About small amounts of data

김 현 성

Vision & Display Systems Lab.

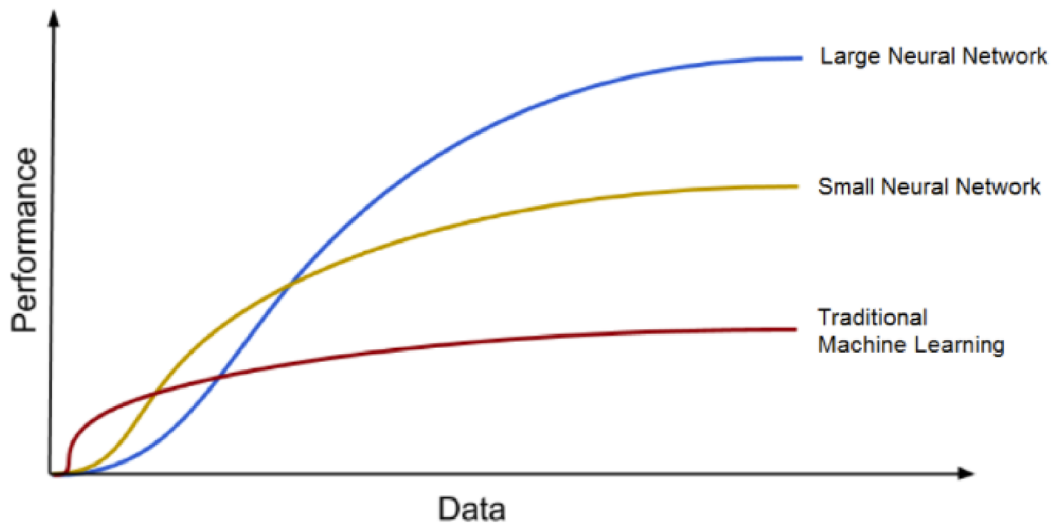
Dept. of Electronic Engineering, Sogang University

Outline

- Introduction
  - Data and Deep learning
- How to deal with?
  - "Small" diversity of data
  - "Small" in some classes
  - Just "Small"
- Summary
- Reference

Introduction

- Data and Deep learning



"The analogy to deep learning is that the rocket engine is the deep learning models and the ***fuel is the huge amounts of data*** *we can feed to these algorithms."*

-Andrew Ng-

Introduction

- Data and Deep learning
  - Limitation
    - Annotation cost
      - Labels(classification) : COCO[1]
        118K images → 11.1K hours
      - Masks(instance segmentation) : COCO[1]
        860K masks → 30.0K hours
      - Captions(image captioning) : nocaps[2]
        118K images → 6.5K hours
    - Privacy
      - **Medical image, industrial image,** etc.

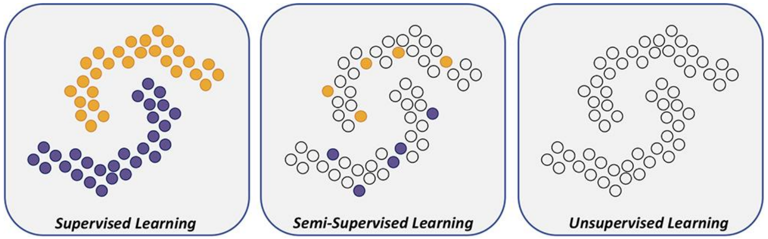
"Hard to get data"

↓

"Unsatisfactory data"

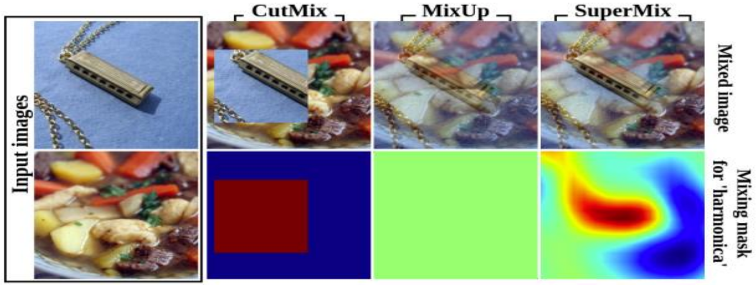
How to deal with?

- Data limitation
  - Learning strategy
    - Semi-supervised, unsupervised, etc.



Labeled data proportion for each learning strategies

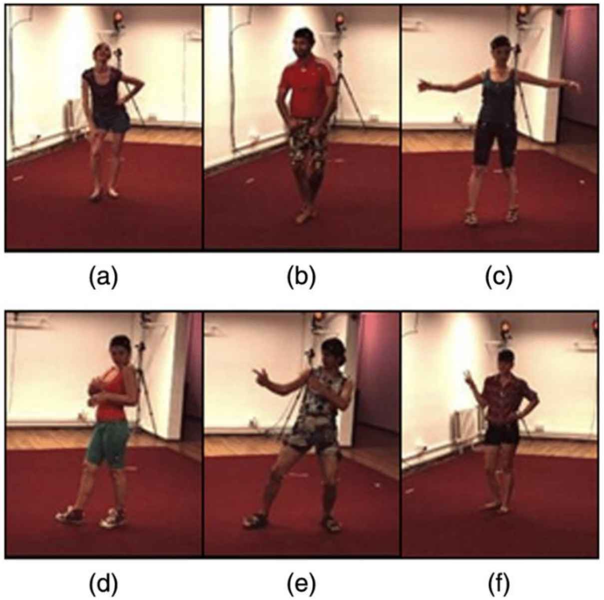
  - Data augmentation
    - Data mixing, affine transform, etc.



Examples of data mixing

How to deal with?

- "Small" diversity of data – PoseAug[3] (CVPR 2021, Oral)
  - Motivation
    - Annotation of **3D human pose estimation** is implemented using '**motion capture**'
      - Hard to get data → Low diversity → **Hard to generalize** to new datasets
    - Offline-manner augmentation has limitation about data diversity
      - **Bio-mechanical rules → many pre-defined rules**
  - Contribution
    - **Differentiable**(online) augmentor that generates **diverse data**
    - By using **discriminator**, the augmentor generates **realistic data**
    - 3D pose estimation network became to get better **generalization property** as well as improve its performance

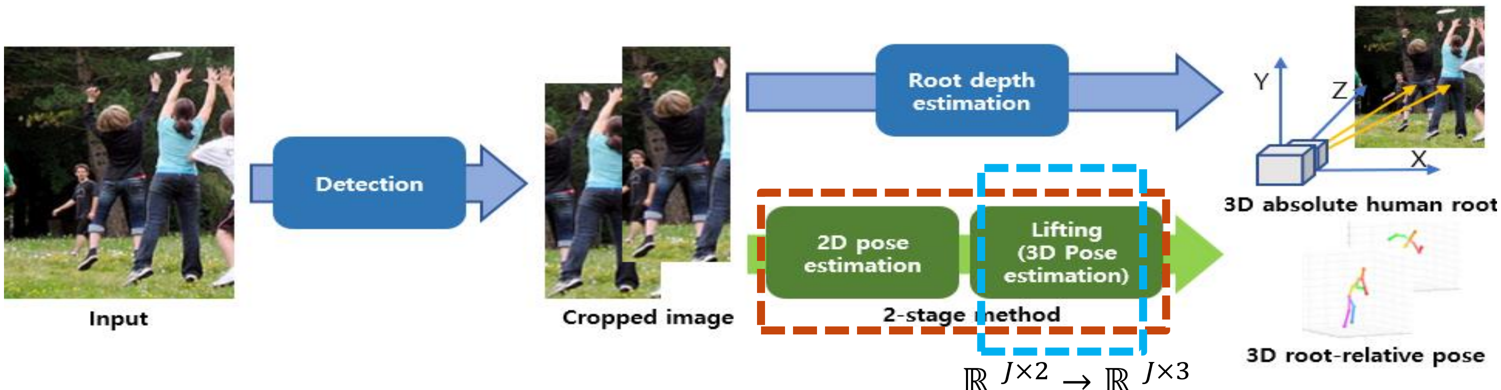


Examples of 3D human keypoint dataset

How to deal with?

- “Small” diversity of data – PoseAug[3] (CVPR 2021, Oral)
  - Background
    - 1-stage method

    - 2-stage method



How to deal with?

$$\underbrace{\begin{bmatrix} x_1 & x_2 & & x_j \\ y_1 & y_2 & \cdots & y_j \\ z_1 & z_2 & & z_j \end{bmatrix}}_{\boldsymbol{X}\ \text{(3d points)}} \times \underbrace{\begin{bmatrix} 0 & 0 & & 0 \\ 1 & 1 & & 0 \\ 0 & 0 & & 1 \\ . & . & & 0 \\ . & . & \cdots & -1 \\ . & 0 & & 0 \\ 0 & -1 & & . \\ -1 & 0 & & . \\ 0 & . & & . \end{bmatrix}}_{\boldsymbol{C}} = \boldsymbol{B}\ \text{(bone vectors)}$$

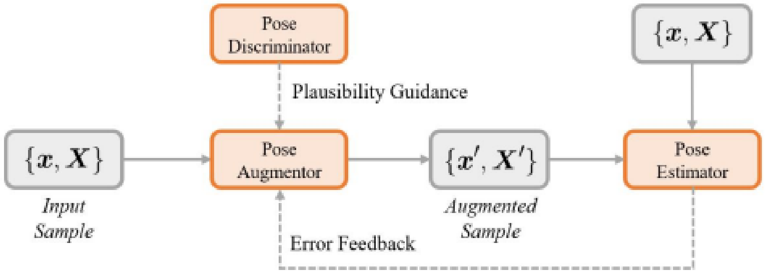
- "Small" diversity of data – PoseAug[3] (CVPR 2021, Oral)
  - Background
    - KCS(kinematic chain space)
      - Transform method between 3D keypoint coordinate and bone vector
        - ✓ $\boldsymbol{b}_k = \boldsymbol{p}_r - \boldsymbol{p}_t = \boldsymbol{X}\boldsymbol{c}$, $\boldsymbol{c} = (0, \ldots, 0, 1, 0, \ldots, 0, -1, 0, \ldots., 0)^T$
        - ✓ 3D pose → $\boldsymbol{X} \in \mathbb{R}^{3\times j}$, bone vectors $\boldsymbol{B}\left(\in \mathbb{R}^{3\times(j-1)}\right) = (\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_{j-1})$
      - Decomposition of bone vectors ***B***
        - ✓ $\widehat{\boldsymbol{B}}(\in \mathbb{R}^{(j-1)\times 3})$: unit vectors of bone vectors → **angle information**
        - ✓ $||\boldsymbol{B}||$ $(\in \mathbb{R}^{(j-1)\times 1})$: L2 norm of each bone vector → **length information**
      - Inverse conversion to 3D keypoints coordinates
        - ✓ $\boldsymbol{X} = \Phi^{-1}(\boldsymbol{B})$

How to deal with?

- “Small” diversity of data – PoseAug[3] (CVPR 2021, Oral)
  - Method
    - Pipeline
      - Formulation

$$min_{\theta} max_{\theta_A} L_P(P_{\theta}, A_{\theta_A}(\chi)),$$

where P: $estimation\ network$, A: $augmentor$, L_P: $criterion$, χ: $(\boldsymbol{x}, \boldsymbol{X}) \leftrightarrow (\text{2D}, \text{3D})$ pair



Overview of framework

      - Feedback loss

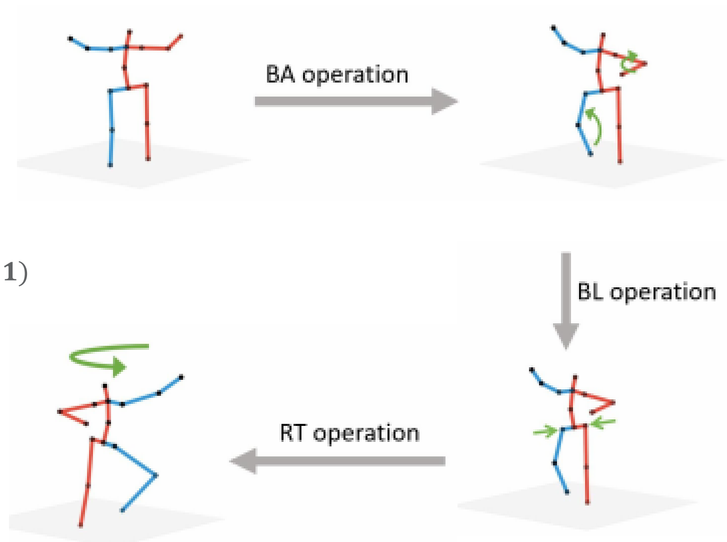
$$L_{fb} = |1.0 - \exp[L_P(\boldsymbol{X}') - \beta L_P(\boldsymbol{X})]|,$$

where $\boldsymbol{X}'$ represents the augmented data

How to deal with?

- Decomposition of bone vectors $\boldsymbol{B}$
  - ✓$\widehat{\boldsymbol{B}}(\in \mathbb{R}^{(j-1)\times 3})$: unit vectors of bone vectors → **angle information**
  - ✓$\|\boldsymbol{B}\|$ $(\in \mathbb{R}^{(j-1)\times 1})$: L2 norm of each bone vector → **length information**

- "Small" diversity of data – PoseAug[3] (CVPR 2021, Oral)
  - Method
    - Augmentation
      - BA operation
        - ✓$\widehat{B}' = \widehat{B} + \gamma_{ba}, \gamma_{ba} \in \mathbb{R}^{3\times(\boldsymbol{j}-\boldsymbol{1})}$
      - BL operation
        - ✓$\|B'\| = \|B\| \times (1 + \gamma_{bl}), \gamma_{bl} \in \mathbb{R}^{1\times(\boldsymbol{j}-\boldsymbol{1})}$
      - RT operation
        - ✓$X' = R\left[\Phi^{-1}(\boldsymbol{B}')\right] + \boldsymbol{t}$
      - Augmentation loss



$$\mathcal{L}_{reg}(\gamma) = \begin{cases} 0, & \text{if } \bar{\gamma} < threshold, \\ \|\gamma\|^2, & \text{otherwise,} \end{cases}$$

How to deal with?

- “Small” diversity of data – PoseAug[3] (CVPR 2021, Oral)
  - Method
    - Discriminator
      - Aim : To ensure the pose plausibility without sacrificing the diversity
      - Loss : LS-GAN loss

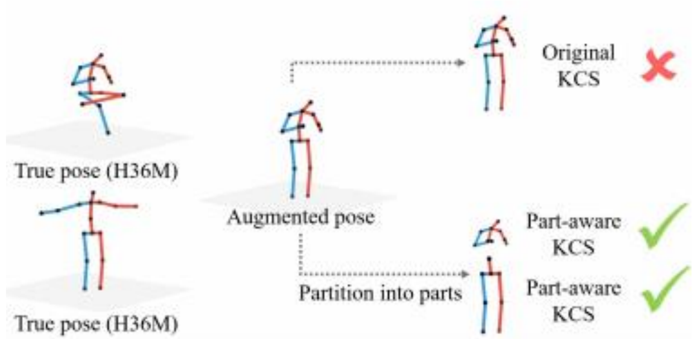
$$\mathcal{L}_{\mathcal{D}} = \mathbb{E}[(D_{3d}(\boldsymbol{X}) - 1)^2] + \mathbb{E}[D_{3d}(\boldsymbol{X}')^2]$$
$$+\mathbb{E}[(D_{2d}(\boldsymbol{x}) - 1)^2] + \mathbb{E}[D_{2d}(\boldsymbol{x}')^2],$$

      - Part-aware **KCS – 3D & 2D**
        - ✓ $KCS_{local}^{i} = \widehat{\boldsymbol{B}}_i^T \widehat{\boldsymbol{B}}_i$

          → encapsulate the **inter joint angle information**

        - ✓Torso, left arm, right arm, left leg, right leg

          → **5 part**



Part-aware KCS

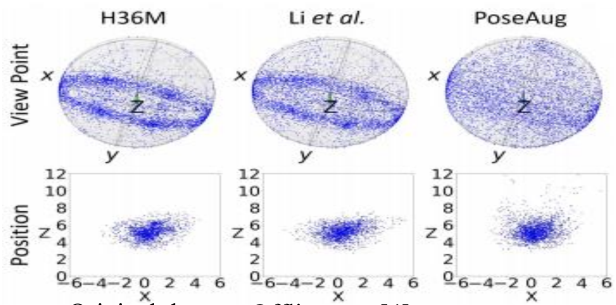
$$\begin{array}{r} \text{L shoulder}\\ \text{L elbow}\\ \text{L wrist}\end{array}\begin{bmatrix} x & y & z\\ x & y & z\\ x & y & z\end{bmatrix} \quad \begin{array}{c} \begin{array}{ccc}\text{L Shoulder} & \text{L elbow} & \text{L wrist}\end{array}\\ \begin{bmatrix} x & x & x\\ y & y & y\\ z & z & z\end{bmatrix}\end{array}$$

$\widehat{\boldsymbol{B}}_i^T$ $\qquad$ $\widehat{\boldsymbol{B}}_i$

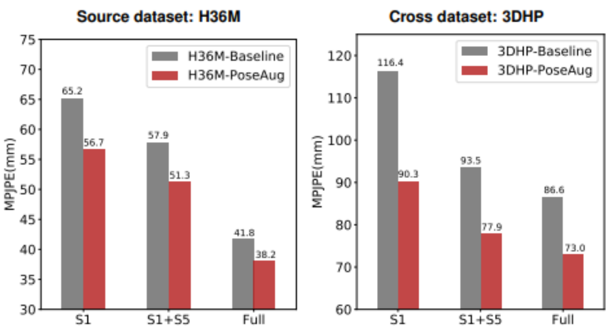
Input of discriminator : KCS_{local}^{i}

How to deal with?

- “Small” diversity of data – PoseAug[3] (CVPR 2021, Oral)
  - Results
    - Diversity



    - Cross dataset scenario

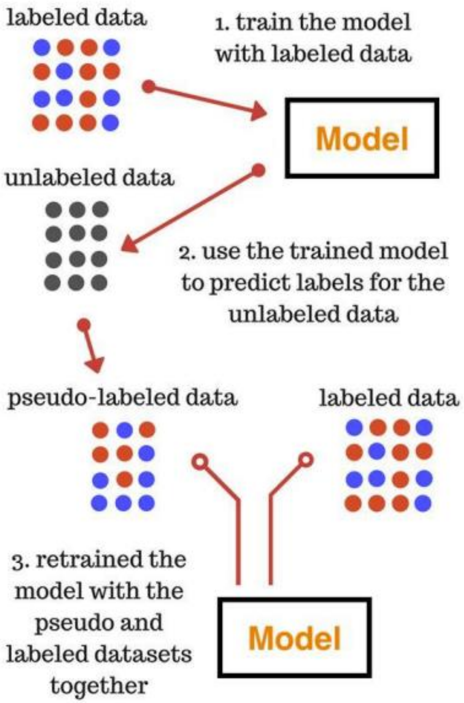


    - Performance improvement

	H36M				3DHP			
Method	DET	CPN	HR	GT	DET	CPN	HR	GT
SemGCN [52]	67.5	64.7	57.5	44.4	101.9	98.7	95.6	97.4
+ PoseAug	**65.2** (-2.3)	**60.0** (-4.8)	**55.0** (-2.5)	**41.5** (-2.8)	**89.9** (-11.9)	**89.3** (-9.4)	**89.1** (-6.5)	**86.1** (-11.2)
SimpleBaseline [26]	60.5	55.6	53.0	43.3	91.1	88.8	86.4	85.3
+ PoseAug	**58.0** (-2.5)	**53.4** (-2.2)	**51.3** (-1.7)	**39.4** (-3.9)	**78.7** (-12.4)	**78.7** (-10.1)	**76.4** (-10.1)	**76.2** (-9.1)
ST-GCN [3] (1-frame)	61.3	56.9	52.2	41.7	95.5	91.3	87.9	87.8
+ PoseAug	**59.8** (-1.5)	**54.5** (-2.4)	**50.8** (-1.5)	**36.9** (-4.8)	**83.5** (-12.1)	**77.7** (-13.6)	**76.6** (-11.3)	**74.9** (-12.9)
VPose [33] (1-frame)	60.0	55.2	52.7	41.8	92.6	89.8	85.6	86.6
+ PoseAug	**57.8** (-2.2)	**52.9** (-2.3)	**50.2** (-2.5)	**38.2** (-3.6)	**78.3** (-14.4)	**78.4** (-11.4)	**73.2** (-12.4)	**73.0** (-13.6)

How to deal with?

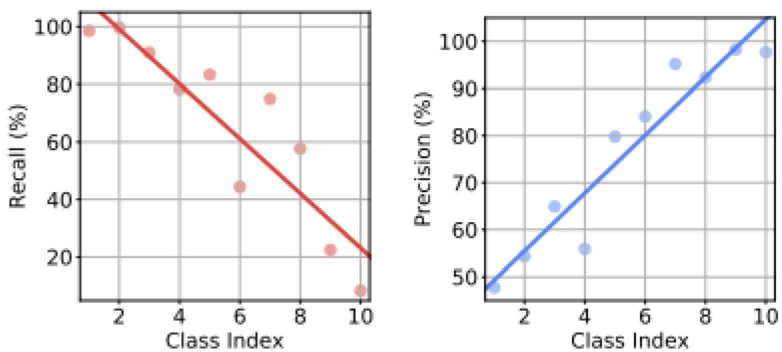
- “Small” in some classes: data imbalance while SSL – CReST[5] (CVPR 2021)
  - Semi-Supervised Learning(SSL)
    - Utilize unlabeled data to improve model performance
      - ‘**Self-training**’ is used widely in classification task
  - Problem
    - Model trained via SSL performs **poorly on class-imbalanced data**
      - Mainly due to **low recall on the minority class**
    - Pseudo-labels **generated by a biased model** trained are problematic
  - Assumption
    - Labeled and unlabeled have same distribution
      - Similarly imbalanced
    - Test-set is a class-balanced dataset



Self-training method

How to deal with?

- “Small” in some classes: data imbalance while SSL – CReST[5] (CVPR 2021)
  - Motivation
    - Performance of the **majority class** is better? “**Partly True**”

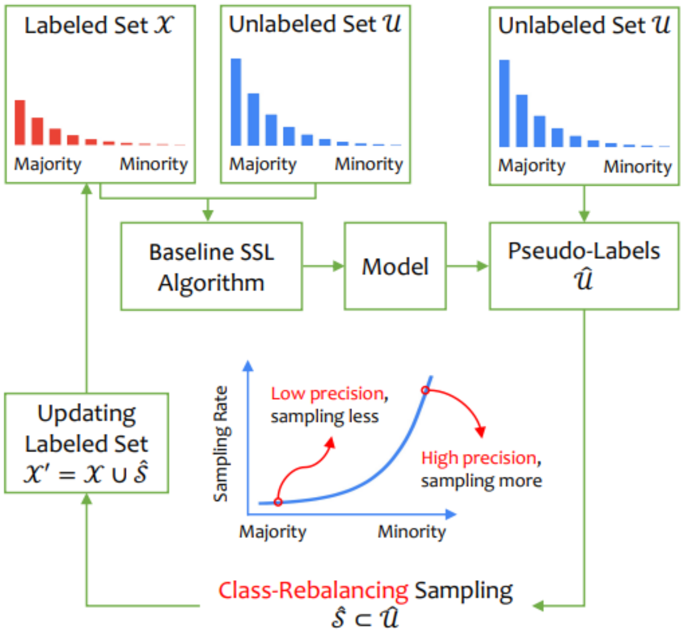


Recall & Precision for SSL model,
class index is sorted by the number of examples in descending order

*“The model is **conservative** in classifying samples into minority class,*
***but** once it makes such a prediction we can be **confident it is correct**.”*

How to deal with?

- "Small" in some classes: data imbalance while SSL – CReST[5] (CVPR 2021)
  - Contribution
    - Original training method
      1. Train on the labeled set
      2. The model's predictions are used to generate pseudo-label set
    - Modified training method
      1. Train on the labeled ***& unlabeled*** set
      2. The model's predictions are used to generate pseudo-label set ***stochastically(μ_l)***



$$\mu_l = \left(\frac{N_{L+1-l}}{N_1}\right)^{\alpha},$$

where α is constant

How to deal with?

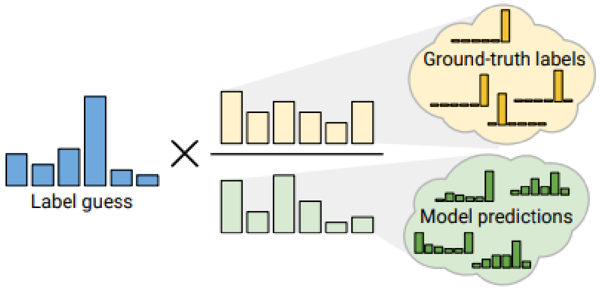
- "Small" in some classes: data imbalance while SSL – CReST[5] (CVPR 2021)
  - Contribution
    - Background
      - $y \in \{1, 2, \ldots, L\}$: represents class index
      - u : unlabeled data sample
      - $p(y)$: labeled set's class distribution → **target distribution**
      - $\tilde{p}(y)$: **moving average** of the model's prediction on **unlabeled examples**
      - $q = p(y|u; f)$: probability that the **unlabeled sample *u* belongs to *y***
    - Distribution Alignment(DA)

1. $q *= \frac{p(y)}{\tilde{p}(y)}$

   ✓ Induce $\tilde{p}(y)$ to have similar distribution with *p(y)*

2. $\tilde{q} = Normalize\left(q * \frac{p(y)}{\tilde{p}(y)}\right), Normalize(x)_i = {x_i}/{\sum_j x_j}$

   ✓ Form a valid probability distribution



Distribution alignment

How to deal with?

- “Small” in some classes: data imbalan
  - Contribution
    - DA with temperature scaling
      - Use $Normalize(p(y)^t)$ instead of $p(y)$, $t \in (0,1)$
      - Strategy to change the value of t
        - ✓**Low *t*** makes the distribution smoother and **balanced**
        - ✓If t is **too low**, however, distribution is **overly smoothed: wrong pseudo-labeling**
        - ✓**Decrease t over generations:** Both **high precision of the minority class** in early generations, and stronger **class-rebalancing** in late generations

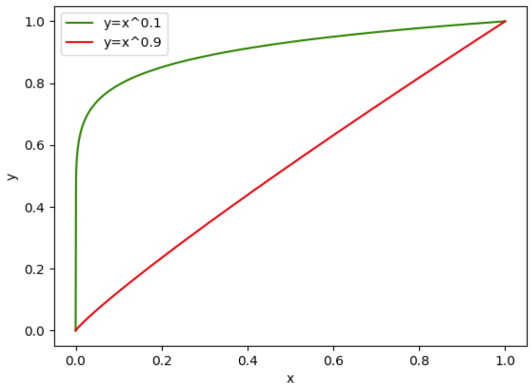
– Distribution Alignment(DA)

1. $q *= \frac{p(y)}{\tilde{p}(y)}$

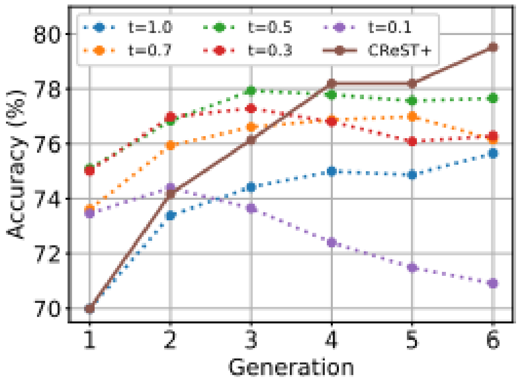
   ✓ Induce $\tilde{p}(y)$ to have similar distribution with $p(y)$

2. $\tilde{q} = Normalize\left(q * \frac{p(y)}{\tilde{p}(y)}\right), Normalize(x)_i = {x_i}/{\sum_j x_j}$

   ✓ to form a valid probability distribution



Graphs of $y = x^{\alpha}$



Accuracy over different t

How to deal with?

- “Small” in some classes: data imbalance while SSL – CReST[5] (CVPR 2021)
  - Results
    - The effectiveness of the two contribution

β: Ratio of labeled data

γ: Imbalance ratio $^{number\ of\ the\ most\ majority\ class}/_{number\ of\ the\ most\ minority\ class}$

Method	CIFAR10-LT						CIFAR100-LT			
	$\beta = 10\%$			$\beta = 30\%$			$\beta = 10\%$		$\beta = 30\%$	
	$\gamma = 50$	$\gamma = 100$	$\gamma = 200$	$\gamma = 50$	$\gamma = 100$	$\gamma = 200$	$\gamma = 50$	$\gamma = 100$	$\gamma = 50$	$\gamma = 100$
FixMatch [39]	79.4$_{\pm 0.65}$	66.3$_{\pm 1.74}$	59.7$_{\pm 0.74}$	81.9$_{\pm 0.30}$	73.1$_{\pm 0.58}$	64.7$_{\pm 0.69}$	33.7$_{\pm 0.94}$	28.3$_{\pm 0.66}$	43.1$_{\pm 0.24}$	38.6$_{\pm 0.45}$
w/ CReST	83.8$_{\pm 0.45}$	75.9$_{\pm 0.62}$	64.1$_{\pm 0.23}$	84.2$_{\pm 0.13}$	77.6$_{\pm 0.86}$	67.7$_{\pm 0.82}$	37.4$_{\pm 0.29}$	32.1$_{\pm 1.52}$	45.6$_{\pm 0.19}$	40.2$_{\pm 0.53}$
w/ CReST+	**84.2**$_{\pm 0.39}$	**78.1**$_{\pm 0.84}$	**67.7**$_{\pm 1.39}$	**84.9**$_{\pm 0.27}$	**79.2**$_{\pm 0.20}$	**70.5**$_{\pm 0.56}$	**38.8**$_{\pm 1.03}$	**34.6**$_{\pm 0.74}$	**46.7**$_{\pm 0.34}$	**42.0**$_{\pm 0.44}$

    - Per class performance

Method / Class	Split	1	2	3	4	5	6	7	8	9	10	Avg.
FixMatch [39]	test	**98.7**	**99.5**	**90.0**	**83.5**	85.0	47.6	69.9	59.0	8.9	7.2	64.9
w/ CReST	test	97.7	98.3	88.8	81.9	**88.2**	59.7	79.5	61.2	47.0	47.9	75.0
		-1.0	-1.2	-1.2	-1.6	+3.2	+12.1	+9.6	+2.2	+38.1	+40.7	+10.1
w/ CReST+	test	93.8	97.7	87.3	76.9	87.5	**69.2**	**84.9**	**67.9**	**60.3**	**70.8**	**79.6**
		-4.9	-1.8	-2.7	-6.6	+2.5	+21.6	+15.0	+8.9	+51.4	+63.6	+14.7
FixMatch [39]	unlabeled	**98.5**	**99.1**	**90.0**	**84.0**	84.7	49.7	64.9	65.6	14.9	22.2	67.4
w/ CReST	unlabeled	97.8	96.8	**90.0**	82.9	87.4	62.4	79.3	64.8	60.8	66.7	78.9
		-0.7	-2.3	0	-1.1	+2.7	+12.7	+14.4	-0.8	+45.9	+44.5	+11.5
w/ CReST+	unlabeled	92.2	95.7	86.1	76.7	**87.6**	**68.1**	**85.1**	**71.2**	**75.7**	**75.6**	**81.4**
		-6.3	-3.4	-3.9	-7.3	+2.9	+18.4	+20.2	+5.6	+60.8	+53.4	+14.0

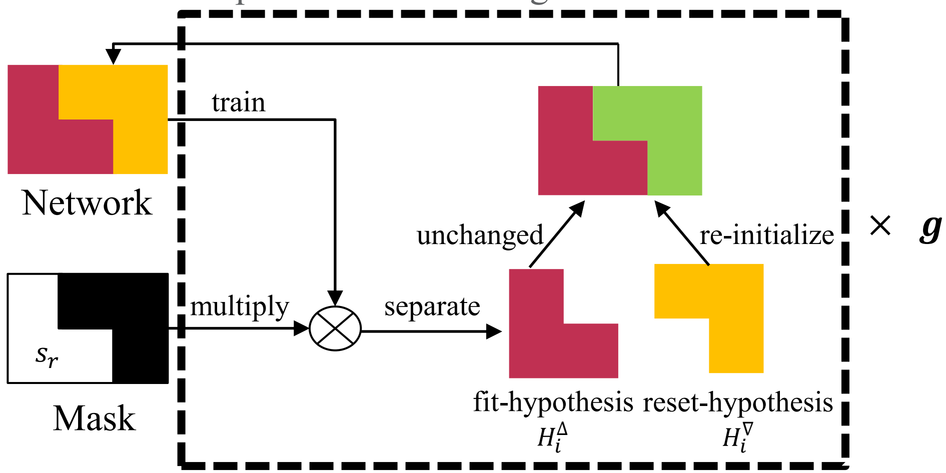
How to deal with?

- Just “Small” – Knowledge Evolution(CVPR 2021, Oral)[6]
  - Motivation
    - Training on a small dataset is challenging. **WHY?**
      - Some parameters are **redundant** and enable **overfitting on a small dataset**
    - Need to do **zero-mapping**(ex: weight decay)
      - Reduce the complexity of the network

“Knowledge Evolution”

How to deal with?

- Just "Small" – Knowledge Evolution(CVPR 2021, Oral)[6]
  - Method
    1. Make a binary mask with sparsity ratio s_r
    2. Initialize the network N with random parameters
    3. Train the network N_i (i th generation)
    4. Separate the network $N_i \rightarrow H_i^{\Delta}, H_i^{\nabla}$
    5. Remain H_i^{Δ} unchanged; re-initialize H_i^{∇}; $i \leftarrow i+1$
    6. Repeat 3-5 while i <= g



How to make a mask?

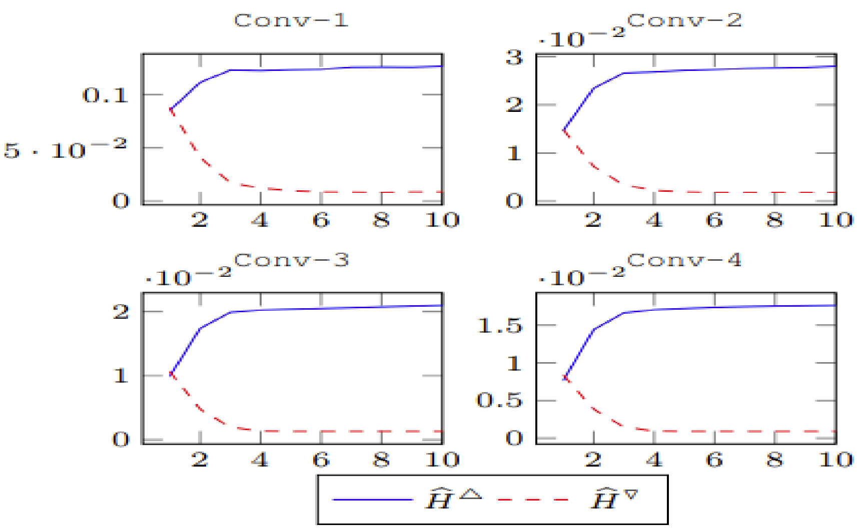
1. **WE**ight-**L**evel **S**plitting (WELS)
   1. Similar as weight pruning
   2. **Advantage** : applicable to any computations(CNN, FC, etc.)
   3. **Disadvantage** : can't split the fit-hypothesis(H^{Δ}) when inferencing
2. **KE**rnel-**L**evel **S**plitting(KELS)
   1. $C_o \times k \times k \times C_i \rightarrow [s_r \times C_o] \times k \times k \times [s_r \times C_i]$
   2. **Advantage** : can split the fit-hypothesis(H^{Δ}) when inferencing
   3. **Disadvantage** : applicable to only CNN

How to deal with?

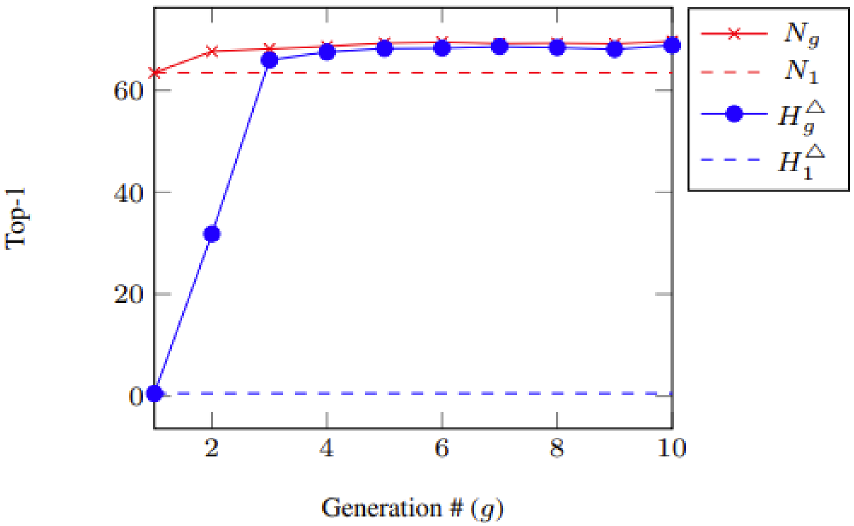
- Just “Small” – Knowledge Evolution(CVPR 2021, Oral)[6]
  - Method
    - Zero-mapping?

	$\mathbb{C}$	Trn	Val	Tst	Total
Flower-102 [36]	102	1020	1020	6149	8189
CUB-200 [52]	200	5994	N/A	5794	11788
Aircraft [33]	100	3334	3333	3333	10000
MIT67 [41]	67	5360	N/A	1340	6700
Stanford-Dogs [24]	120	12000	N/A	8580	20580

Table. Small amounts of data



Absolute values inside H^{Δ} and H_i^{∇}



Evaluation on CUB_200[7]

How to deal with?

- Just "Small" – Knowledge Evolution(CVPR 2021, Oral)[6]
  - Results
    - Classification

Method	Flower	CUB	Aircraft	MIT	Dog
CE (N_1)	45.76	55.49	51.96	57.37	65.09
CE + KE-N_3 **(ours)**	50.50	57.73	56.34	60.64	66.08
CE + KE-N_{10} **(ours)**	**58.78**	**58.96**	**61.70**	**61.76**	**67.30**
Smth (N_1)	45.85	59.01	58.45	57.07	66.31
Smth + KE-N_3 **(ours)**	53.69	**62.38**	63.18	**59.52**	68.00
Smth + KE-N_{10} **(ours)**	**65.88**	60.57	**65.60**	59.15	**68.66**
CS-KD (N_1)	49.32	66.71	57.62	56.77	68.82
CS-KD + KE-N_3 **(ours)**	59.67	**69.63**	59.43	57.14	70.66
CS-KD + KE-N_{10} **(ours)**	**66.34**	69.35	**59.76**	**57.37**	**70.59**

Based on KELS, $s_r = 0.8$

Method	Flower	CUB	Aircraft	MIT	Dog
CE (N_1)	44.88	56.32	51.61	55.13	66.15
CE + KE-N_3 **(ours)**	50.23	**59.81**	56.25	**60.27**	66.44
CE + KE-N_{10} **(ours)**	**58.03**	59.38	**60.80**	59.45	**67.25**
Smth (N_1)	45.92	58.70	56.73	58.26	66.48
Smth + KE-N_3 **(ours)**	54.84	**62.41**	62.68	60.49	67.98
Smth + KE-N_{10} **(ours)**	**64.69**	60.36	**65.62**	**62.13**	**68.26**
CS-KD (N_1)	46.75	66.66	58.87	56.85	69.22
CS-KD + KE-N_3 **(ours)**	58.27	69.67	60.98	**57.51**	70.94
CS-KD + KE-N_{10} **(ours)**	**64.18**	**71.37**	**61.37**	57.22	**71.33**

Based on WELS, $s_r = 0.7$

	C	Trn	Val	Tst	Total
Flower-102 [36]	102	1020	1020	6149	8189
CUB-200 [52]	200	5994	N/A	5794	11788
Aircraft [33]	100	3334	3333	3333	10000
MIT67 [41]	67	5360	N/A	1340	6700
Stanford-Dogs [24]	120	12000	N/A	8580	20580

Table. Small amounts of data

CUB on VGG11_bn

	s_r	Acc_1	Acc_{10}	$\blacktriangle_{\text{acc}}$	#Ops	$\blacktriangle_{\text{ops}}$	#Param
N_g	0.5	63.47	69.65	6.1%	15.22	-	259.16
$H_g^{\triangle}$		0.52	68.84	5.3%	3.85	74.7%	65.20

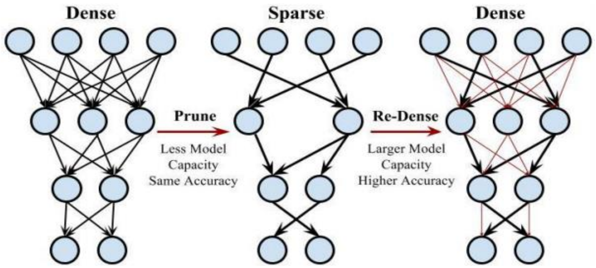
FLW on ResNet18

	s_r	Acc_1	Acc_{100}	$\blacktriangle_{\text{acc}}$	#Ops	$\blacktriangle_{\text{ops}}$	#Param
N_g	0.8	53.87	75.62	21.7%	3.63	-	22.44
$H_g^{\triangle}$		6.41	75.62	21.7%	2.39	34.1%	14.43
N_g	0.5	52.62	74.60	21.9%	3.63	-	22.44
$H_g^{\triangle}$		0.37	74.60	21.9%	0.96	73.5%	5.64

Based on KELS, $s_r = 0.8$

How to deal with?

- Just "Small" – Knowledge Evolution(CVPR 2021, Oral)
  - Connection
    - DSD[8]?
      - Special case of 'Knowledge evolution'
        - ✓Re-initialize randomly instead of using 0
          - **Bad for kernels**
        - ✓DSD is done for only one generation



Training approach of DSD[8]

Method	Flower	CUB	Aircraft	MIT	Dog
CE + AdaCos	49.96	**62.20**	56.15	50.89	65.33
CE + RePr	39.75	47.01	36.04	49.77	55.63
CE + DSD	48.85	56.11	53.66	58.31	65.76
CE + BANs-N_{10}	44.92	57.30	52.56	57.66	65.49
CE (N_1)	45.85	55.16	51.73	56.62	64.82
CE + KE-N_3 **(ours)**	52.44	57.75	56.70	**59.67**	67.06
CE + KE-N_{10} **(ours)**	**60.15**	58.01	**59.73**	58.71	**67.75**

Table. Based on WELS.

Summary

- About small amounts of data
  - Lack of diversity
    - Differentiable augmentor
  - Data imbalance while semi-supervised learning
    - Data re-balancing
  - Overfitting when training on a small dataset
    - Zero-mapping
    - Iterative learning

Reference

[1]Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and C Lawrence Zitnick, "Microsoft COCO: Common objects in context," in ECCV, 2014.

[2]Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson, "nocaps: novel object captioning at scale," in Proceedings of the IEEE International Conference on Computer Vision, pp. 8948–8957, 2019

[3]Gong, Kehong, Jianfeng Zhang, and Jiashi Feng. "PoseAug: A Differentiable Pose Augmentation Framework for 3D Human Pose Estimation." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.

[4] Shichao Li, Lei Ke, Kevin Pratama, Yu-Wing Tai, Chi-Keung Tang, and Kwang-Ting Cheng. Cascaded deep monocular 3d human pose estimation with evolutionary training data. In

CVPR, 2020.

[5]Wei, Chen, et al. "Crest: A class-rebalancing self-training framework for imbalanced semi-supervised learning." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.

[6]https://arxiv.org/abs/2103.05152

[7]Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011

[8] Song Han, Jeff Pool, Sharan Narang, Huizi Mao, Enhao Gong, Shijian Tang, Erich Elsen, Peter Vajda, Manohar Paluri, John Tran, et al. Dsd: Dense-sparse-dense training for deep neural networks. arXiv preprint arXiv:1607.04381, 2016.