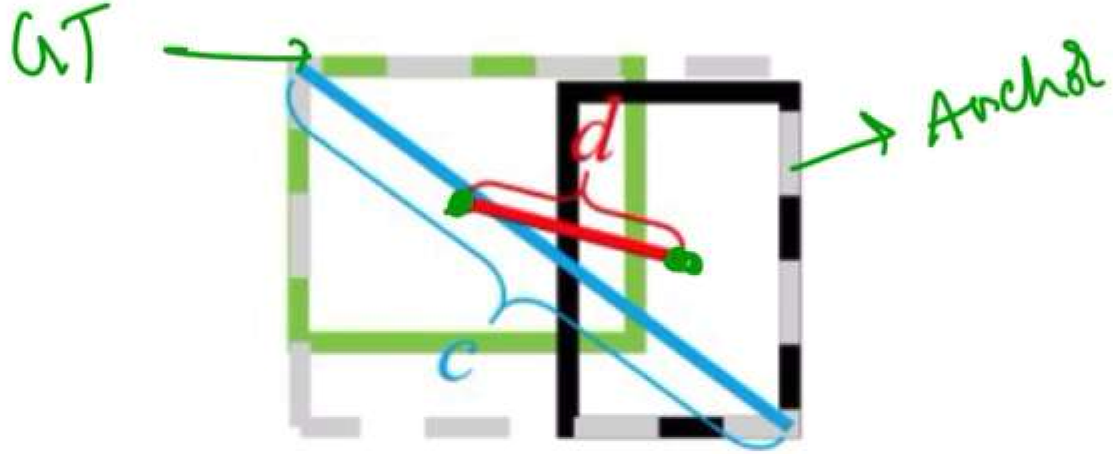


Figure 5: DIoU loss for bounding box regression, where the normalized distance between central points can be directly minimized. c is the diagonal length of the smallest enclosing box covering two boxes, and $d = \rho(\mathbf{b}, \mathbf{b}^{gt})$ is the distance of central points of two boxes.

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Distance IOU Loss

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\boxed{\rho^2(b, b^{gt})} \rightarrow d}{c^2}$$

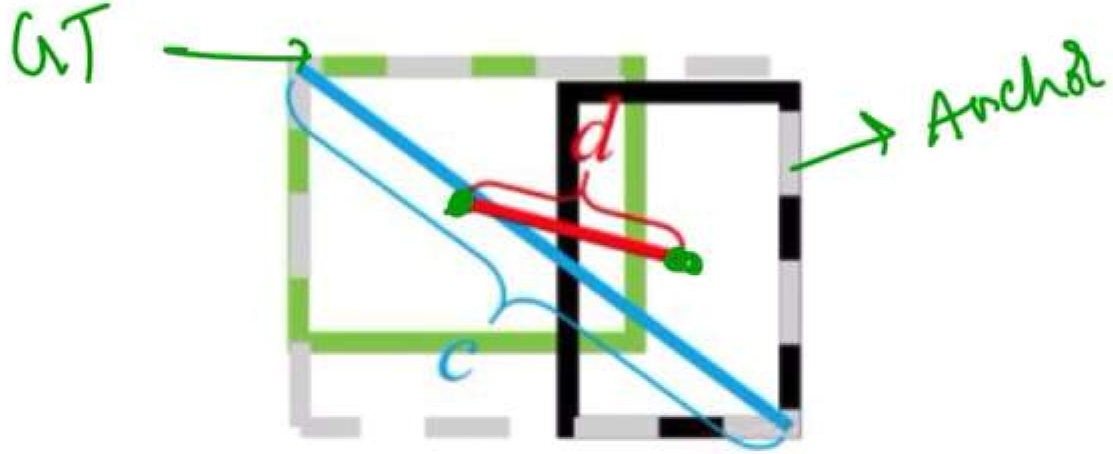


Figure 5: DIoU loss for bounding box regression, where the normalized distance between central points can be directly minimized. c is the diagonal length of the smallest enclosing box covering two boxes, and $d = \rho(\mathbf{b}, \mathbf{b}^{gt})$ is the distance of central points of two boxes.

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Distance IOU Loss

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} \rightarrow d$$

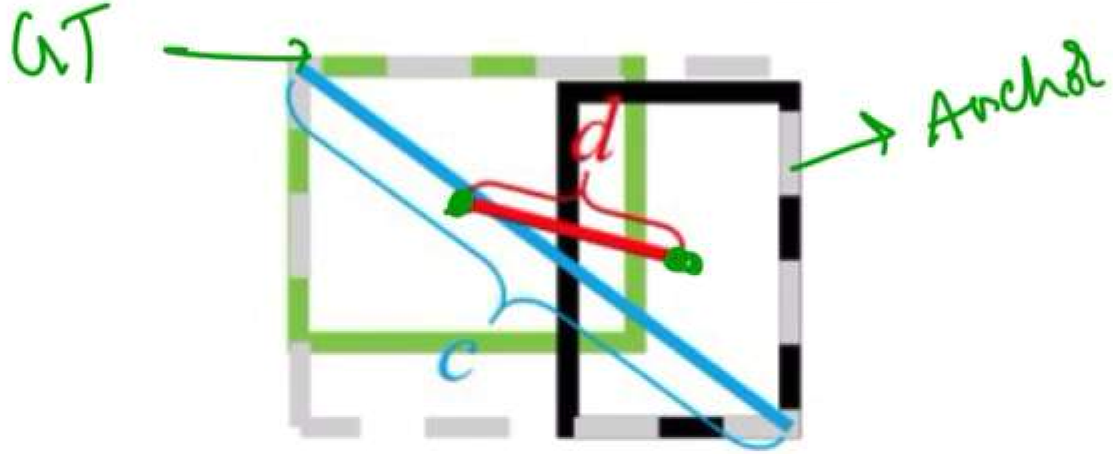


Figure 5: DIoU loss for bounding box regression, where the normalized distance between central points can be directly minimized. c is the diagonal length of the smallest enclosing box covering two boxes, and $d = \rho(\mathbf{b}, \mathbf{b}^{gt})$ is the distance of central points of two boxes.

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

Distance IOU Loss

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

d

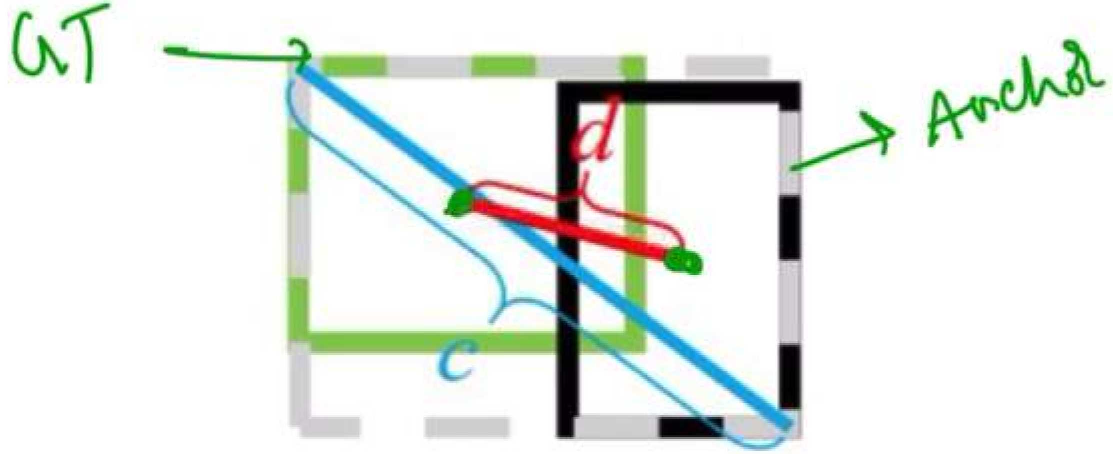
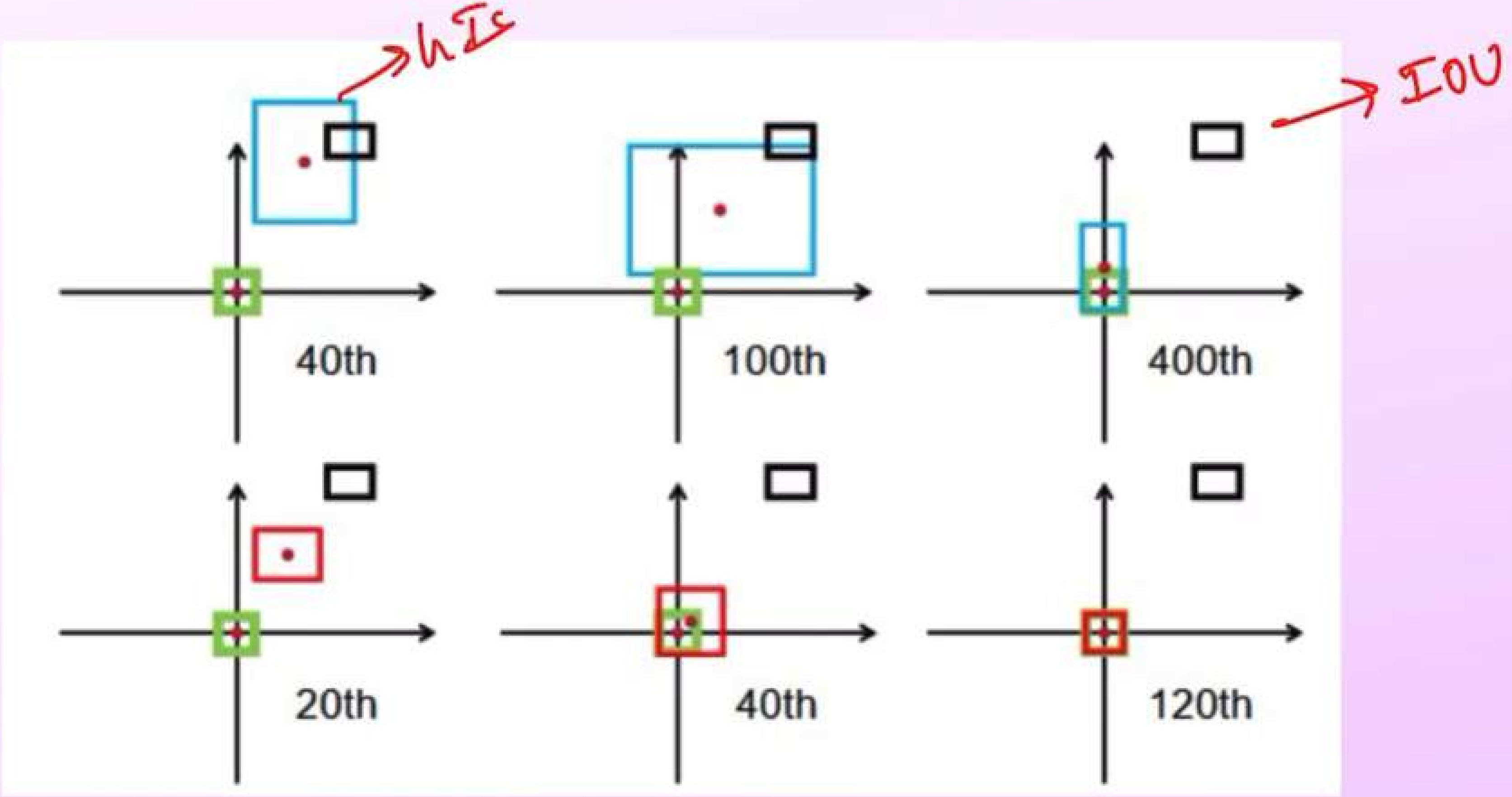


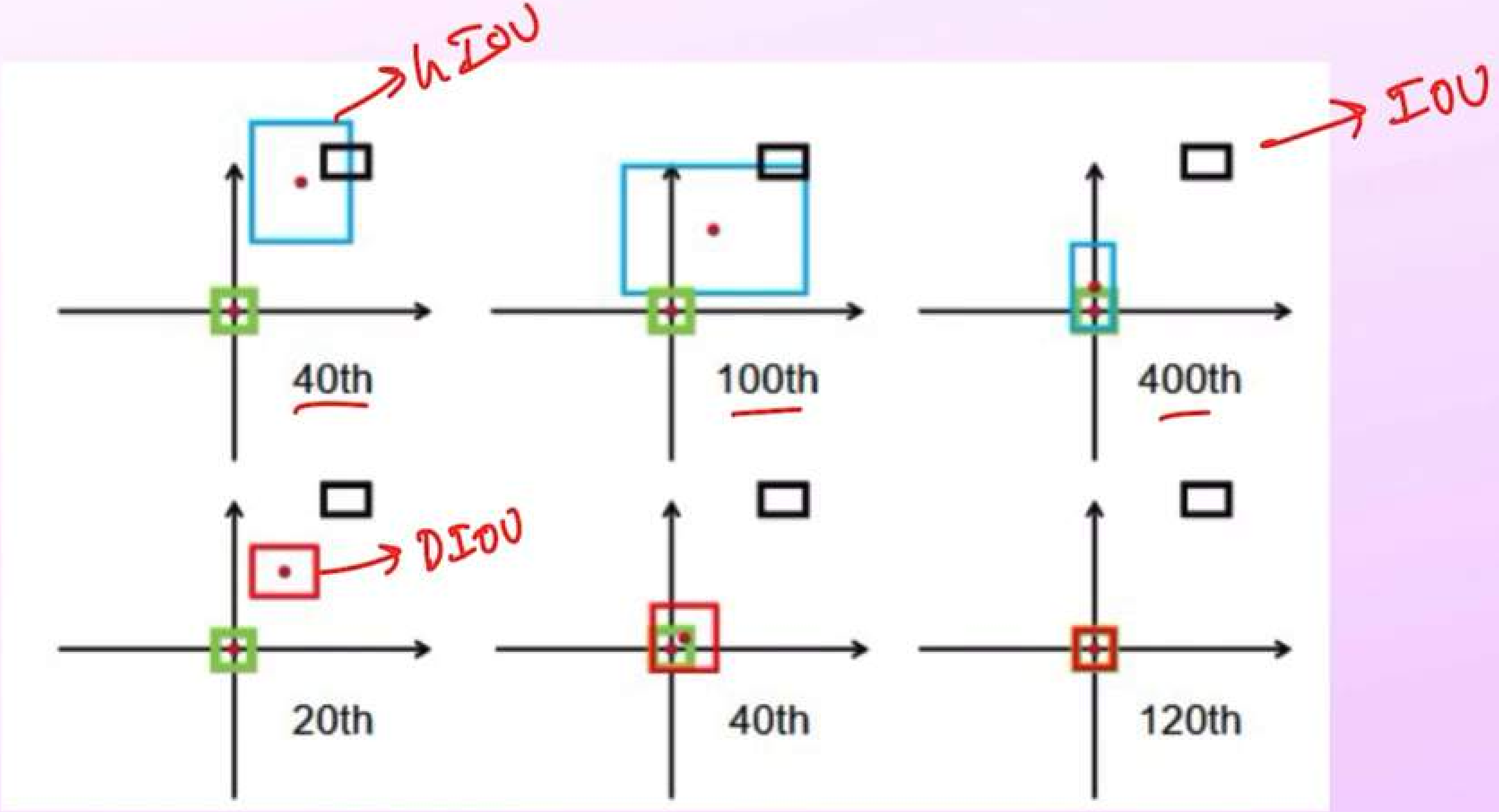
Figure 5: DIoU loss for bounding box regression, where the normalized distance between central points can be directly minimized. c is the diagonal length of the smallest enclosing box covering two boxes, and $d = \rho(\mathbf{b}, \mathbf{b}^{gt})$ is the distance of central points of two boxes.

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}$$

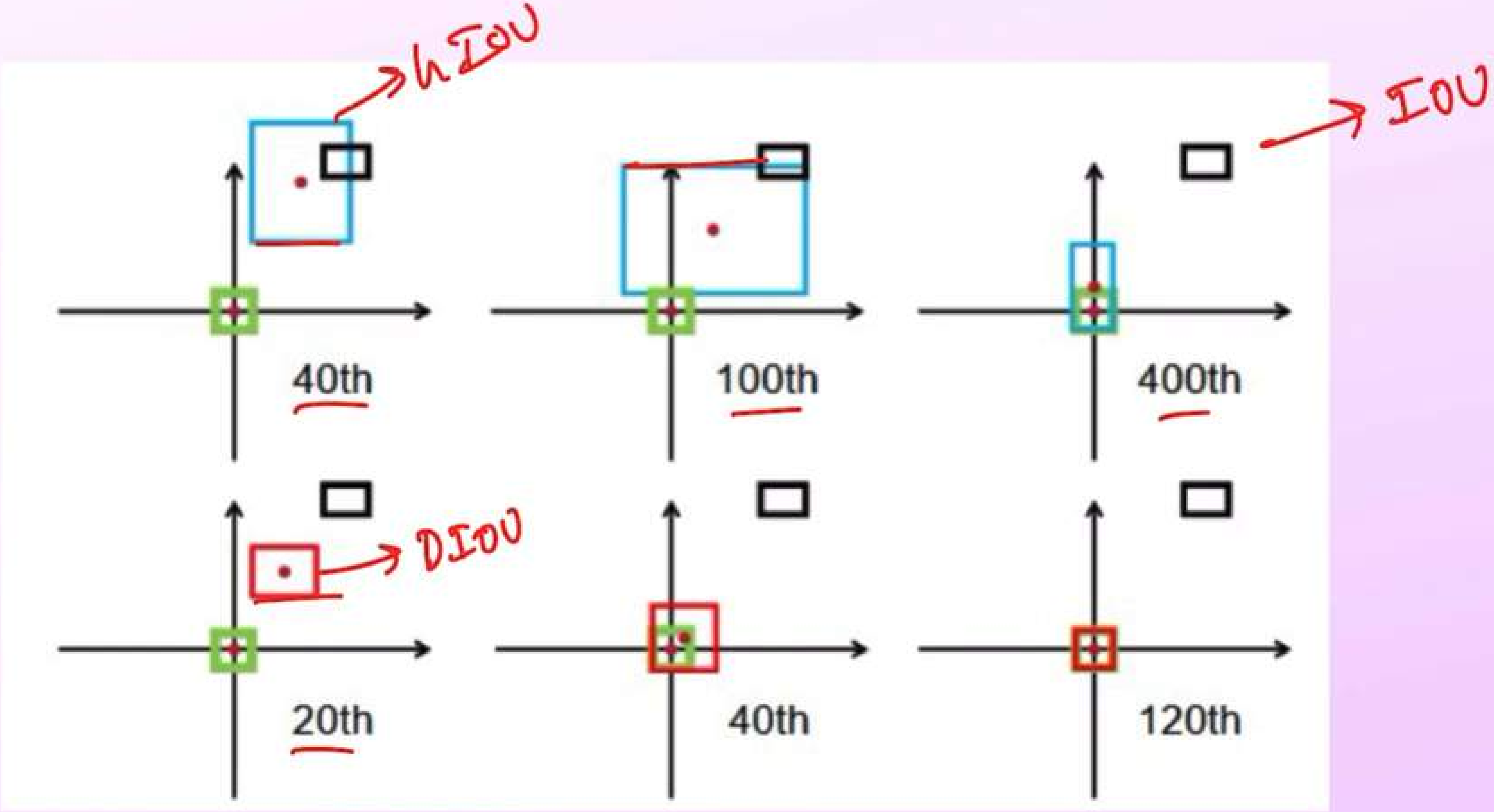
Distance IOU Loss



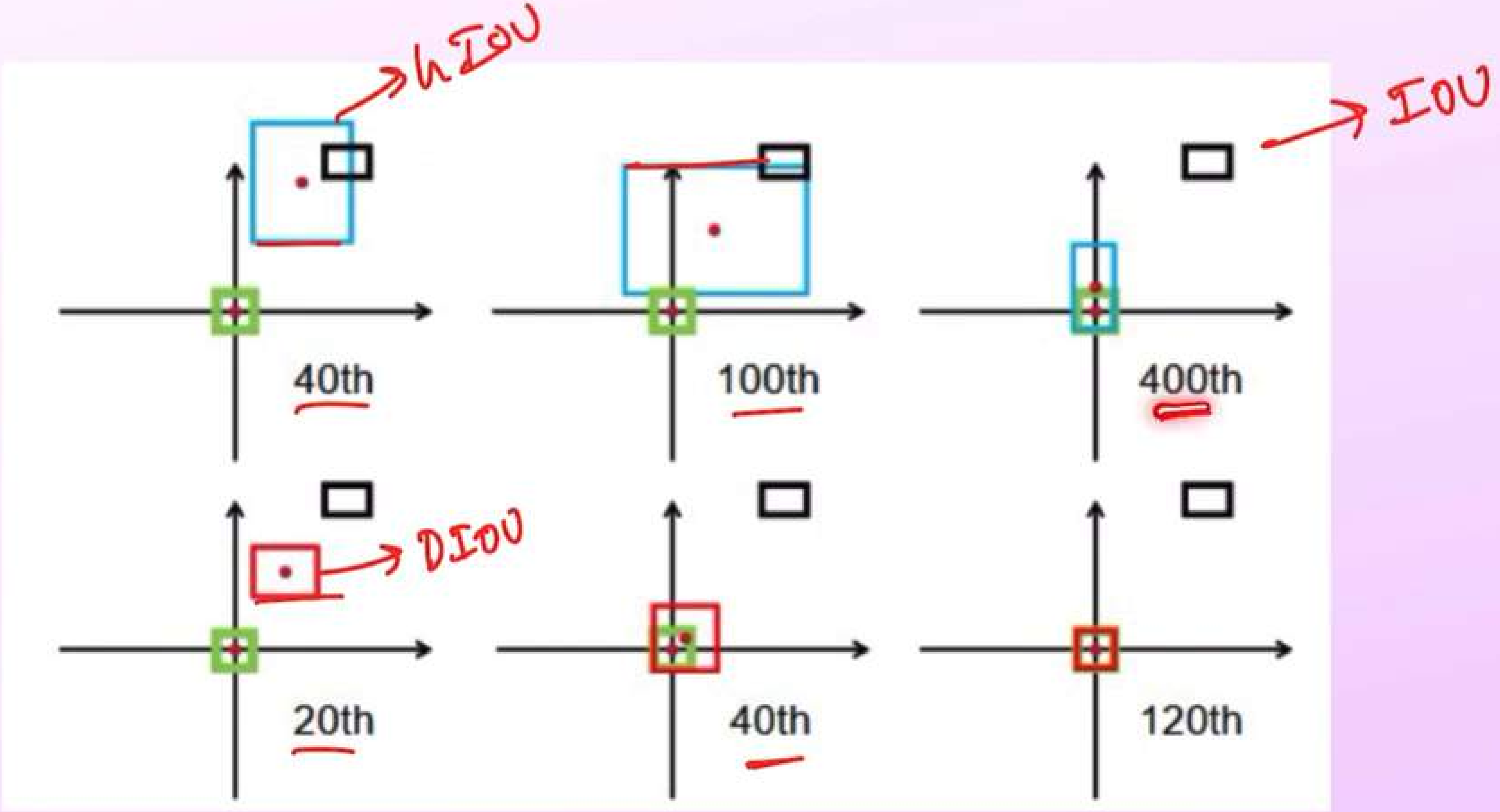
Distance IOU Loss



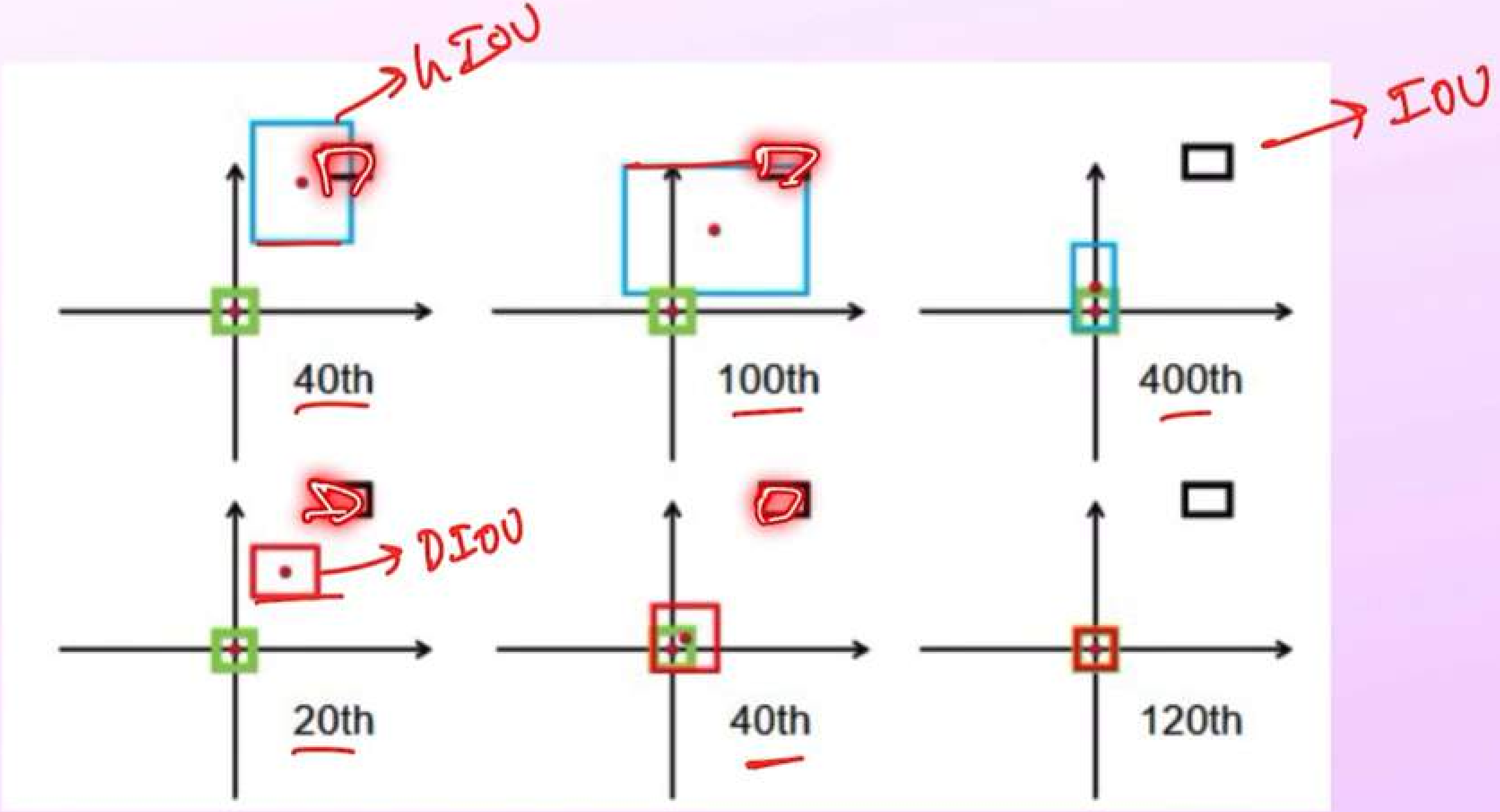
Distance IOU Loss



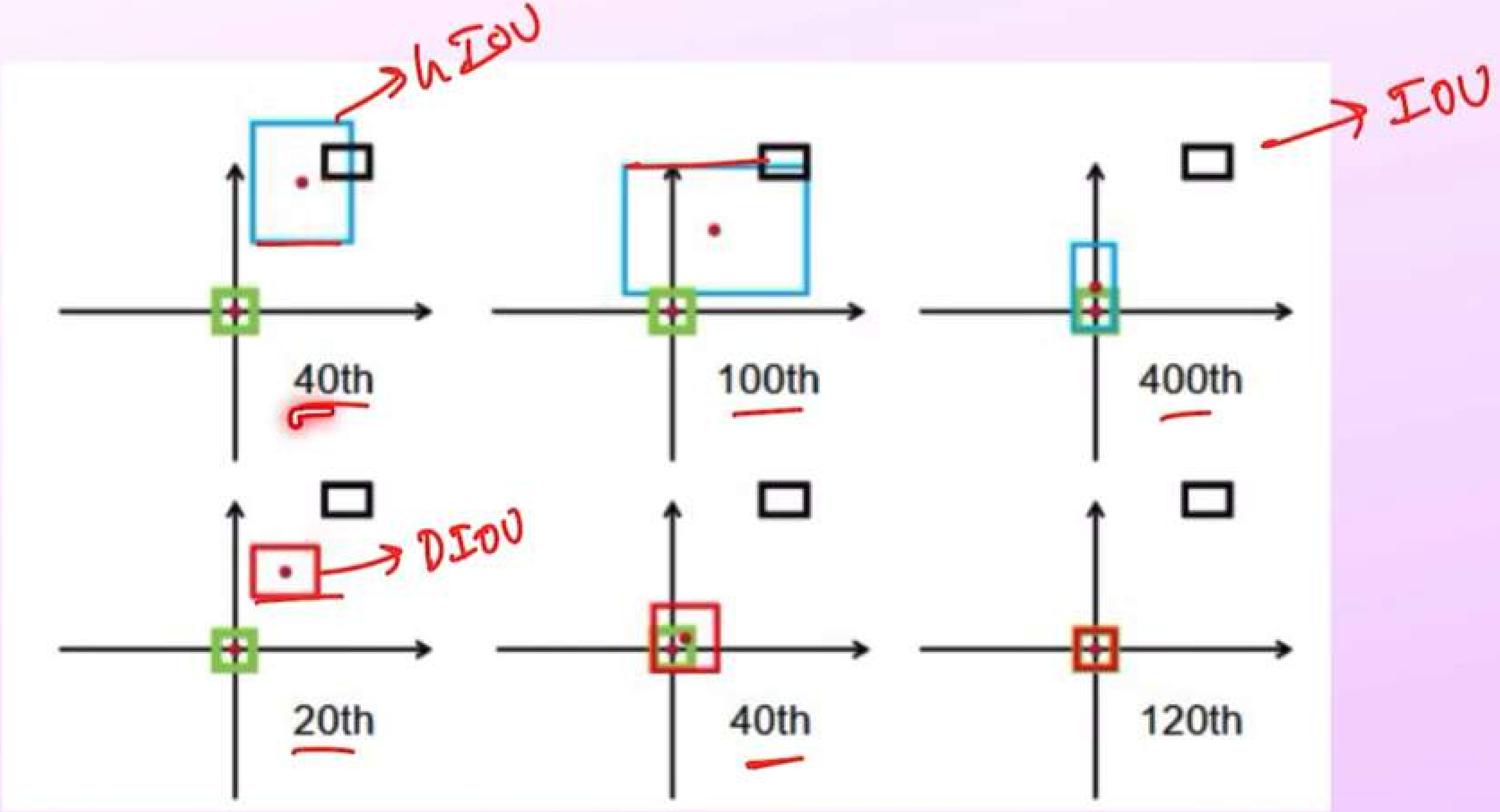
Distance IOU Loss



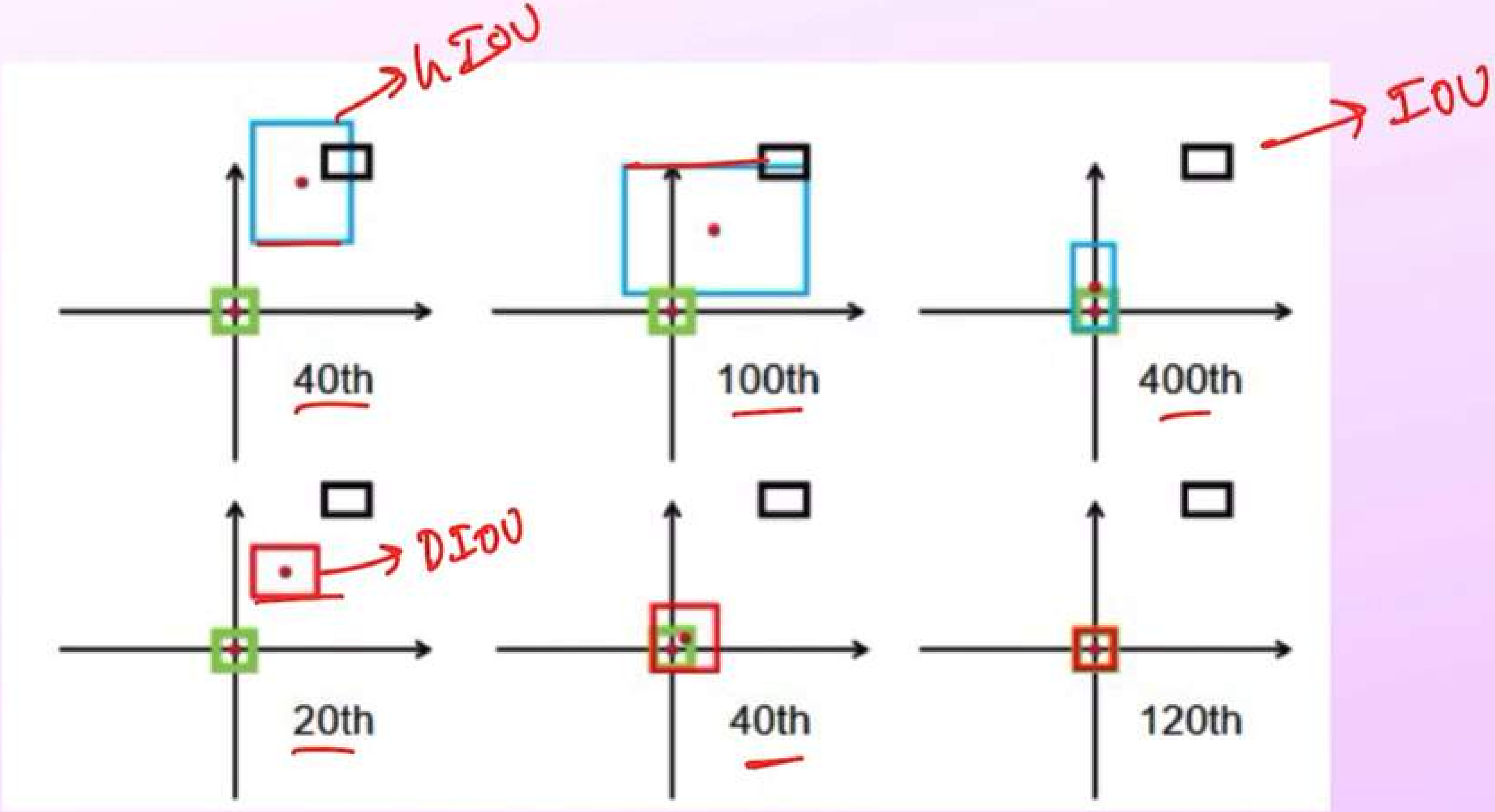
Distance IOU Loss



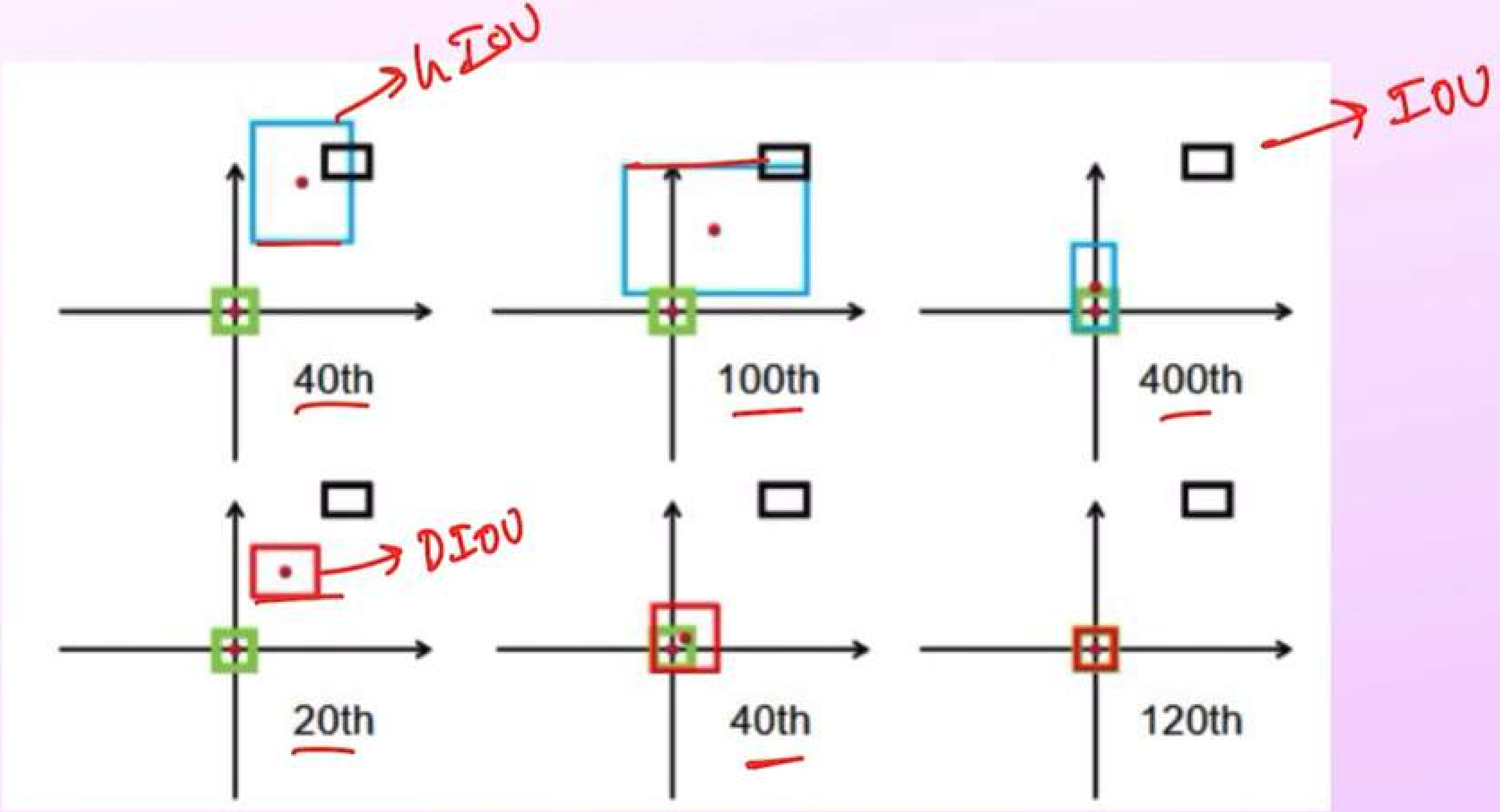
Distance IOU Loss



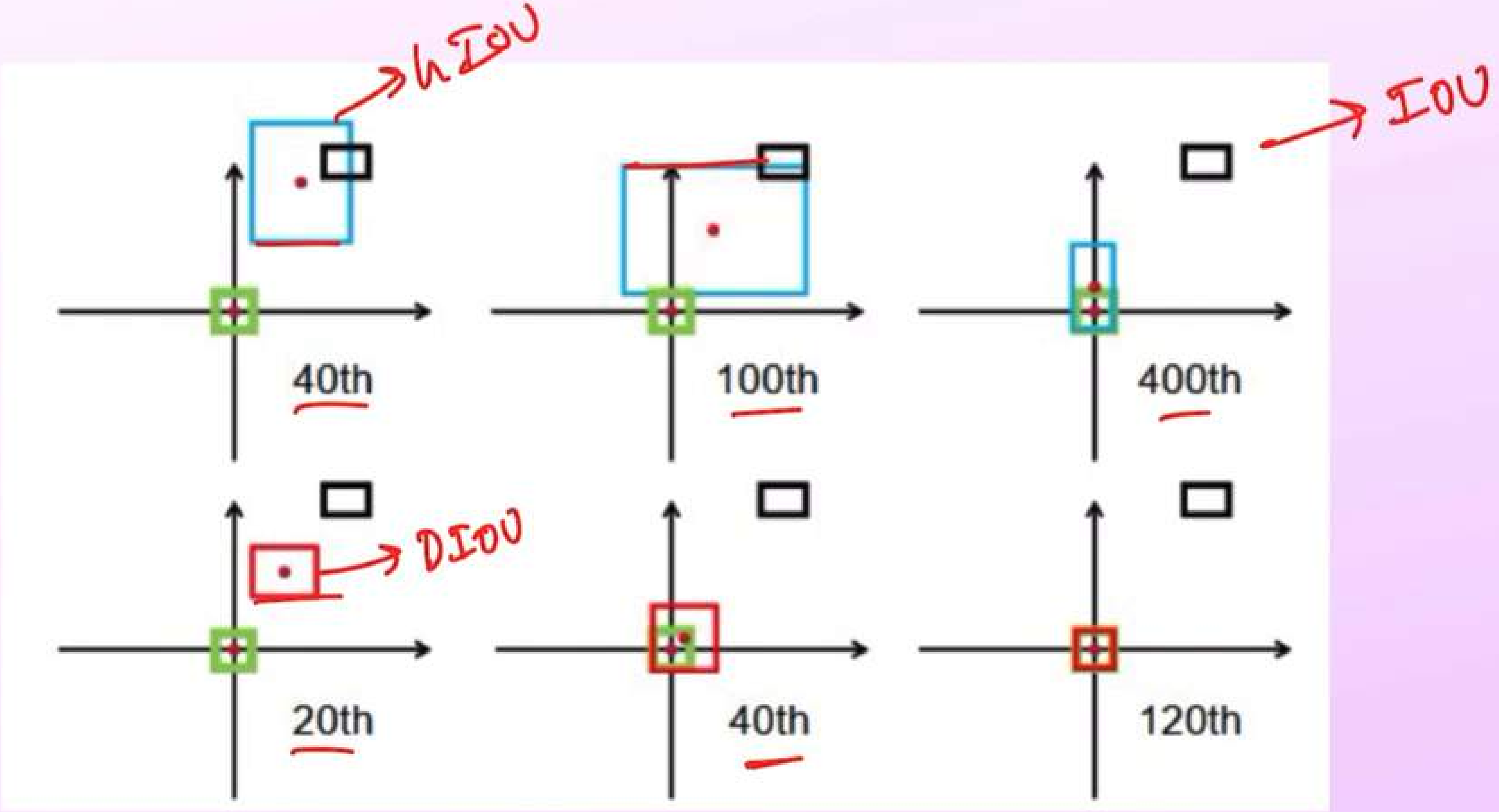
Distance IOU Loss



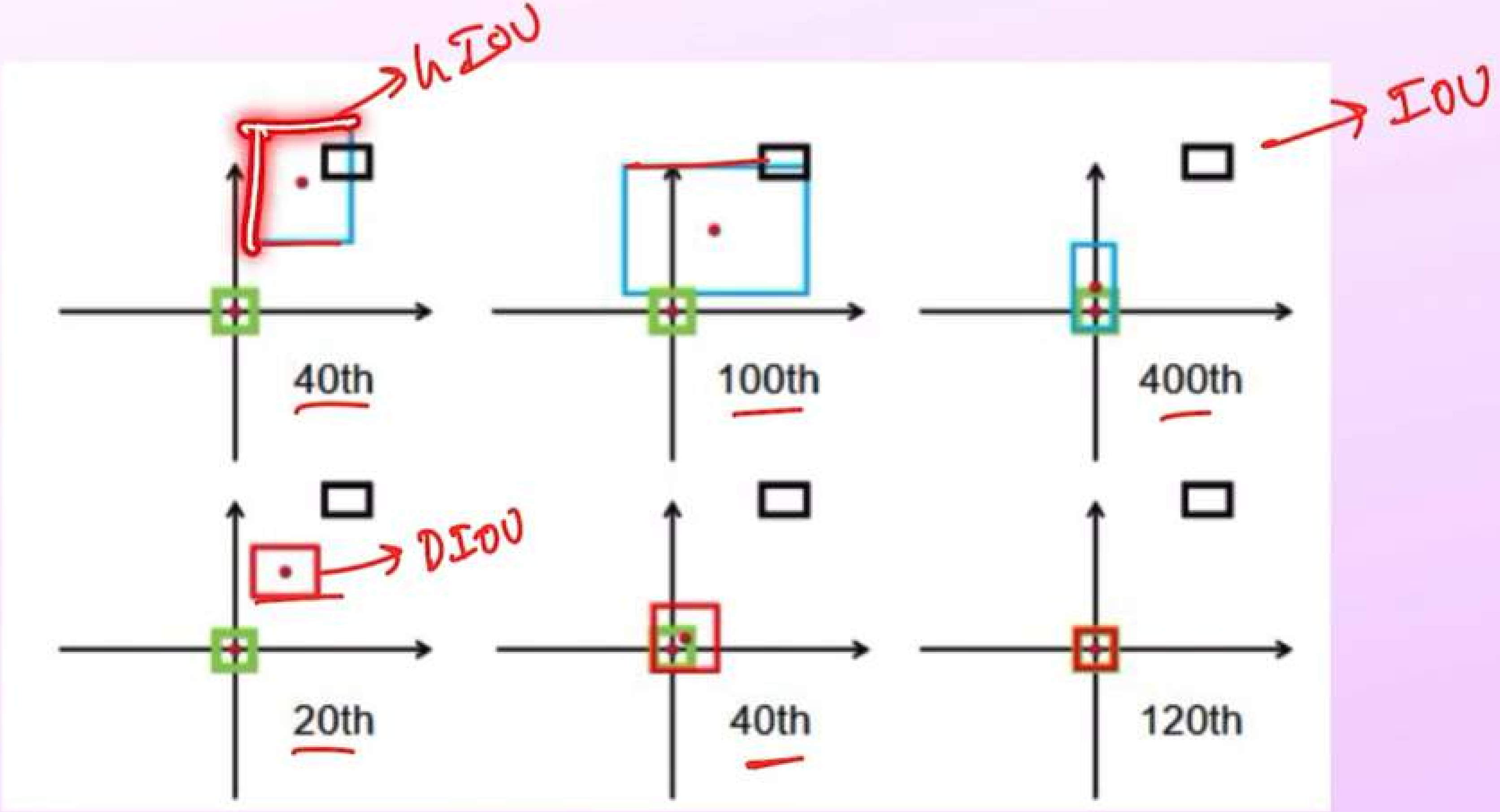
Distance IOU Loss



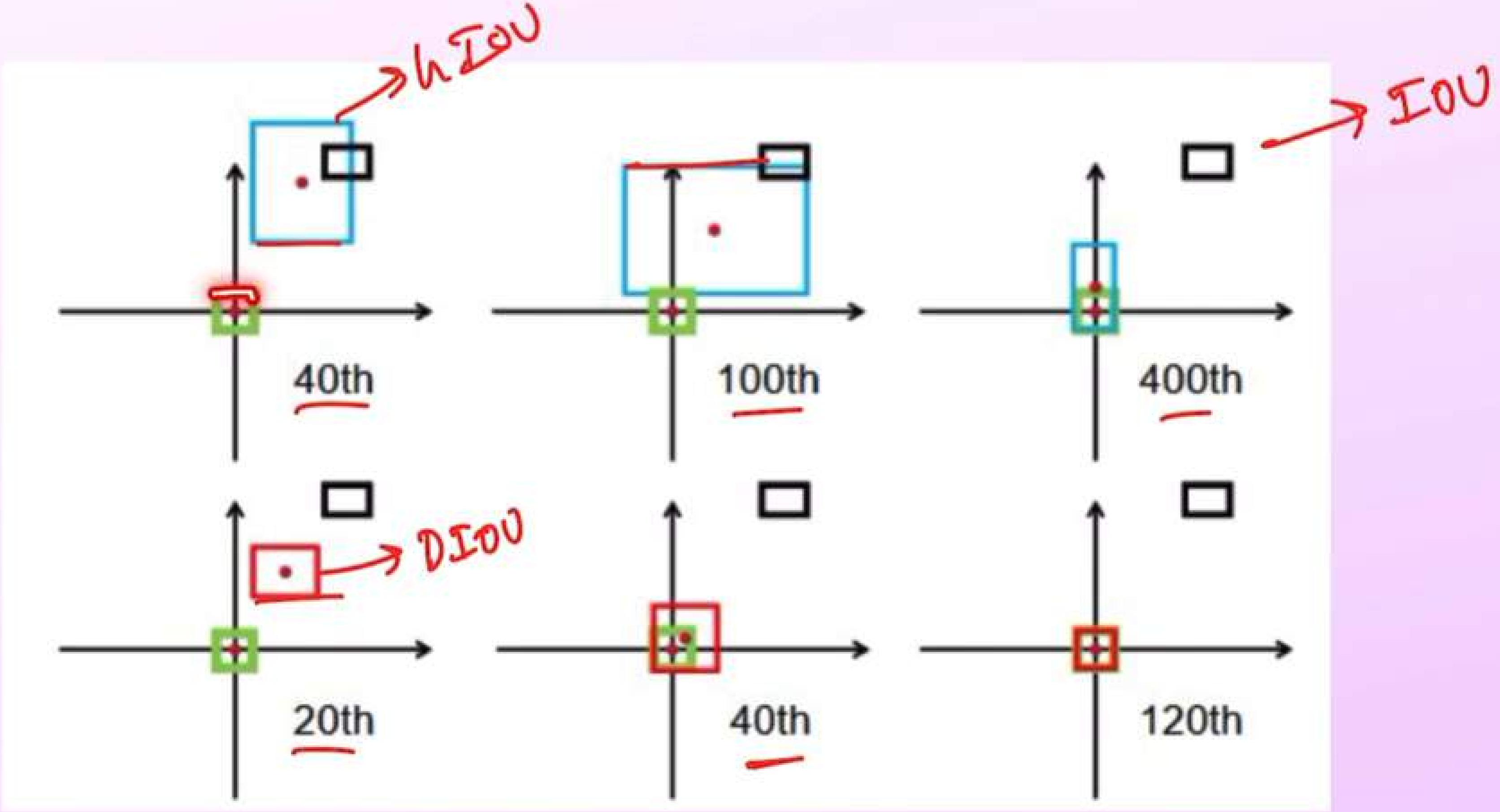
Distance IOU Loss



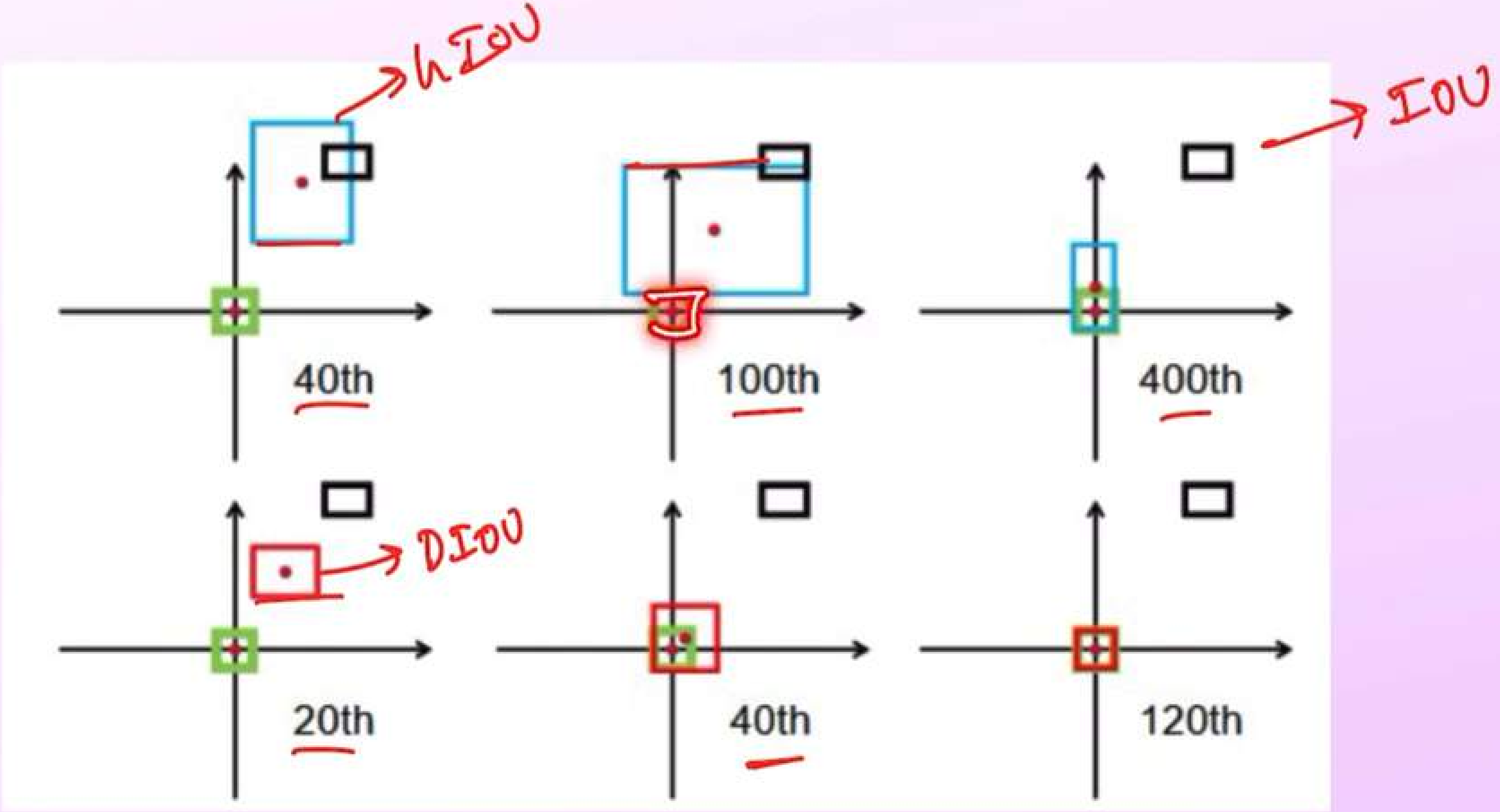
Distance IOU Loss



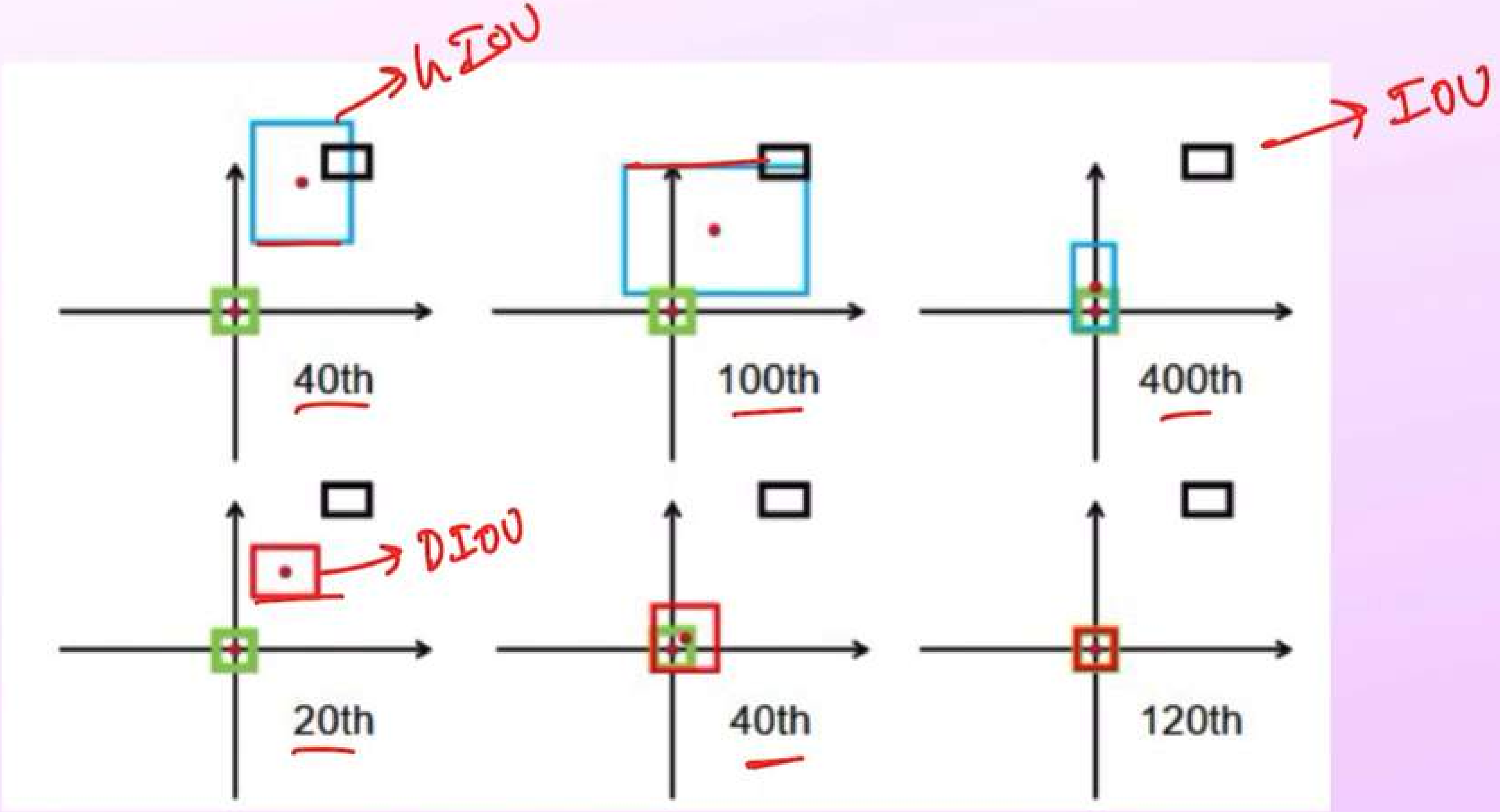
Distance IOU Loss



Distance IOU Loss



Distance IOU Loss



Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

$$v = \frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

IOU

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

$$v = \frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

IOU Loss

Distance penalty

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

$$v = \frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - \underbrace{IoU}_{\text{IOU Loss}} + \frac{\rho^2(b, b^{gt})}{c^2}$$

Distance penalty

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

Aspect ratio

$$v = \frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - \underbrace{IoU}_{\text{IOU Loss}} + \frac{\rho^2(b, b^{gt})}{c^2}$$

→ Distance penalty

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

→ Aspect ratio.

Width Height

$$v = \frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

IOU Loss

Distance penalty

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

Aspect ratio.

Width Height

$$v = \frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

IOU Loss

→ Distance penalty

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

→ Aspect ratio.

Width Height

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

IOU Loss

Distance penalty

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

Aspect ratio.

Width Height

$$v = \frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - \underbrace{IoU}_{\text{IOU Loss}} + \frac{\rho^2(b, b^{gt})}{c^2}$$

→ Distance penalty

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

→ Aspect ratio.

Width Height

$$v = \frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

Distance penalty

IOU Loss

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

Aspect ratio.

Width Height

$$v = \frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

Distance penalty

IOU Loss

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

Aspect ratio

Width Height

$$v = \frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

Distance penalty

IOU loss

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

Aspect ratio.

Width Height

$$v = \frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

→ Distance penalty

IOU Loss

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

→ Aspect ratio

Width Height

$$v = \frac{4}{\pi^2}(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2$$

$$\alpha = \frac{v}{(1 - IoU) + v}$$

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

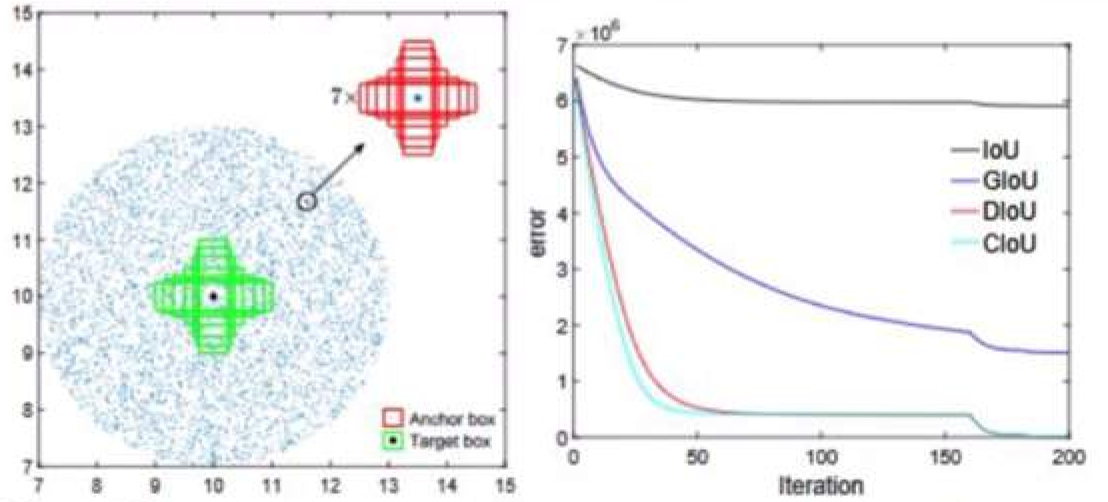


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

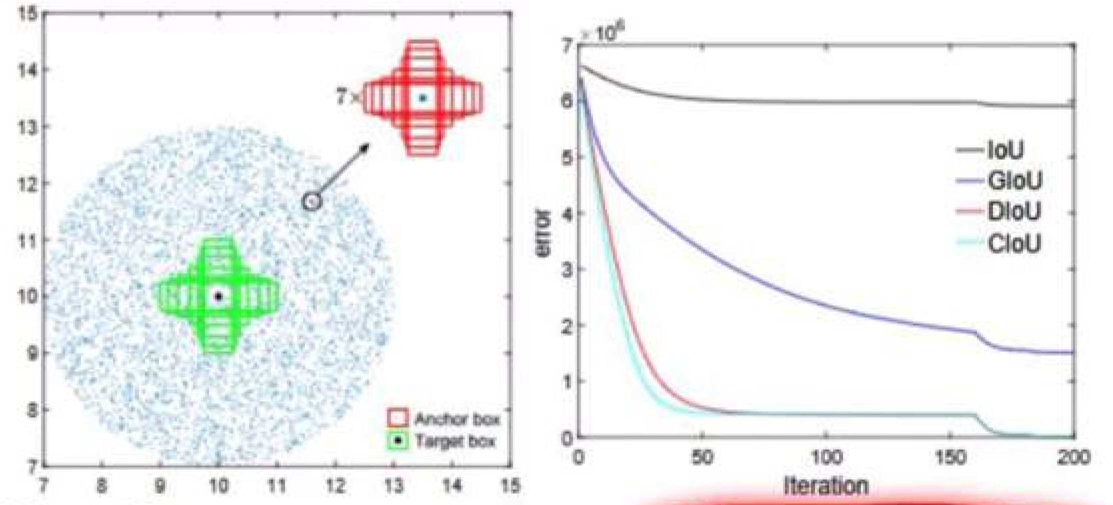


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

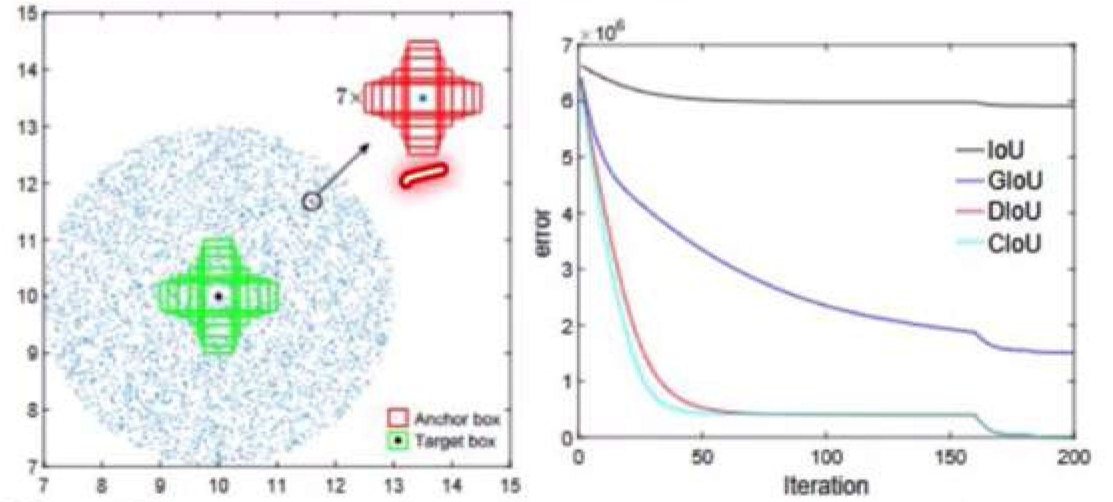


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

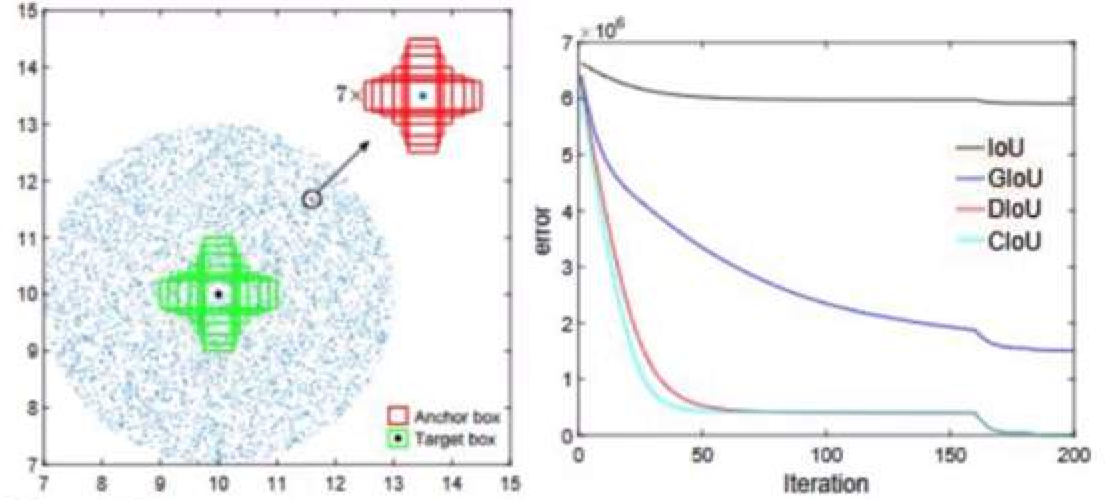


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

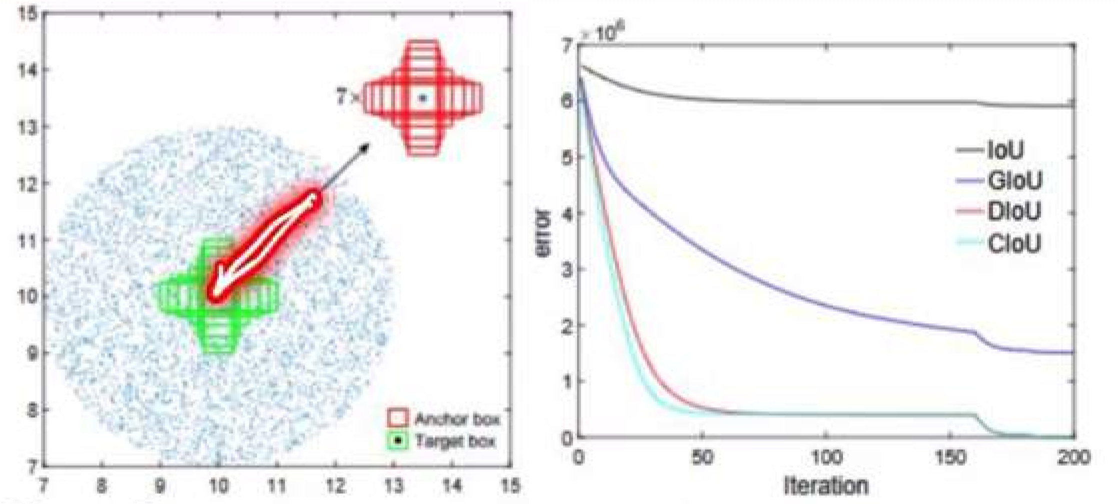


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

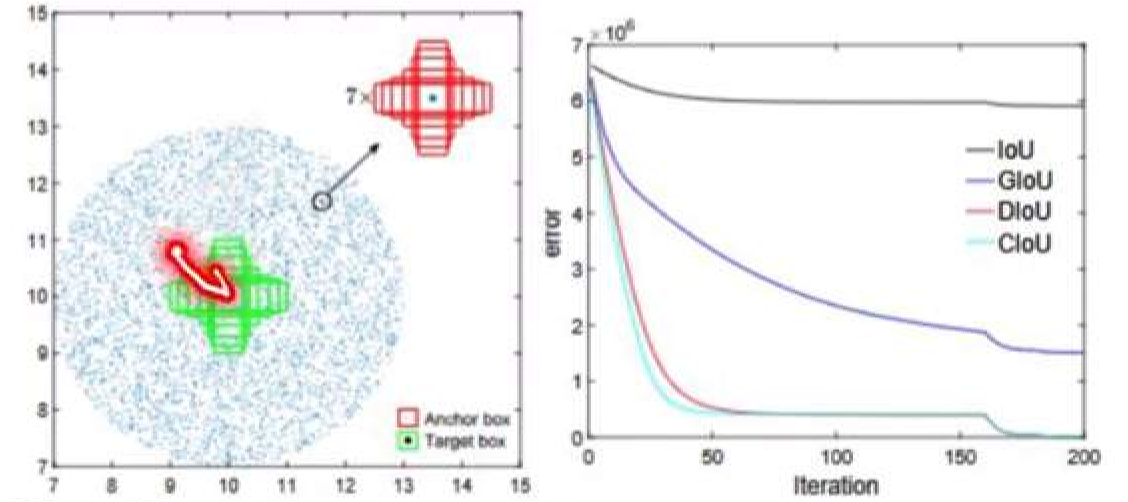


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

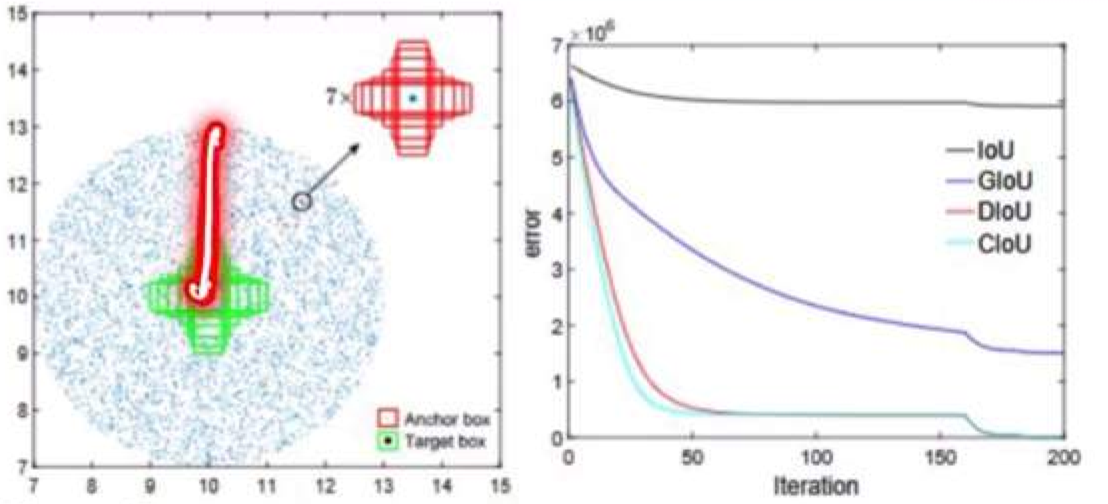


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

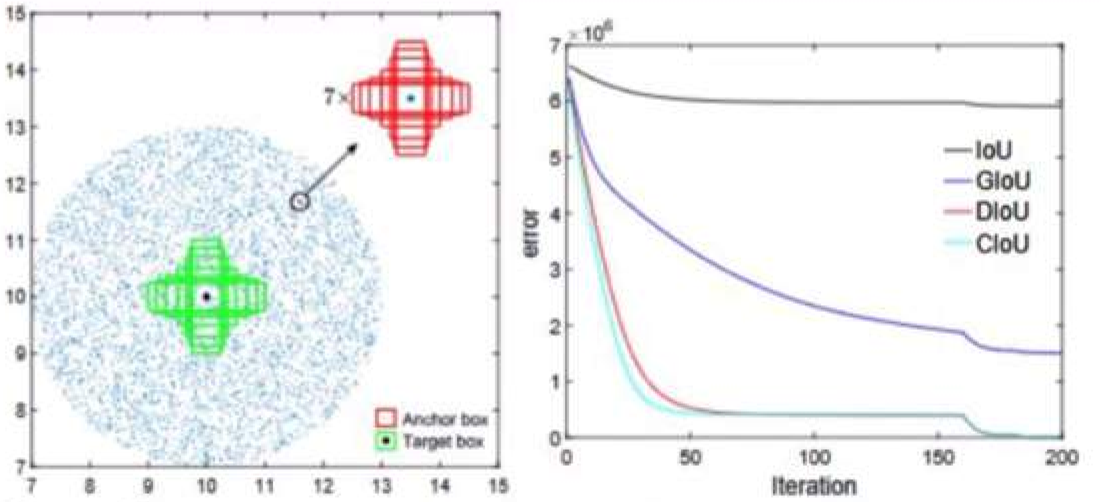


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

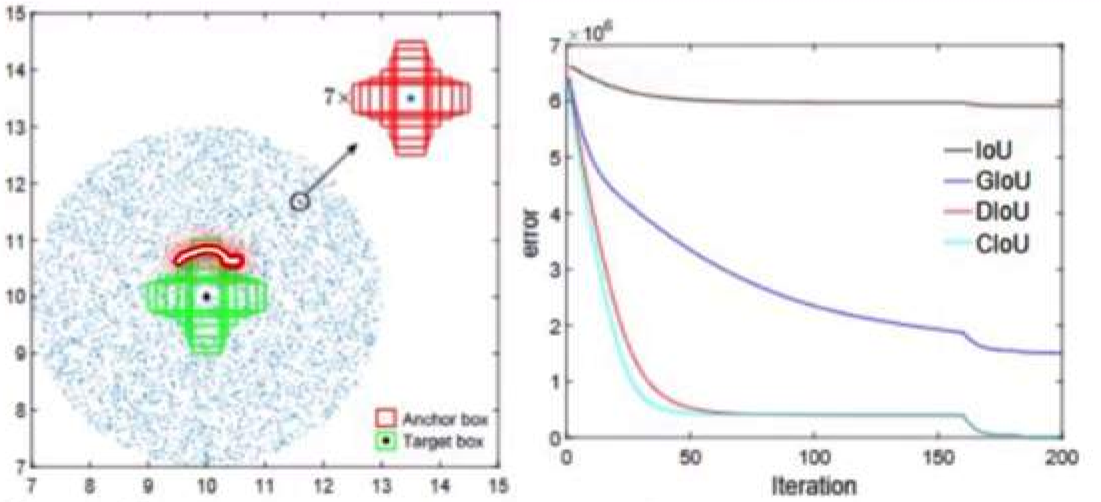


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

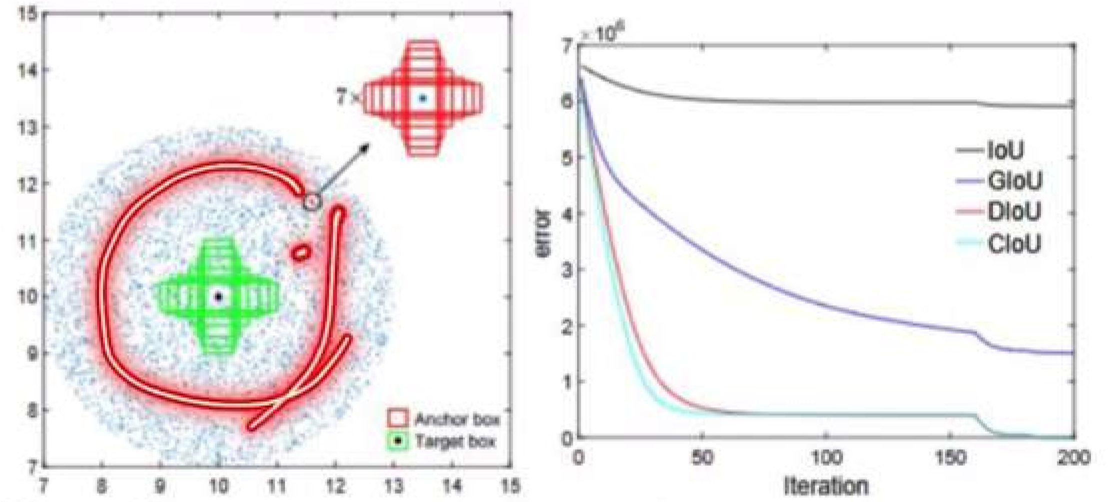


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

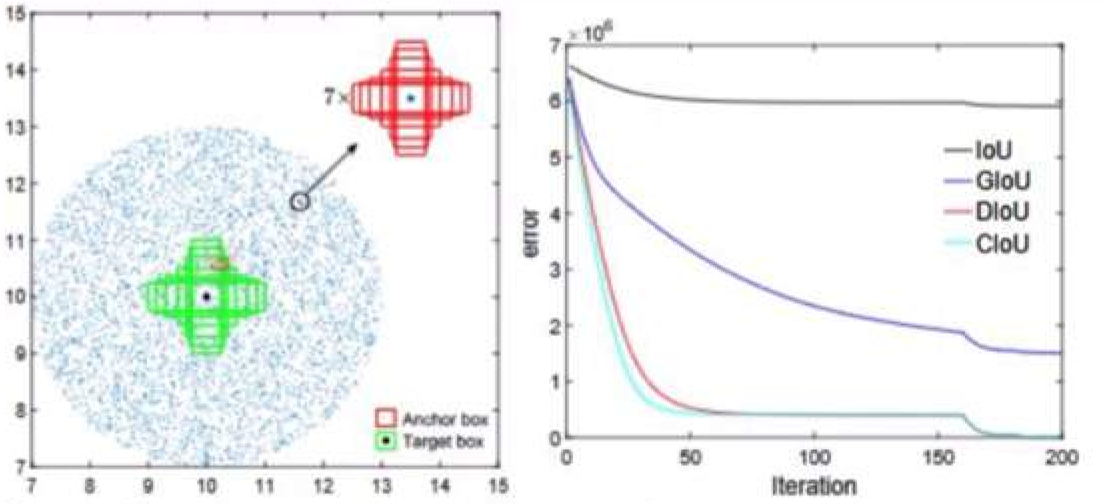


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t,n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

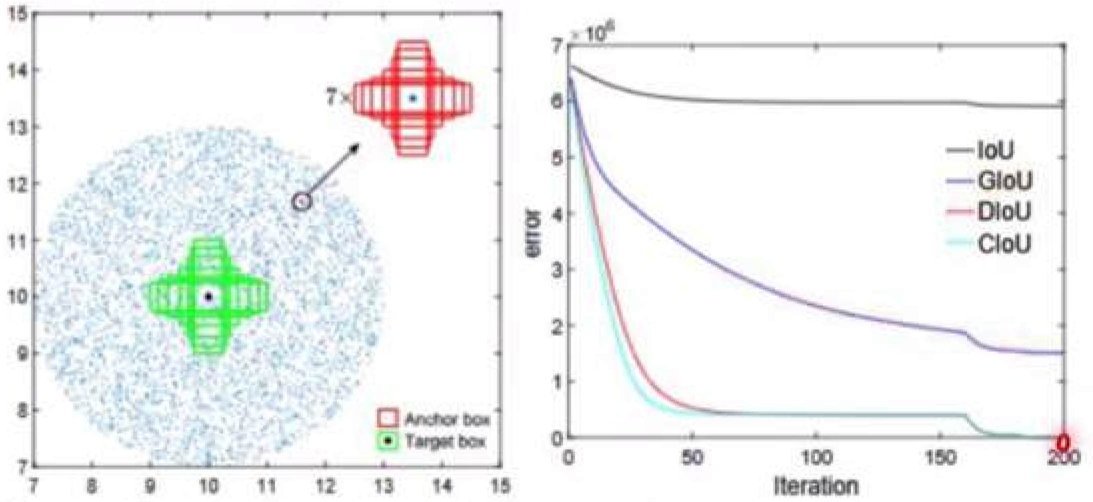


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

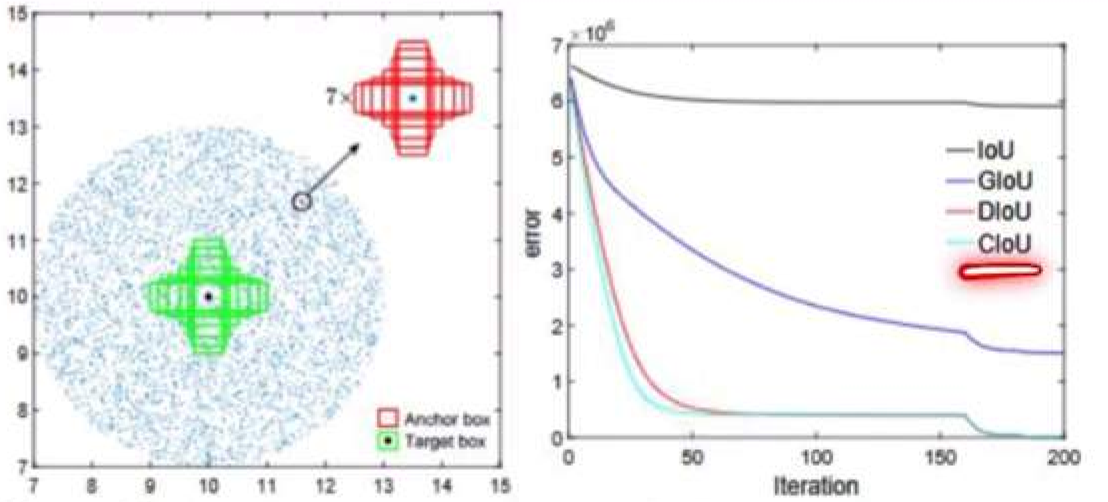


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

Complete IOU Loss

- Overlapping Area
- Distance between centers
- Aspect ratios

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

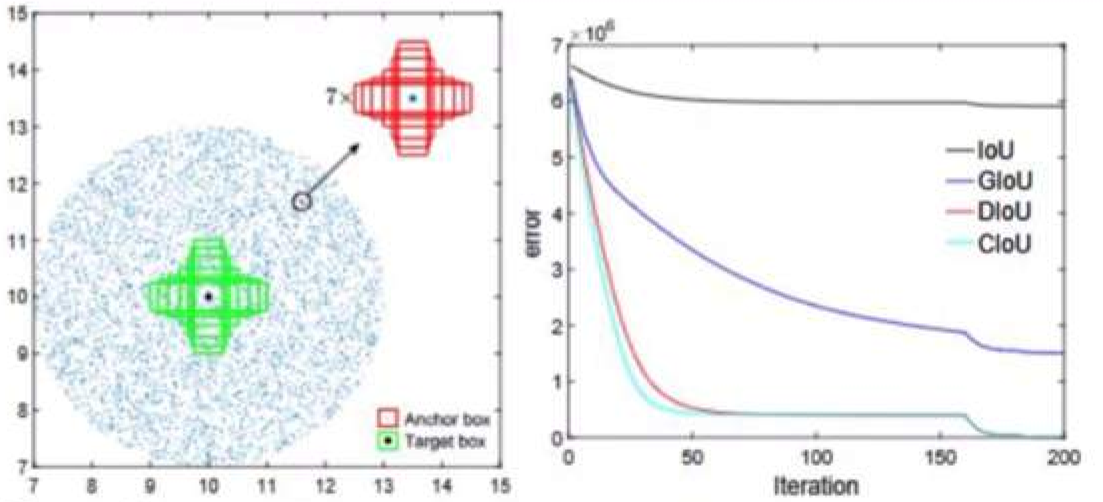


Figure 3: Simulation experiments: (a) 1,715,000 regression cases are adopted by considering different distances, scales and aspect ratios, (b) regression error sum (i.e., $\sum_n \mathbf{E}(t, n)$) curves of different loss functions at iteration t.

In this Video

	Backbone	Detector
Bag of Freebies (BoF)	• CutMix • Mosaic data augmentation • DropBlock • Class label smoothing	• CIoU-loss • Cross mini-Batch Normalization • DropBlock • Mosaic data augmentation • Self-Adversarial Training • Multiple anchors for a single ground truth • Cosine annealing scheduler • Optimal hyperparameters • Random training shapes
Bag of Specials (BoS)	• Multi-input weighted residual connections (MiWRC)	• DIoU-NMS

In this Video

	Backbone	Detector
Bag of Freebies (BoF)	• CutMix • Mosaic data augmentation • DropBlock • Class label smoothing	• CIoU-loss • Cross mini-Batch Normalization • DropBlock • Mosaic data augmentation • Self-Adversarial Training • Multiple anchors for a single ground truth • Cosine annealing scheduler • Optimal hyperparameters • Random training shapes
Bag of Specials (BoS)	• Multi-input weighted residual connections (MiWRC)	• DIoU-NMS

In this Video

	Backbone	Detector
Bag of Freebies (BoF)	• CutMix • Mosaic data augmentation • DropBlock • Class label smoothing	• CIoU-loss • Cross mini-Batch Normalization • DropBlock • Mosaic data augmentation • Self-Adversarial Training • Multiple anchors for a single ground truth • Cosine annealing scheduler • Optimal hyperparameters • Random training shapes
Bag of Specials (BoS)	• Multi-input weighted residual connections (MiWRC)	• DIoU-NMS