압축 심층신경망의 설명가능성 손상과 그 방지

Attribution Preservation in Network Compression for Reliable Network Interpretation(NeurIPS `20)

Machine Learning and Intelligence Lab @ KAIST

양은호 교수님 연구실

발표자: 양준용 (laoconeth@kaist.ac.kr)

KAIST

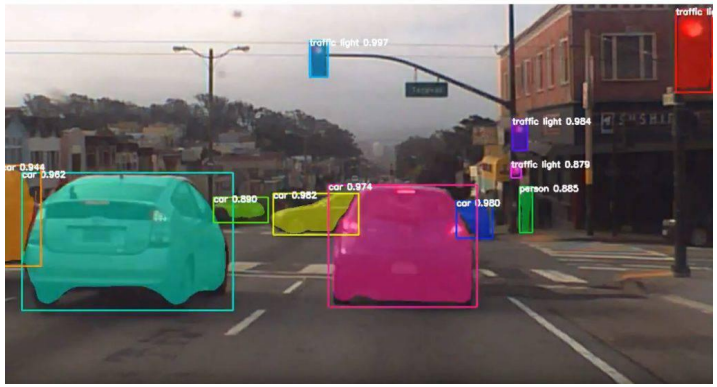
MLILAB Machine Learning & Intelligence

*Equal contribution.

Safety-Critical Deep Learning

Safety-critical applications:

- Decisions and predictions may cause **massive consequences**
- Autonomous driving, autopilot, healthcare, etc.
- Need **reliability**(Constancy) and **trust**.



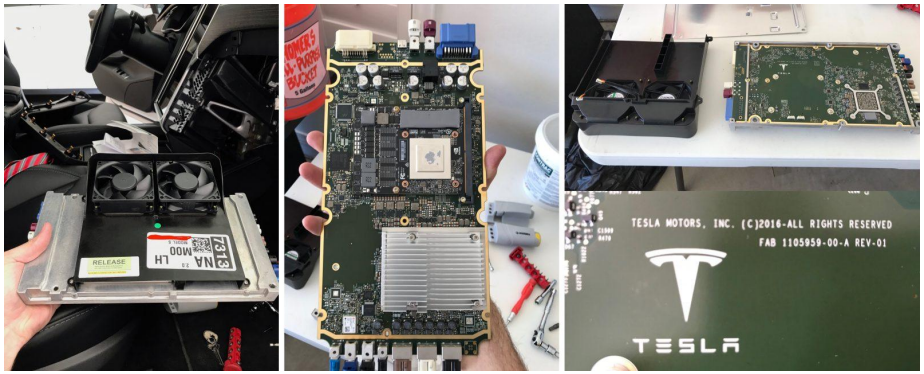
Autonomous Driving

Wearable Health Monitors

Safety-Critical Deep Learning

Safety-critical applications require:

- **Reliability(Constancy): Must provide constant uptime.**
  - Cloud based models are unreliable due to communication failures(+ latency, privacy issues).
  - Need to embed models directly on edge devices.
  - **Network compression** is needed to fit them inside smaller spaces
- Trust: Provide explanations to the decisions of the model to trust the model.
  - 'Why' did this self-driving car ran a red light?
  - **Explainable AI(XAI)** algorithms are needed



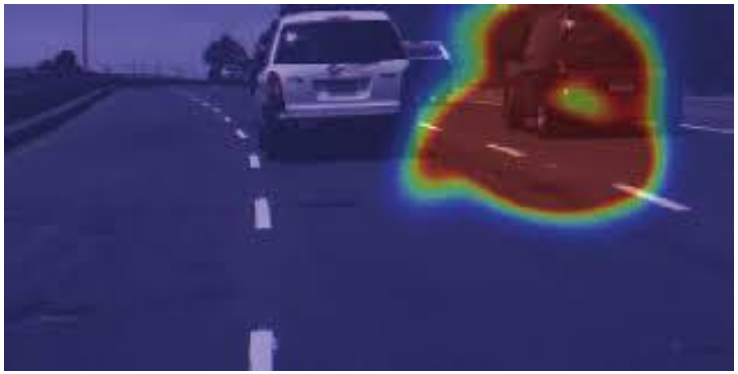
Autonomous Driving

Wearable Health Monitors

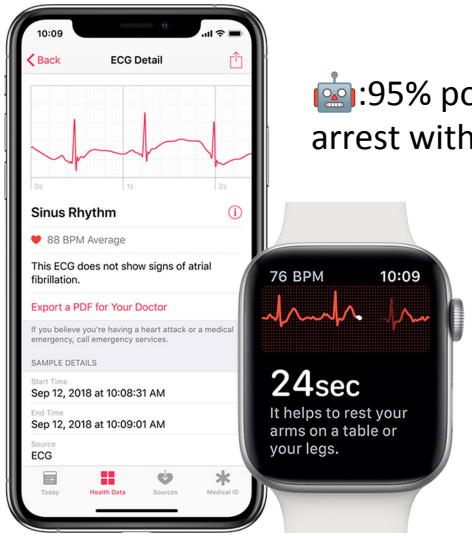
Safety-Critical Deep Learning

Safety-critical applications require:

- Reliability(Constancy): Must provide constant uptime.
  - Cloud based models are unreliable due to communication failures(+ privacy issues).
  - Need to embed models directly on edge devices.
  - **Network compression** is needed to fit them inside smaller spaces
- **Trust: Provide explanations to the decisions of the model to trust the model.**
  - 'Why' did this self-driving car ran a red light?
  - **Explainable AI(XAI)** algorithms are needed



Autonomous Driving
(image courtesy Kim et al. ICCV 2017)



Wearable Health Monitors

🤖:95% possibility of cardiac arrest within in 3 hrs

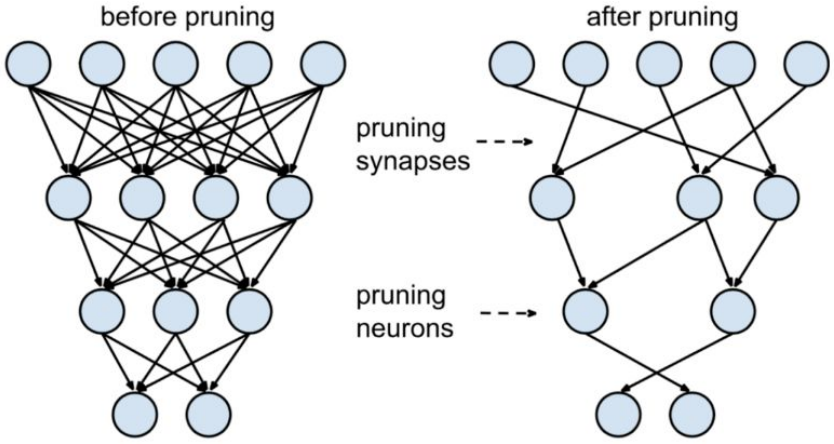
But why?!? : 👨

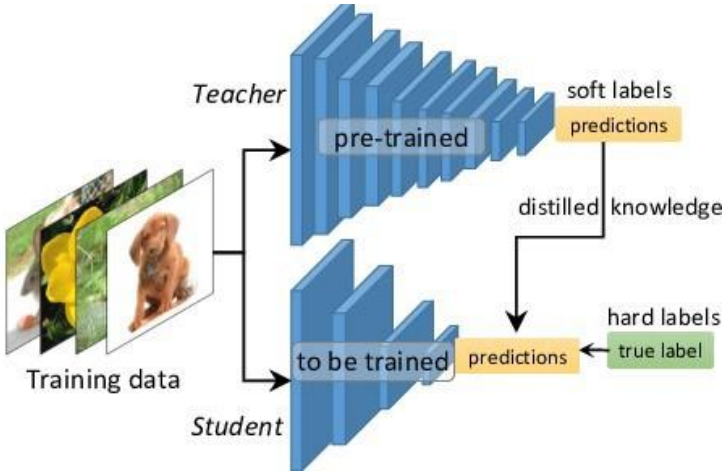
MLILAB Machine Learning & Intelligence

Network Compression

Network Compression

- Modern neural networks require massive computing power
- Make the network consume **less computational(time, space) cost** while **maintaining its prediction power**
- Why: Embedding DNNs directly in edge devices(**reliability**)
- Knowledge distillation, Network pruning, Sparsification, Weight Quantization, etc.

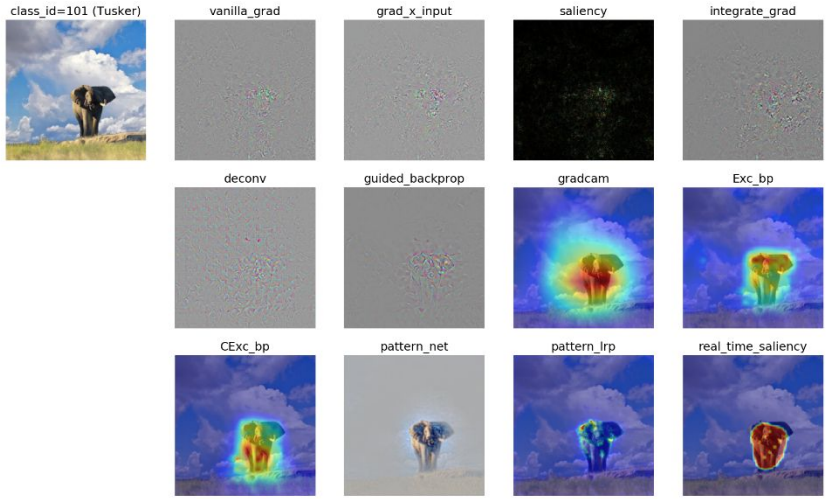


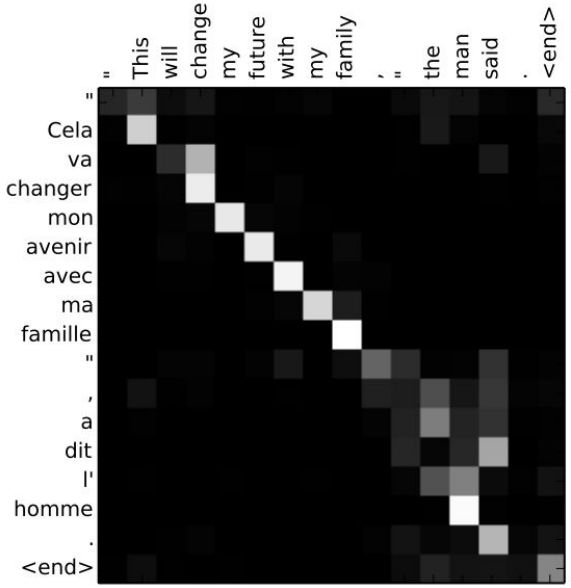


Explainable AI

Explainable AI(XAI)

- DNNs are practically black boxes
- Produce **human-understandable interpretations** of decisions and predictions of neural networks
- Why: **Trustworthiness**, transparency
- **Input attribution**, interpretable models, etc.





1) Yulong Wang., Pytorch Visual Attribution 2018, github.com/yulongwang12/visual-attribution

Attribution Deformation Problem

Compression Breaks Attribution

- *Compressed networks produce **deformed attribution** maps compared to their **former selves** and the **ground truth segmentations**.*
- Happens across **various** compression methods and attribution methods

Attribution Deformation Problem

Compression Breaks Attribution

- The attributions of the compressed network are not only different from their past counterparts but also broken down compared to their respective segmentation ground truths.
- Happens across **various** compression methods and attribution methods
- Space restriction forces the network to abandon its standard decision procedures and resort to using **human-indecipherable shortcuts and hints**, which emerges in its deformed attribution maps.

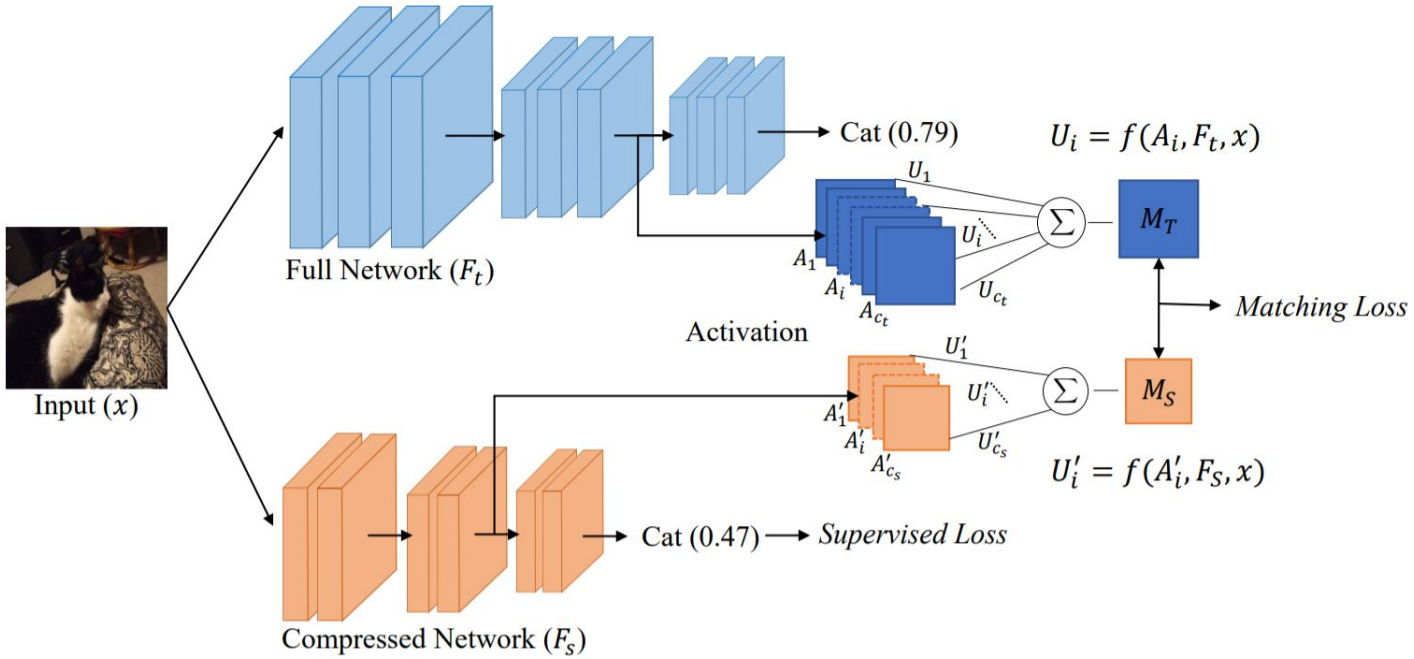
	For samples with correct pred.		
Method	#Param	AUC	Point Acc
Full (Teacher)	15.22M	88.79	80.21
Knowledge Distillation	0.29M	78.74	67.26
Structured Pruning	3.27M	79.98	75.29
Unstructured Pruning	0.53M	84.13	75.43
KD (w/ Ours)	0.29M	88.06	79.12

Table 1: Evaluation of how many samples were broken compared to the ground truth (segmentation labels) by various compression methods.

Attribution Matching to Preserve Attribution

- While compressing, make the attribution map of the compressing network follow the map of the pre-compression network
- Employ **a matching loss** to keep the maps of the compressing network close to the pre-compression network

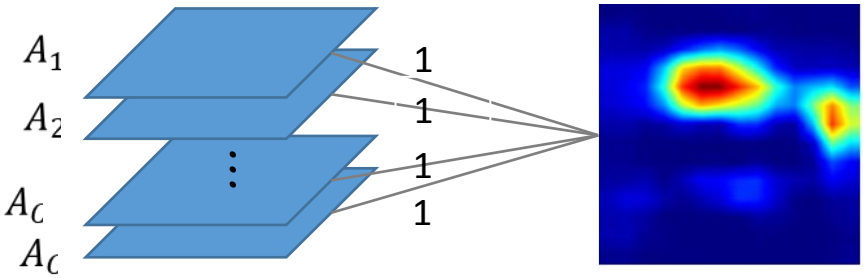
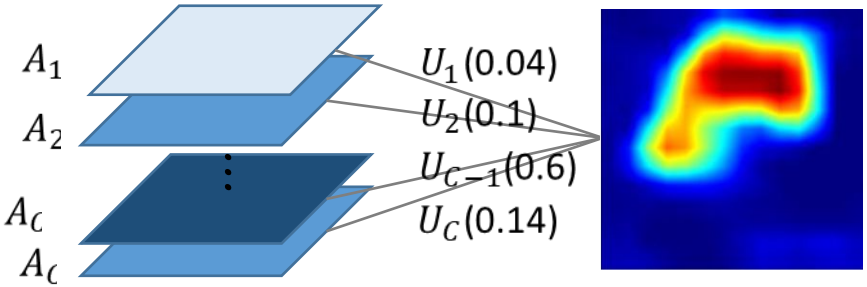
$$L_{total} = L(W_s, x) + \beta \sum_{j \in I} \left\| \frac{M_s^{(j)}}{\left\| M_s^{(j)} \right\|_2} - \frac{M_t^{(j)}}{\left\| M_t^{(j)} \right\|_2} \right\|_2$$



Weight Collapsed Attribution Matching

- Generate attribution maps by collapsing them in the channel dimension

$$M^{(l)} = V\left(\sum_{c=1}^{C} U_c^{(l)} \cdot T\left(A_c^{(l)}\right)\right)$$

Equally weighted	Sensitivity weighted
$M_{act} = \sum_{c=1}^{C} A_c^2$	$M_{cg} = ReLU\left(\sum_{c=1}^{C} U_c \cdot A_c\right)$ $U_c = \sum_{a_h, a_w} \frac{\partial y^t}{\partial A_{(a_h, a_w, c)}}$
	

Stochastic Matching

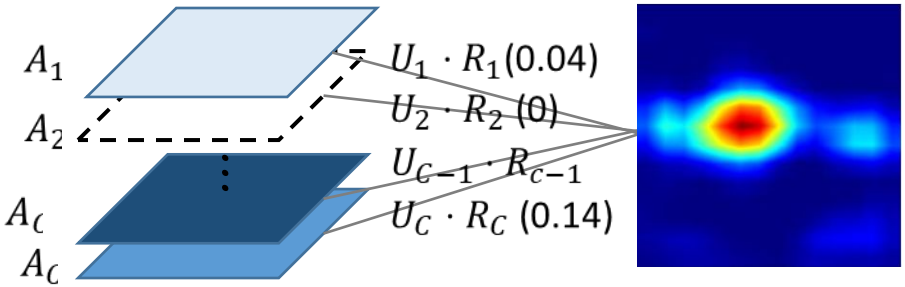
- We generate attribution maps by collapsing them in the channel dimension

$$M^{(l)} = V\left(\sum_{c=1}^{C} U_c^{(l)} \cdot T\left(A_c^{(l)}\right)\right)$$

- When collapsing the channels, stochastically drop certain channels
- Increases generalization performance(similar to channel-wise dropout)

Stochastic sensitivity weighted

$$R_c \sim Bern(p)$$
$$U_c = R_c \cdot u_c$$
$$u_c = \phi(A_c, F_t, x)$$



Knowledge Distillation

Basic knowledge distillation

- Naïve Knowledge Distillation causes attribution map distortion
- Our framework effectively preserves the attribution map similarly to the pre-compression network, which in turn **preserves attribution against ground truth**.
- Attribution preservation also helps in preserving the **predictive performance**.

		Prediction Performance		Attribution Score	
Network	Method	mAP	F1 Score	AUC	Point Acc
VGG16	Teacher	91.83	78.44	88.79	80.21
VGG16 / 2	KD	83.75	65.92	82.53	72.01
	EWA	86.48	**68.19**	85.05	80.42
	SWA	**86.56**	67.78	88.12	80.66
	SSWA	86.42	67.94	**88.89**	**81.13**
VGG16 / 4	KD	81.31	62.50	80.61	68.86
	EWA	82.46	63.57	84.18	79.34
	SWA	83.67	65.14	87.90	80.05
	SSWA	**84.47**	**66.13**	**88.10**	**80.26**
VGG16 / 8	KD	76.91	52.51	78.74	67.26
	EWA	79.56	58.91	81.99	78.49
	SWA	80.14	60.70	87.88	**79.59**
	SSWA	**80.86**	**61.43**	**88.06**	79.12

		Similarity	
Network	Method	Cos	ℓ_2
VGG16	Teacher	-	-
VGG16 / 2	KD	0.705	29.84
	EWA	0.788	21.21
	SWA	**0.873**	**12.98**
	SSWA	0.859	14.36
VGG16 / 4	KD	0.650	35.52
	EWA	0.750	25.24
	SWA	**0.841**	**16.23**
	SSWA	0.837	16.63
VGG16 / 8	KD	0.563	44.10
	EWA	0.652	34.90
	SWA	0.813	**19.04**
	SSWA	**0.842**	19.49

Unstructured Pruning

Weight magnitude based unstructured pruning

- Since unstructured pruning is relatively tolerable, attribution distortion occurs less than other compression methods.
- Similar phenomena were observed: attribution score and predictive performance is preserved.

Table 4: Unstructured pruning models evaluated against ground truth (segmentation). Among the results of iterative pruning, the last remaining small-est network was evaluated.

	Prediction Performance		Attribution Score	
Method	mAP	F1 Score	AUC	Point Acc
Full(Teacher)	91.83	78.44	88.79	80.21
Naive	87.42	70.24	84.13	75.43
EWA	89.75	74.83	86.67	79.37
SWA	89.79	75.11	88.22	**79.86**
SSWA	**89.96**	**75.51**	**88.45**	79.25

Table 5: Unstructured pruning results for attribution map deformation from teacher to student network.

Method	Cos	ℓ_2
Full(Teacher)	-	-
Naive	0.790	21.21
EWA	0.895	10.71
SWA	0.913	8.407
SSWA	**0.920**	**7.826**

Structured Pruning

L1-magnitude based structured pruning

- Similar phenomena were seen in the structured pruning method. As the structured pruning prune with the unit of channels, attribution distortion occurs more than unstructured pruning.
- Our framework also help preserving the attribution maps with structured pruning.

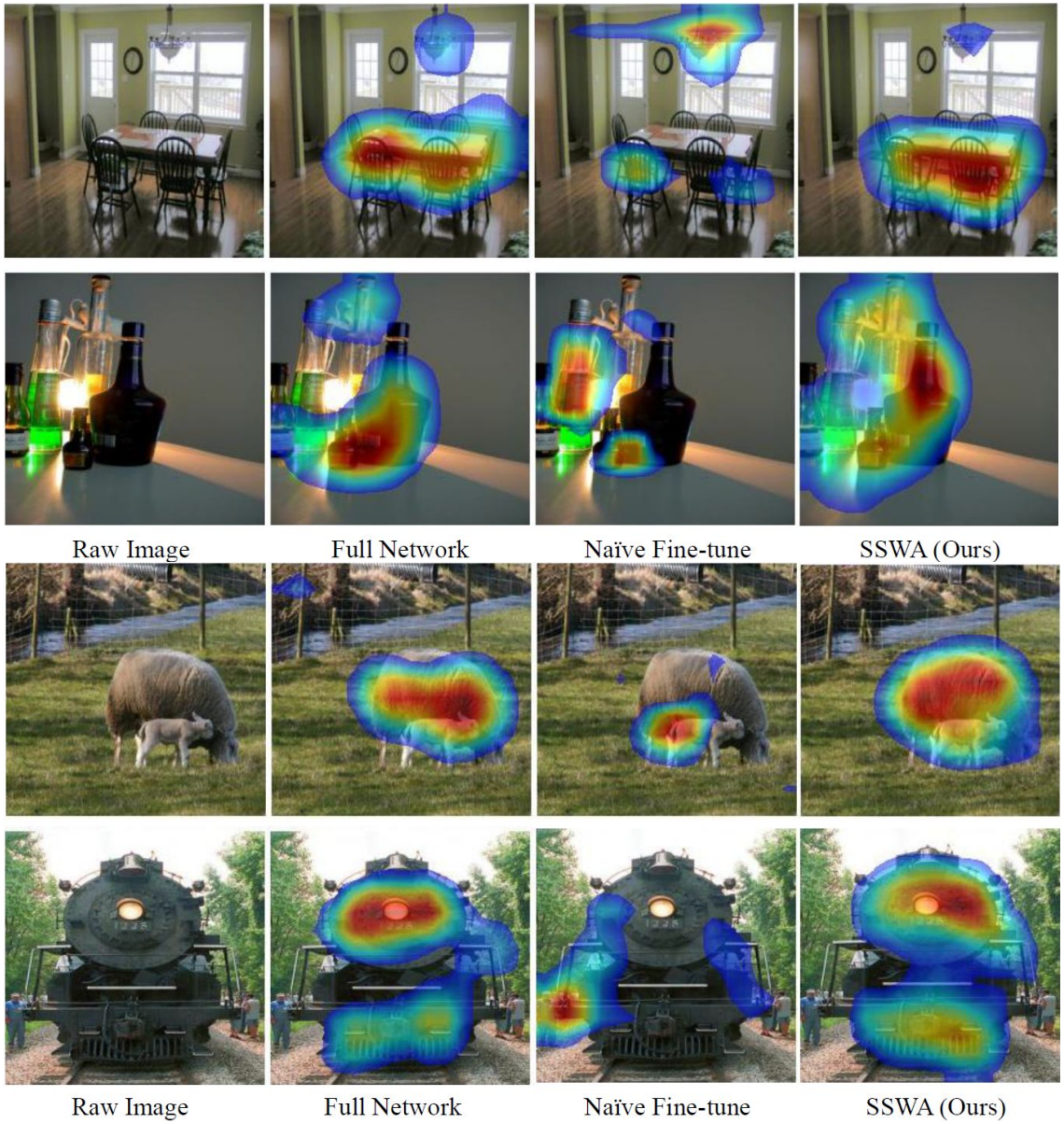
Table 6: ℓ_1-structured pruning models evaluated against ground truth (segmentation).

	Prediction Performance		Attribution Score	
Method	mAP	F1 Score	AUC	Point Acc
Full(Teacher)	91.83	78.44	88.79	80.21
Naive	83.76	60.71	79.98	75.29
EWA	87.62	66.05	83.96	78.84
SWA	88.39	67.70	86.99	**81.65**
SSWA	**89.07**	**68.28**	**88.34**	81.08

Table 7: ℓ_1-structured pruning results for attribution map deformation from teacher to student network.

Method	Cos	ℓ_2
Full(Teacher)	-	-
Naive	0.764	30.04
EWA	0.855	14.74
SWA	**0.911**	**9.102**
SSWA	0.910	9.232

Sample Images(Structured Pruning)



Other Attribution maps

Effects on Other Attribution Methods

- We observe that the maps of the three attribution methods are indeed deformed when compression is performed, and exhibit inferior point accuracy and ROC-AUC performance compared to the network before compression.
- Even though our framework (SSWA) utilized gradient based attribution maps akin to Grad-Cam, **employing this regularizer helps to preserve other attribution methods.**

Table 8: Attribution deformation and preservation results on other attribution methods. For this experiment, we use the knowledge distillation with VGG/8. We report the AUC and Point accuracy to evaluate the localization ability of the attribution maps.

AUC/Point Acc	Grad Cam	Excitation Bp	$\text{LRP}_{\alpha=1,\beta=0}$	RAP
Full (Teacher)	88.79/80.21	84.14/74.80	85.29/65.48	84.54/69.49
Naive	78.74/67.26	76.31/66.31	79.60/53.43	80.85/56.87
SSWA (Ours)	**88.06/79.12**	**82.31/71.24**	**82.46/64.08**	**83.53/65.66**

Conclusion

- Discovered that naïve network compression causes the **Attribution Deformation Problem,** which has not been considered before.

- Proposed and constructed **the Attribution Map Matching** framework, which enforces the attribution maps of the compressed network to follow the pre-compression network.

- Experiments show that our method indeed **preserves various kinds of attribution**. Also, our method yields gains in terms of **predictive performance**.

Future works

Preservation for black box models and attribution algorithms

- Recent works propose black box attribution via input occlusion/perturbation
- However, our work requires intermediate representations and gradients
- Preserve attributions without using intermediate representations

Getting rid of the teacher network while training

- Our method uses more computation in the training phase due to the teacher(full network)
- Preserve attributions without a teacher network

End of Presentation

Machine Learning and Intelligence Lab @ KAIST

Presenter: Juneyong Yang

laoconeth@kaist.ac.kr