Language Technologies Institute

Carnegie Mellon University

Multimodal Machine Learning

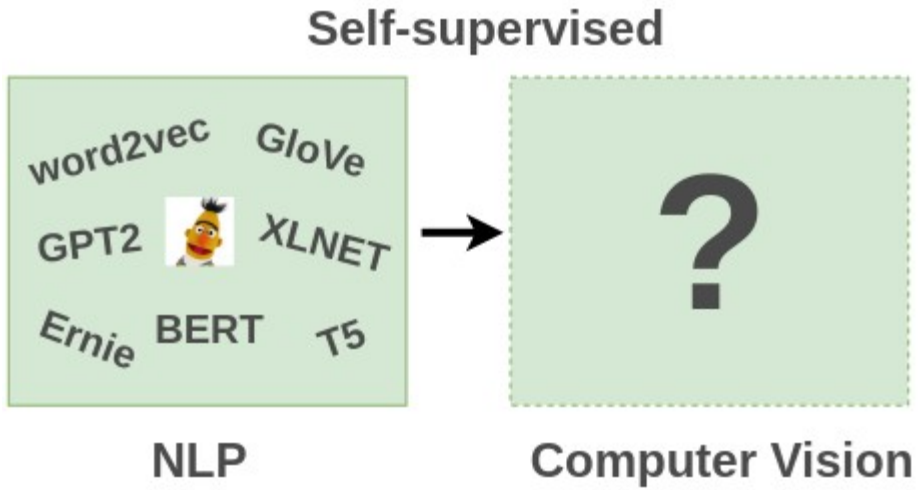
Lecture 12.1: Future Directions

Louis-Philippe Morency, Amir Zadeh and
11-777 Fall 2021 TA team

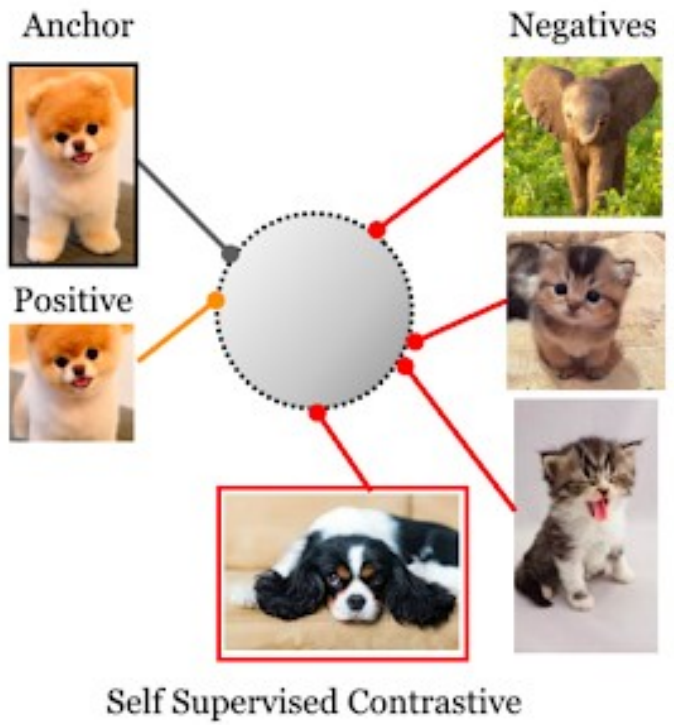
Self-supervised Contrastive Learning

Self-supervised learning

- A form of unsupervised learning where the data provides the supervision
- Predominant in NLP, but not so much in CV
- Until recently…

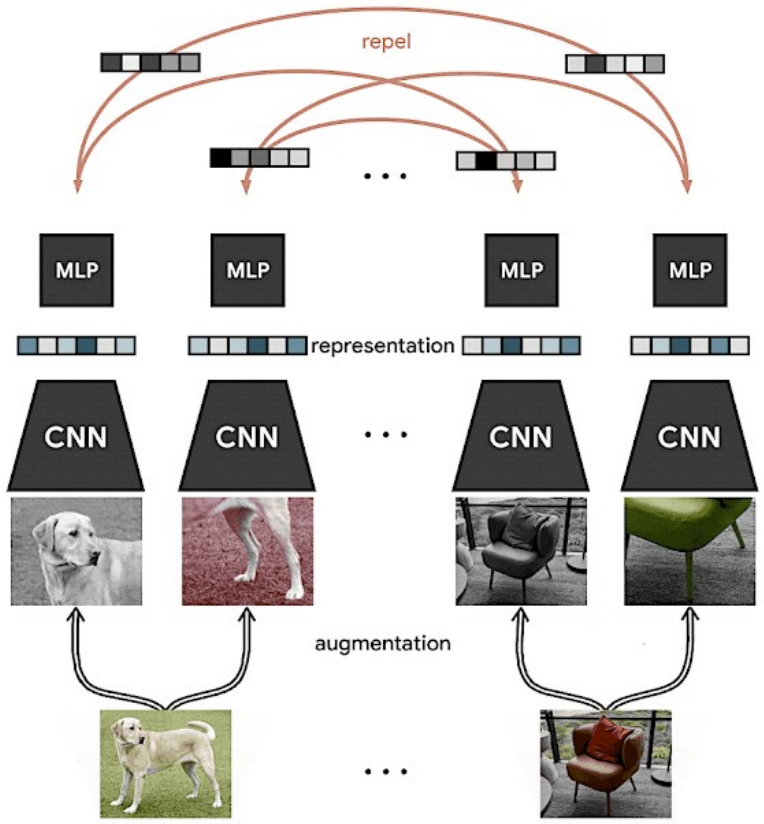


Contrastive Learning for Self-Supervised Learning



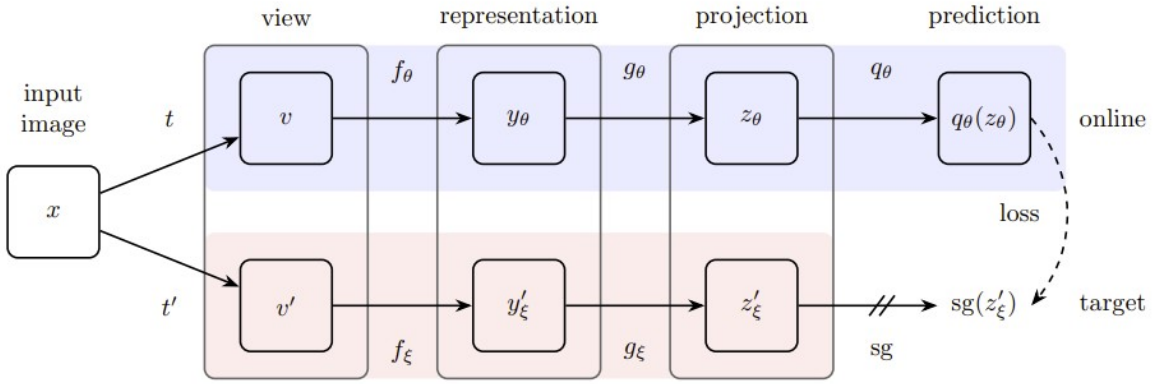
- Three elements: an anchor point, positive samples, negative samples
- Construct an embedding space, where the positive samples are close to the anchor point, and the negative samples are away from the anchor point
- Recently achieving very strong results

Contrastive Learning for Self-Supervised Learning



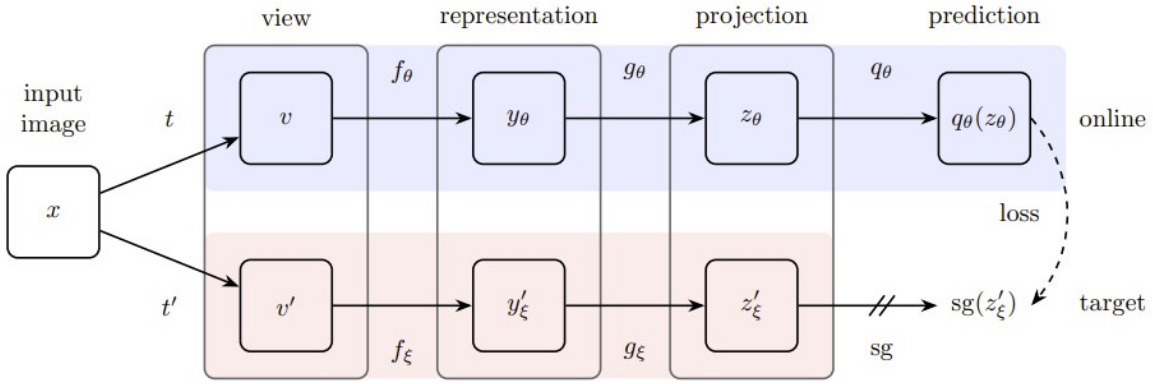
- Components (SimCLR [Chen et al. ICML 2020]):
- Stochastic Data Augmentation
- Encoder (CNN and MLP)
- A contrastive loss, InfoNCE
- Model learns to distinguish positive from negative pairs

BYOL (Bootstrap your own latent) [Grill et al. NeurIPS 2020]



- Stochastic data augmentation
- Encoder: two parallel networks: online and target
- Target network is more consistent than the online network (target network uses momentum update)
- An extra prediction network in the online network to create asymmetry and avoid collapsing
- MSE loss between the presentations from the online network and the target network

BYOL (Bootstrap your own latent) [Grill et al. NeurIPS 2020]



- Significance: good contrastive learning methods used to need a large batch size (4096 images in Google TPU) or a large dictionary (65536 images) to store negative samples

- Why it works: debatable research question (as collapsing is very easy):
  - Asymmetric structure so that the slowly updated target network is different from the online network

  - Other theories: batch normalization stop gradient, and more

Visual Counting

Visual Counting: Explicit Counting Module vs. Implicit Counting Module

- Explicit Counting Module: high interpretability, whereas limited scalability

Figure 1: IRLC takes as input a counting question and image. Detected objects are added to the returned count through a sequential decision process. The above example illustrates actual model behavior after training.

Alexander Trott, Caiming Xiong, and Richard Socher. Interpretable counting for visual question answering. In *ICLR*, 2018.

Visual Counting: Explicit Counting Module vs. Implicit Counting Module

- Implicit Counting Module: high scalability and efficiency, but reduced interpretability

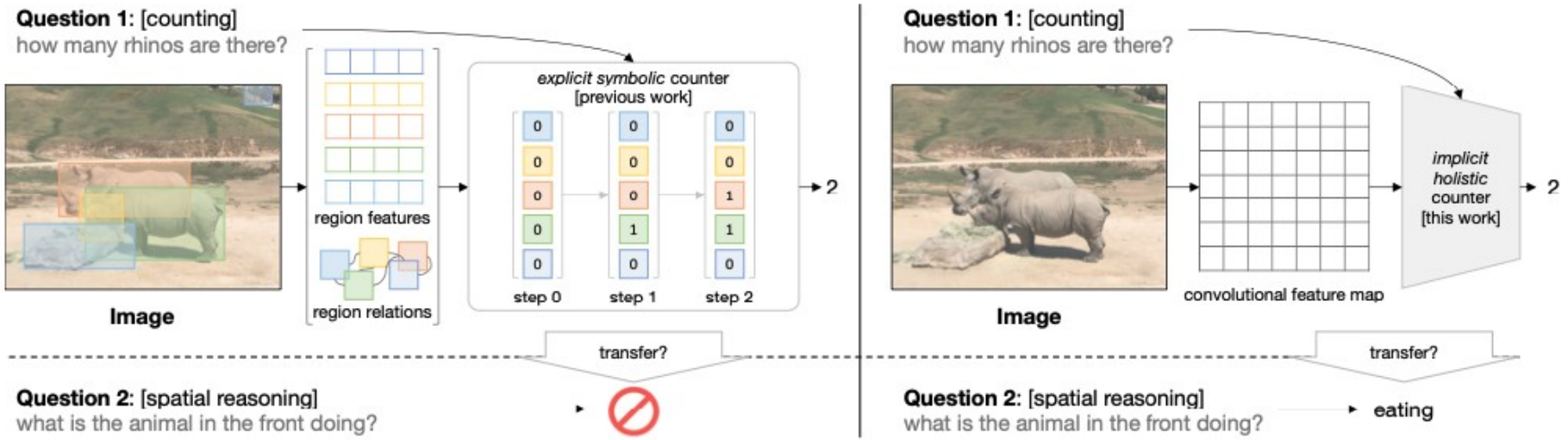
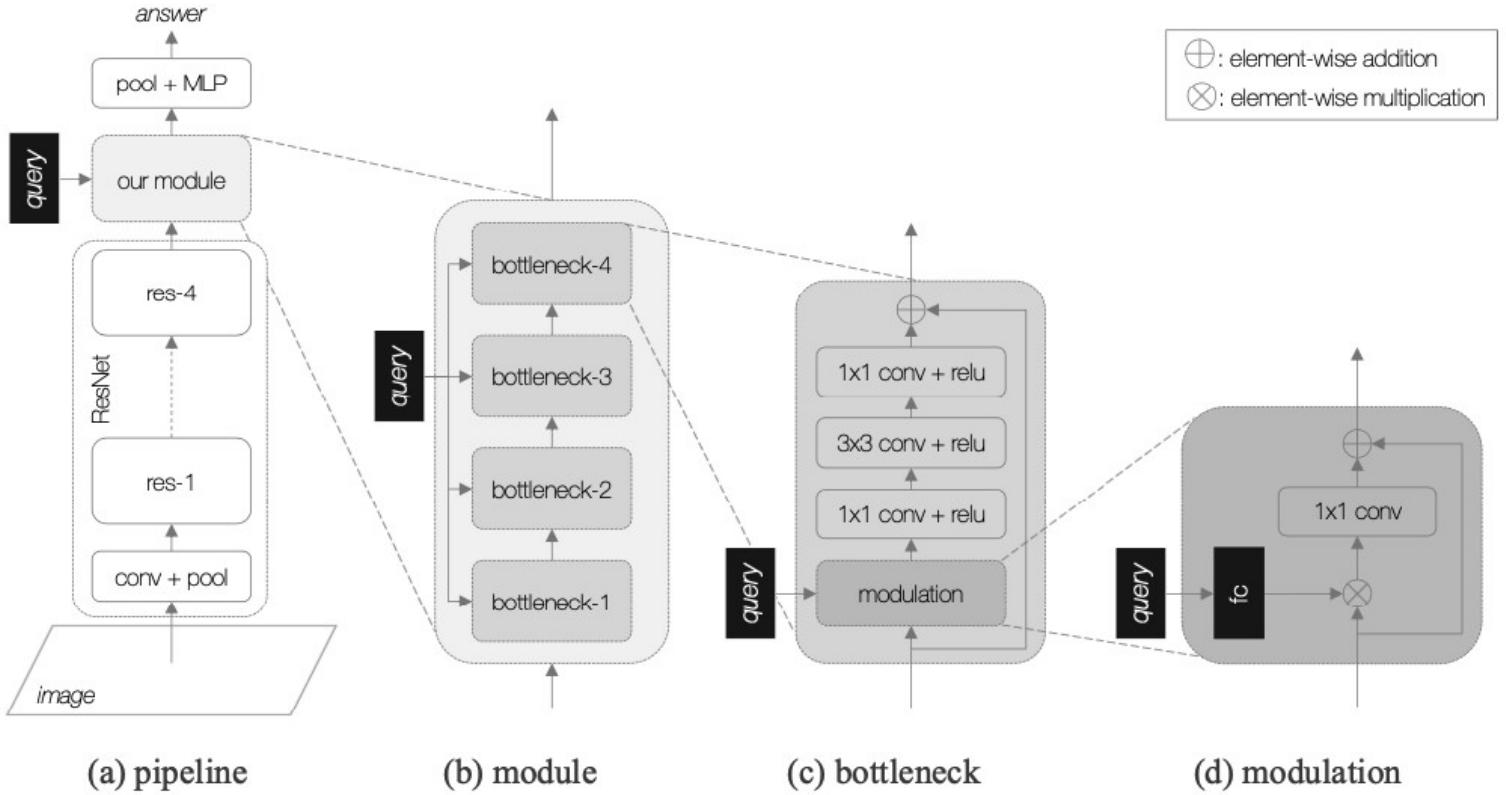


Figure 1: We study visual counting. Different from previous works that perform explicit, symbolic counting (left), we propose an implicit, holistic counter, MoVie, that directly modulates convolutions (right) and can outperform state-of-the-art methods on multiple benchmarks. Its simple design also allows potential generalization beyond counting to other visual reasoning tasks (bottom).

Nguyen, D. K., Goswami, V., & Chen, X. (2020, September). MoVie: Revisiting Modulated Convolutions for Visual Counting and Beyond. In *International Conference on Learning Representations*.

MoVie: Modulated Convolution for Visual counting

- Modulated convolution to fuse query and image locally, not globally



$$\bar{\mathbf{v}}_{\text{MoVie}} = \mathbf{v} \oplus \mathbf{W}^T(\mathbf{v} \otimes \Delta\gamma).$$

MoVie as a Counting Module for VQA

Results: Outperform state-of-the-arts on *three* major benchmarks in visual counting, namely HowMany-QA, Tally-QA and COCO.

Error cases:

1) Fail to recognize objects(Image modality)
2) Query is more complicated

Method	Backbone	#params (M)	FLOPs (G)	HowMany-QA		TallyQA-Simple		TallyQA-Complex	
				ACC ↑	RMSE ↓	ACC ↑	RMSE ↓	ACC ↑	RMSE ↓
MUTAN (2017)	R-152	60.2	-	45.5	2.93	56.5	1.51	49.1	1.59
Count module (2018)	R-101	44.6	-	54.7	2.59	70.5	1.15	50.9	1.58
IRLC (2018)	R-101	44.6	-	56.1	2.45	-	-	-	-
TallyQA (2019)	R-101+152	104.8	1883.5	60.3	2.35	71.8	1.13	56.2	**1.43**
TallyQA (FG-Only)	R-101	44.6	1790.9	-	-	69.4	1.18	51.8	1.50
MoVie	R-50	25.6	176.1	61.2	2.36	70.8	1.09	54.1	1.52
MoVie	X-101	88.8	706.3	**64.0**	**2.30**	**74.9**	**1.00**	**56.8**	**1.43**

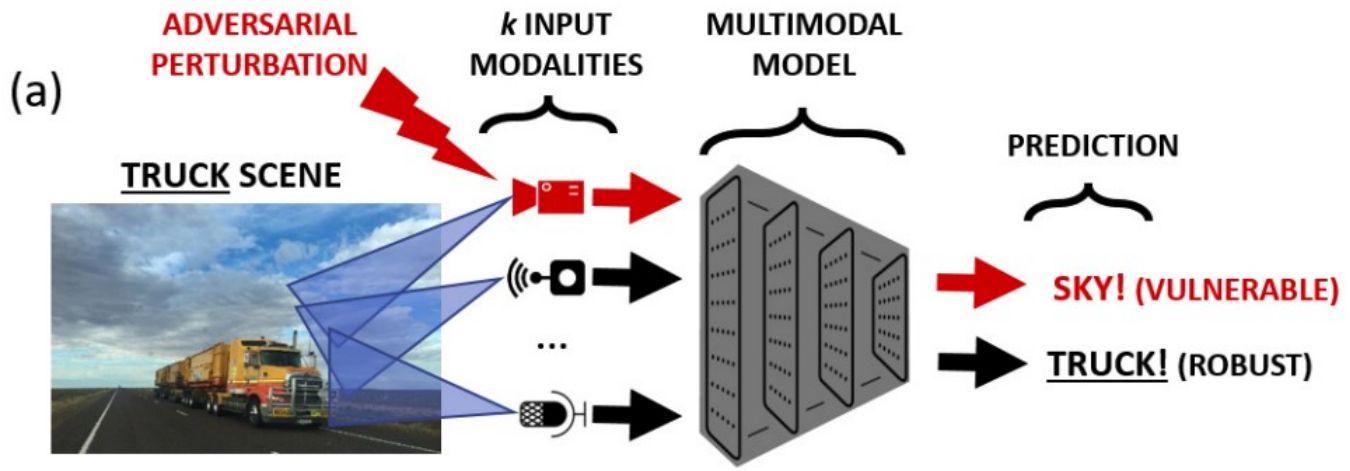
Table 2: **Open-ended counting** on Howmany-QA and TallyQA *test* set. MoVie outperforms prior arts with lower parameter counts and FLOPs. X: ResNeXt Xie et al. (2017).

Nguyen, D. K., Goswami, V., & Chen, X. (2020, September). MoVie: Revisiting Modulated Convolutions for Visual Counting and Beyond. In *International Conference on Learning Representations*.

Robustness of Multimodal models

Robustness of Multimodal models against single modality failure

- If one of the modalities (*e.g.*, RGB) receives a worst-case or adversarial perturbation, does the model fail to detect the truck in the scene?
- Does the model make a robust prediction using the remaining k − 1 unperturbed modalities (*e.g.*, LIDAR, audio, *etc*.)?



Yang, K., Lin, W. Y., Barman, M., Condessa, F., & Kolter, Z. (2021). Defending Multimodal Fusion Models Against Single-Source Adversaries. CVPR.

Robustness of Multimodal models against single modality failure

- Standard multimodal fusion practices are not sufficiently robust against worst-case perturbations on a single modality.
- E.g. Action recognition on EPIC-Kitchens.

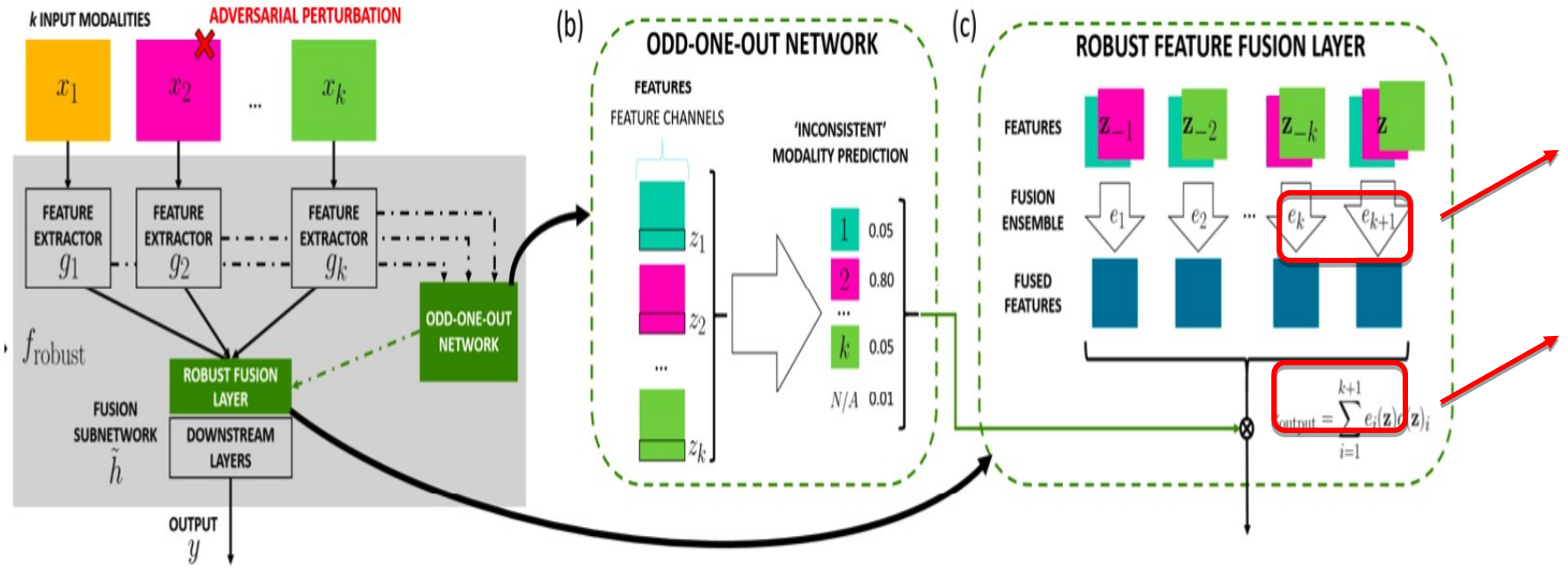
Fusion	Clean			Visual Perturbation			Motion Perturbation			Audio Perturbation		
	Verb	Noun	Action	Verb	Noun	Action	Verb	Noun	Action	Verb	Noun	Action
Oracle (Upper Bound)	-	-	-	55.8	31.4	21.9	50.0	37.2	23.8	53.9	39.2	25.6
Concat Fusion	59.0	42.1	30.2	0.1	0.0	0.0	0.2	0.0	0.0	0.1	0.0	0.0
Mean Fusion	56.8	40.4	27.6	0.3	0.8	0.0	0.3	0.3	0.0	0.4	0.3	0.0

- Or Sentiment Analysis on CMU-MOSI.

Fusion	Clean		Audio Perturbation		Video Perturbation		Text Perturbation	
	2-class	7-class	2-class	7-class	2-class	7-class	2-class	7-class
Oracle (Upper Bound)	-	-	78.64	49.10	73.36	47.84	69.82	40.28
Concat Fusion	79.82	49.69	56.92	21.38	51.23	19.75	39.50	9.97
Mean Fusion	78.09	46.14	52.63	20.75	49.37	17.02	35.50	8.88

Odd-one-out Network

- Odd-one-out learning is a self-supervised task that aims to identify the inconsistent modality from a set consistent elements.



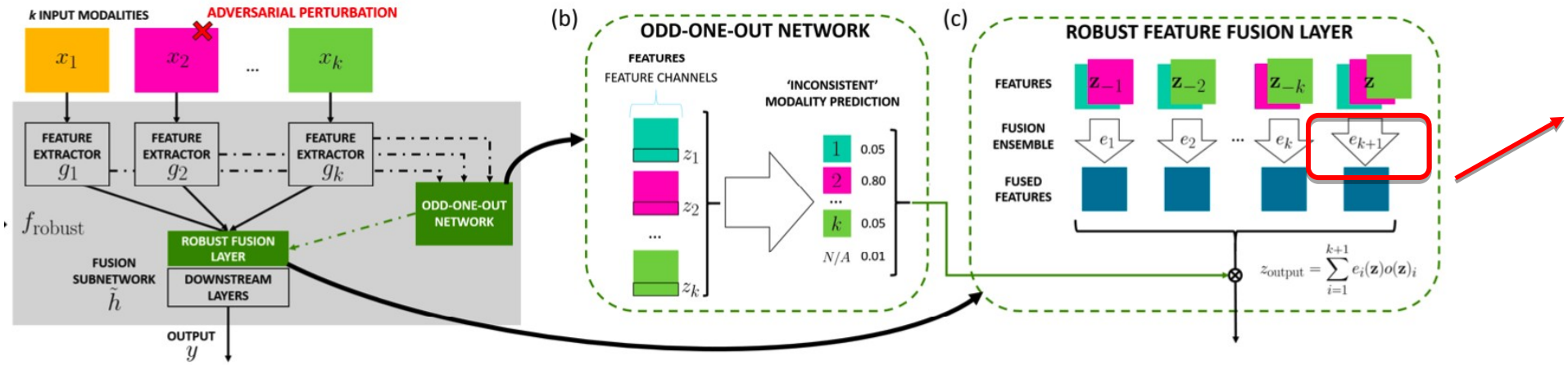
The probability that modality *i* has been perturbed.

The probability that none of the modalities are perturbed.

$$= -\mathbf{E}_{\substack{(\mathbf{x},y)\sim\mathcal{D} \\ z_i=g_i(x_i)}} \left[\log o(\mathbf{z})_{k+1} + \sum_{i=1}^{k} \log o(z_i^*, \mathbf{z}_{-\mathbf{i}})_i\right],$$

Robust Feature Fusion Layer

- Robust Fusion Layer aims to maximize the weight for the consistent modalities excluding the perturbed modality, by using the output from the odd-one-out layer.



: a fusion of features from all the modalities *except* for i.

fuses features from all the modalities.

$$e_i(\mathbf{z}) = \mathrm{NN}\big(\oplus \mathbf{z}_{-i}\big)\ \forall i \in [k], \quad e_{k+1}(\mathbf{z}) = \mathrm{NN}\big(\oplus \mathbf{z}\big),$$

$$z_{\text{output}} = \sum_{i=1}^{k+1} e_i(\mathbf{z})o(\mathbf{z})_i,$$

Output of Odd-one-out network

Performance and Future Direction

- The model demonstrated significant robustness improvement against single modality failure, without affecting its performance on clean data.

- E.g. Action recognition on EPIC-Kitchens

Fusion	Clean			Visual Perturbation			Motion Perturbation			Audio Perturbation		
	Verb	Noun	Action	Verb	Noun	Action	Verb	Noun	Action	Verb	Noun	Action
Oracle (Upper Bound)	-	-	-	55.8	31.4	21.9	50.0	37.2	23.8	53.9	39.2	25.6
Concat Fusion	59.0	42.1	30.2	0.1	0.0	0.0	0.2	0.0	0.0	0.1	0.0	0.0
Mean Fusion	56.8	40.4	27.6	0.3	0.8	0.0	0.3	0.3	0.0	0.4	0.3	0.0
LEL+Robust [17]	61.2	**43.1**	30.5	22.3	11.6	6.6	25.4	24.6	12.0	20.4	17.7	8.0
Gating+Robust [16, 15]	60.9	43.0	30.6	26.0	10.9	6.2	35.9	26.9	14.3	21.3	16.2	7.0
Ours	**61.5**	42.5	**31.4**	**48.0**	**24.2**	**16.8**	**48.5**	**35.6**	**22.1**	**46.5**	**33.3**	**22.1**
Δ-Clean	**2.5**	**0.3**	**1.2**	**47.7**	**23.4**	**16.8**	**48.2**	**35.3**	**22.1**	**46.1**	**33.0**	**22.1**
Δ-Robust	**0.3**	-0.6	**0.8**	**22.0**	**13.3**	**10.2**	**12.6**	**8.7**	**7.8**	**25.2**	**15.6**	**14.1**

Future Directions:

1) approaches for defending attacks on multiple modalities as once
2) physically-realisable attacks, etc.

Intermediate Fusion

MMTM: Multimodal Transfer Module for CNN Fusion

- Late fusion is still the predominant method utilized for multimodal learning.

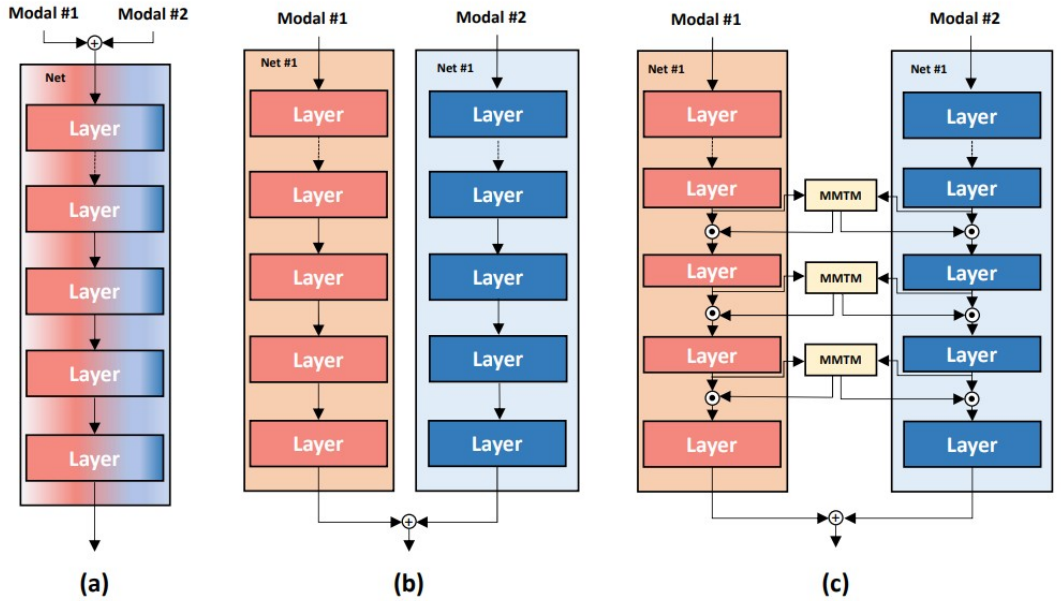
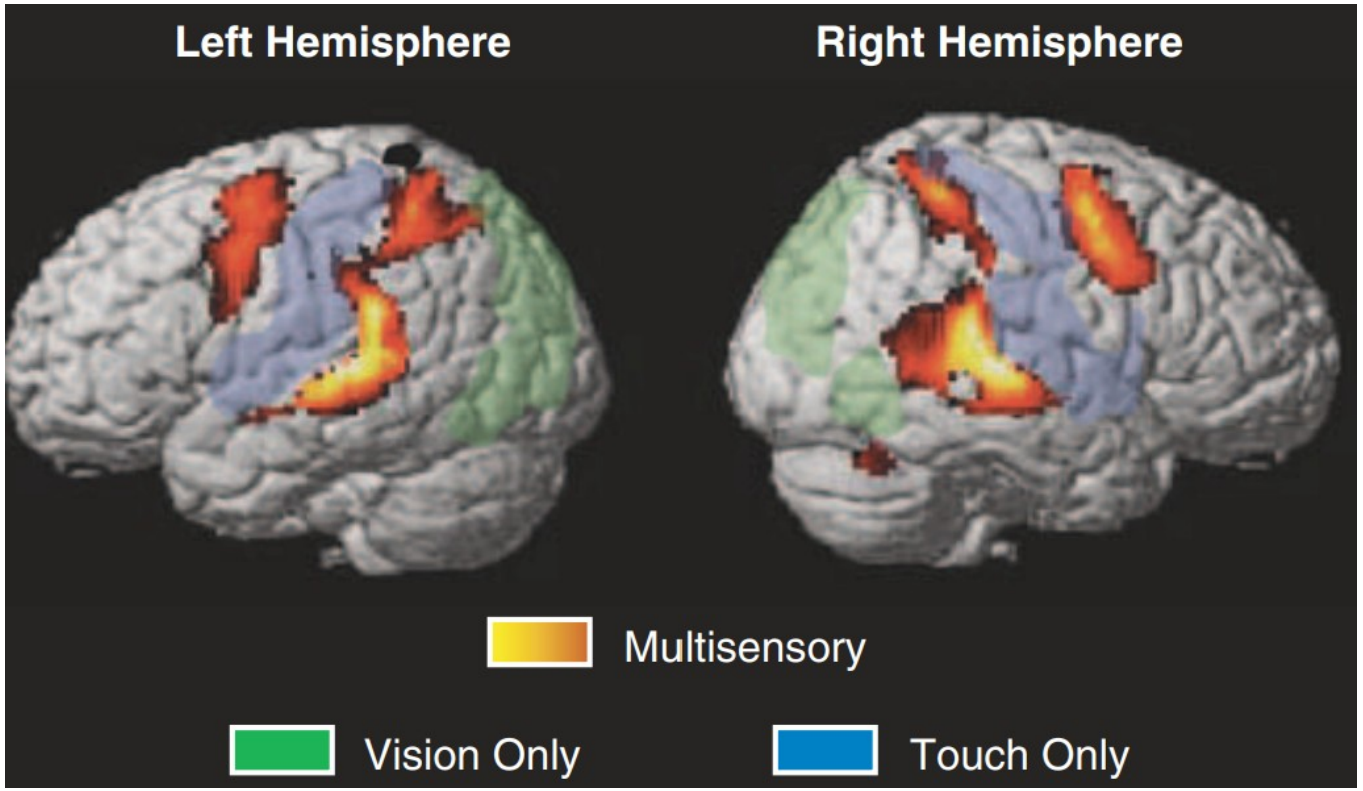


Figure 1. (a) early fusion (b) late fusion (c) intermediate fusion with Multimodal Transfer Module (MMTM). MMTM operates between CNN streams and uses information from different modalities to recalibrate channel-wise features in each modality.

Joze, Hamid Reza Vaezi, et al. "MMTM: Multimodal transfer module for CNN fusion." *CVPR* 2020.

MMTM: Multimodal Transfer Module for CNN Fusion

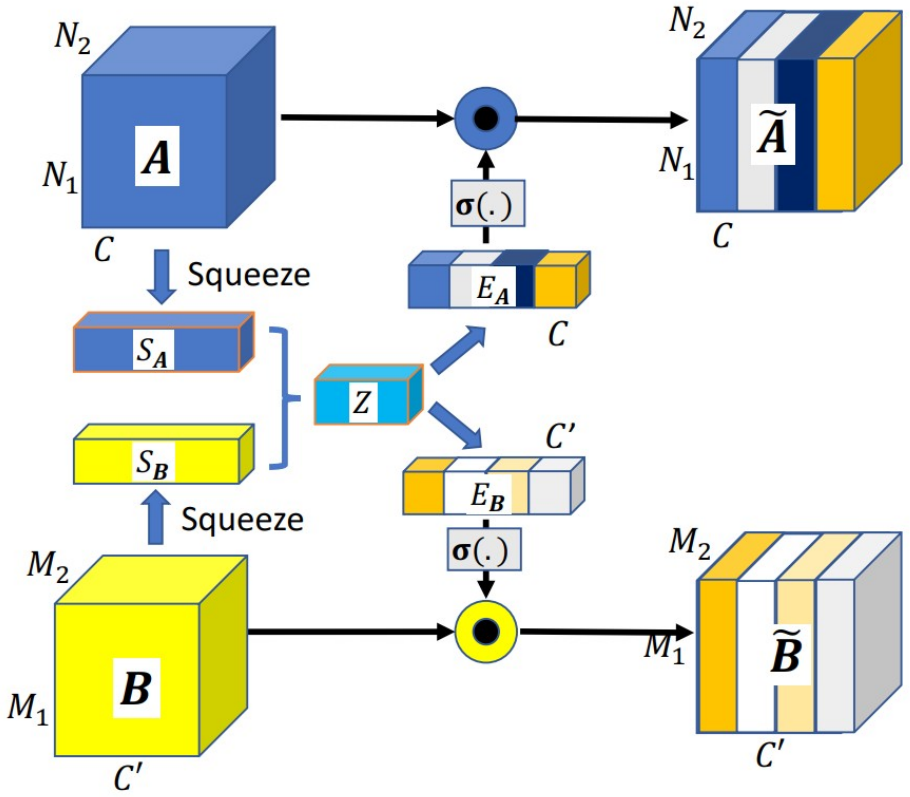
- Intermediate fusion exists in neuroscience.



Emiliano Macaluso. Multisensory processing in sensory specific cortical areas. The neuroscientist, 2006.

MMTM: Multimodal Transfer Module for CNN Fusion

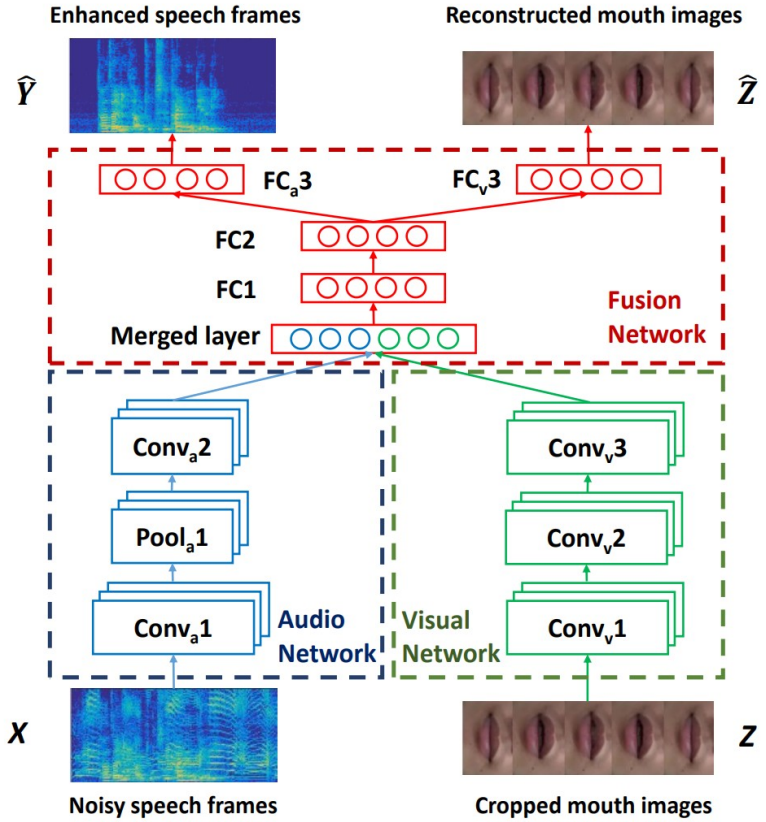
- Mechanism for intermediate multimodality fusion.



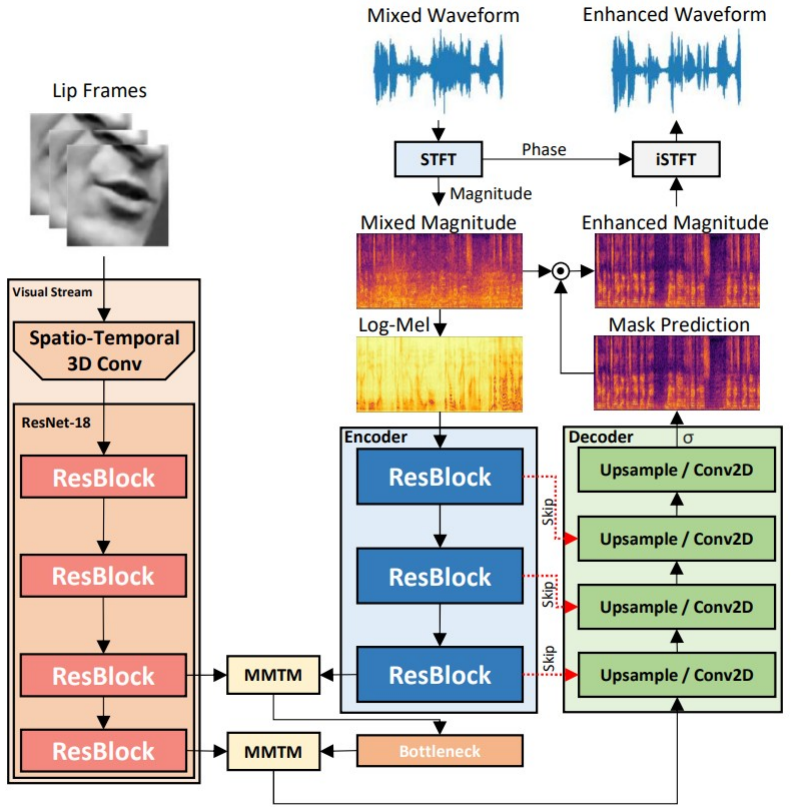
Joze, Hamid Reza Vaezi, et al. "MMTM: Multimodal transfer module for CNN fusion." *CVPR* 2020.

MMTM: Multimodal Transfer Module for CNN Fusion

- Applying intermediate fusion in **Audio visual Speech enhancement**



SOTA model with late fusion

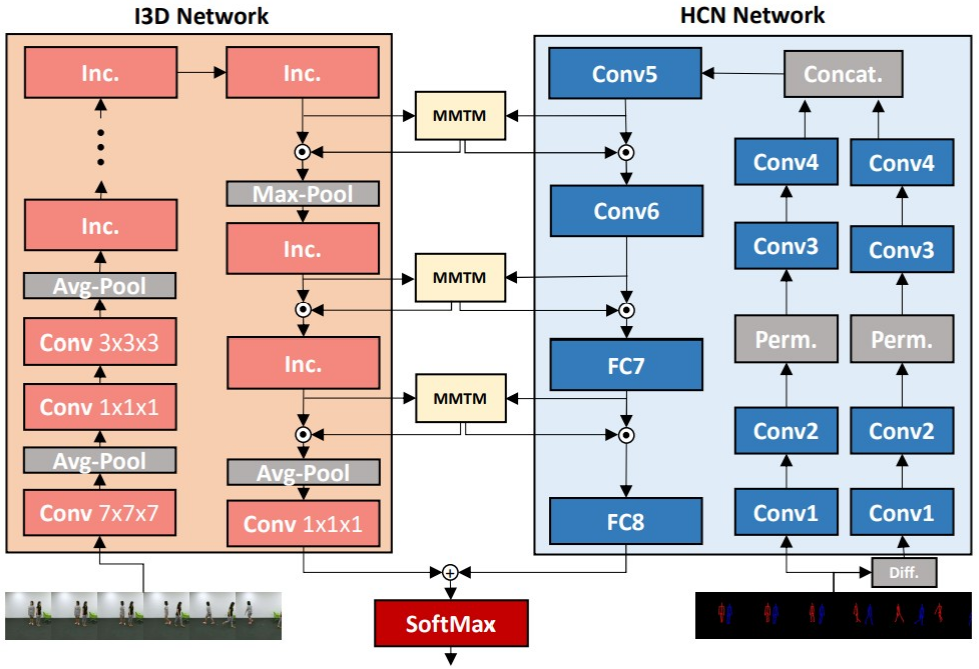


MMTM intermediate fusion

Hou et al. Audio-Visual Speech Enhancement Using Multimodal Deep Convolutional Neural Networks. 2017.

MMTM: Multimodal Transfer Module for CNN Fusion

- Applying intermediate fusion in **Human Action Recognition**



RGB video

3D Pose

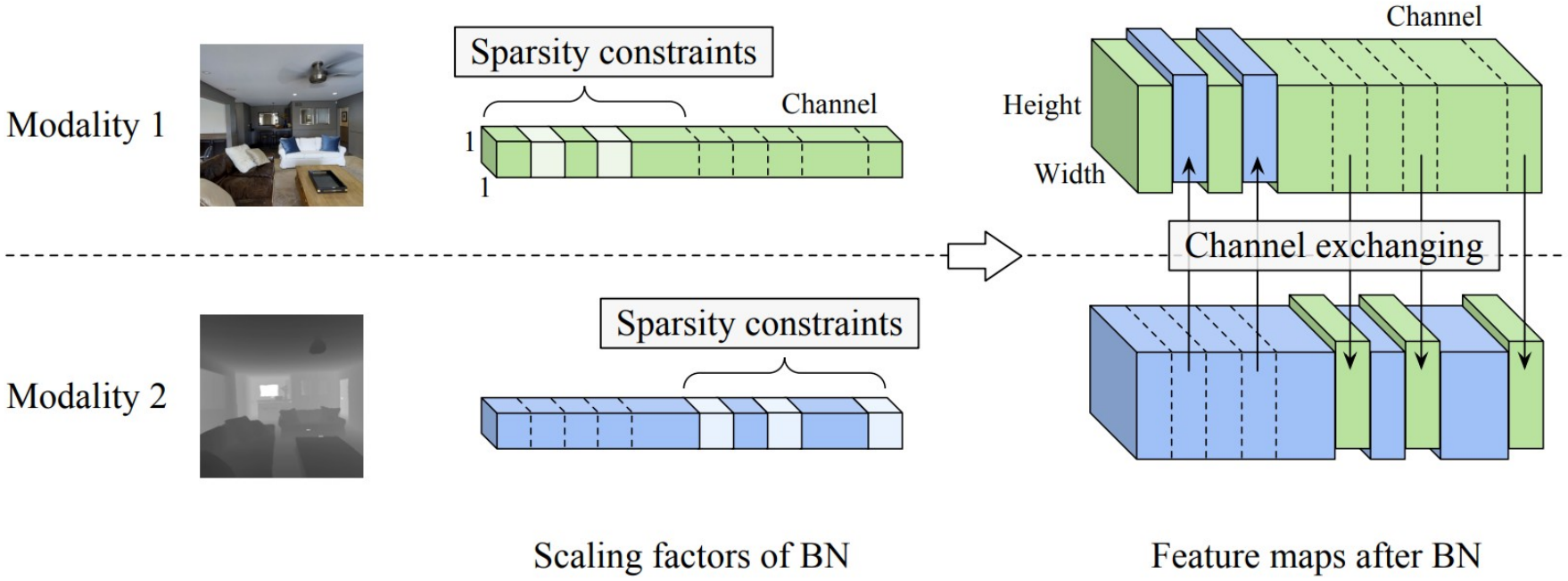
	Method	Input Modalities	Accuracy
Unimodal	HCN [53]	Pose	85.24
	Infalated Resnet-50 [76]	RGB	83.91
Late fusion	Two Stream [14]	RGB+Pose	88.60
Intermediate fusion	GMU [23]	RGB+Pose	85.80
	CentralNet [18]	RGB+Pose	89.36
	MFAS [17]	RGB+Pose	90.04
	Ours	RGB+Pose	**90.11**

Table 5. Comparison of state-of-the-art multimodal fusion algorithms on the NTU-RGBD dataset [55]. All methods use HCN and Infalated Resnet-50 backbone unimodal architectures.

Joze, Hamid Reza Vaezi, et al. "MMTM: Multimodal transfer module for CNN fusion." *CVPR* 2020.

Deep Multimodal Fusion by Channel Exchanging

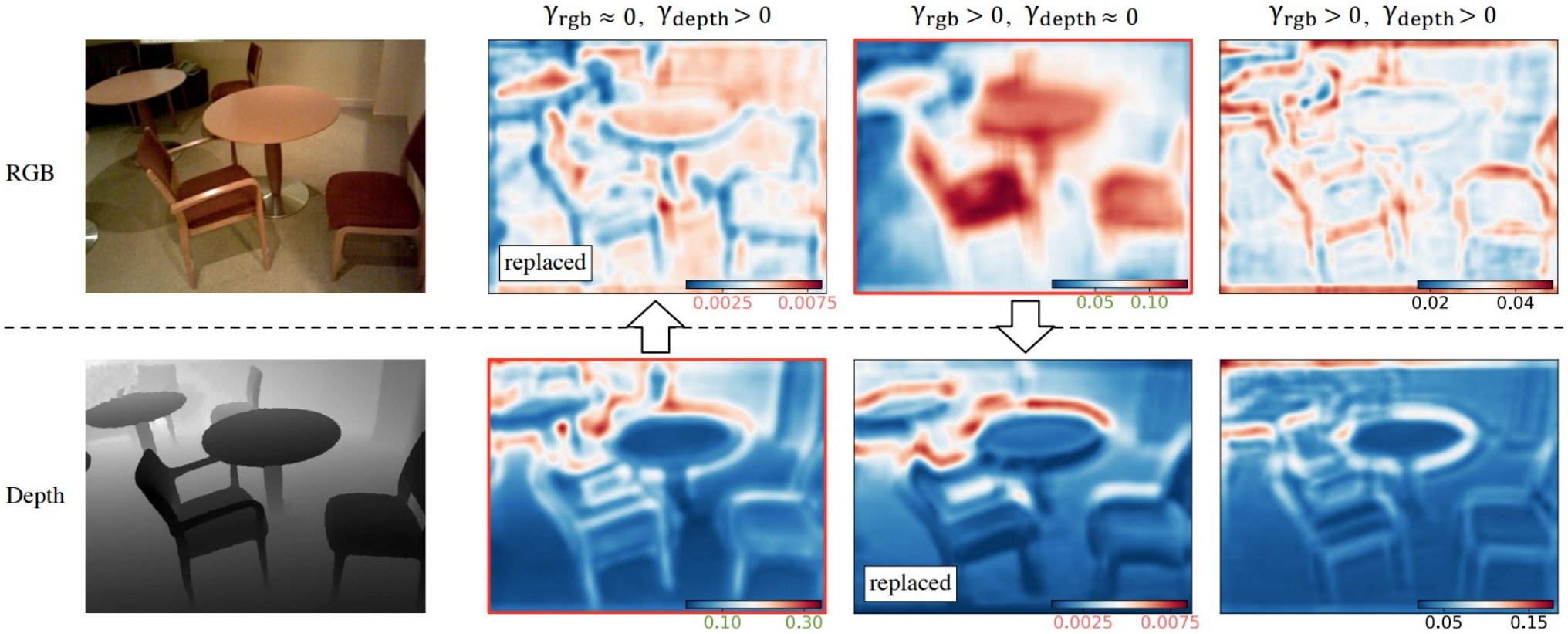
- Another idea of intermediate fusion using channel exchanging



Wang, Yikai, et al. "Deep multimodal fusion by channel exchanging." *NIPS* 2020.

Deep Multimodal Fusion by Channel Exchanging

- Weak response channel in one modality get replaced by mean response in another modality within its group.



Wang, Yikai, et al. "Deep multimodal fusion by channel exchanging." *NIPS* 2020.

Deep Multimodal Fusion by Channel Exchanging

- Show performance improvement on semantic segmentation task.

Table 3: Comparison with SOTA methods on semantic segmentation.

Modality	Approach	Backbone Network	NYUDv2 Pixel Acc. (%)	NYUDv2 Mean Acc. (%)	NYUDv2 Mean IoU (%)	SUN RGB-D Pixel Acc. (%)	SUN RGB-D Mean Acc. (%)	SUN RGB-D Mean IoU (%)
RGB	FCN-32s [34]	VGG16	60.0	42.2	29.2	68.4	41.1	29.0
	RefineNet [32]	ResNet101	73.8	58.8	46.4	80.8	57.3	46.3
	RefineNet [32]	ResNet152	74.4	59.6	47.6	81.1	57.7	47.0
RGB-D	FuseNet [19]	VGG16	68.1	50.4	37.9	76.3	48.3	37.3
	ACNet [22]	ResNet50	-	-	48.3	-	-	48.1
	SSMA [45]	ResNet50	75.2	60.5	48.7	81.0	58.1	45.7
	SSMA [45] †	ResNet101	75.8	62.3	49.6	81.6	60.4	47.9
	CBN [46] †	ResNet101	75.5	61.2	48.9	81.5	59.8	47.4
	3DGNN [37]	ResNet101	-	-	-	-	57.0	45.9
	SCN [31]	ResNet152	-	-	49.6	-	-	50.7
	CFN [30]	ResNet152	-	-	47.7	-	-	48.1
	RDFNet [29]	ResNet101	75.6	62.2	49.1	80.9	59.6	47.2
	RDFNet [29]	ResNet152	76.0	62.8	50.1	81.5	60.1	47.7
	Ours-RefineNet (single-scale)	ResNet101	76.2	62.8	51.1	82.0	60.9	49.6
	Ours-RefineNet	ResNet101	77.2	63.7	51.7	82.8	61.9	50.2
	Ours-RefineNet	ResNet152	77.4	64.8	52.2	83.2	62.5	50.8
	Ours-PSPNet	ResNet152	**77.7**	**65.0**	**52.5**	**83.5**	**63.2**	**51.1**

† indicates our implemented results.

Wang, Yikai, et al. "Deep multimodal fusion by channel exchanging." *NIPS* 2020.

Deep Multimodal Fusion by Channel Exchanging

- Show performance improvement on multimodal image translation task.

Table 4: Comparison on image-to-image translation. Evaluation metrics are FID/KID ($\times 10^{-2}$). Lower values indicate better performance.

Modality	Ours	Baseline	Early	Middle	Late	All-layer
Shade+Texture →RGB	**62.63 / 1.65**	Concat	87.46 / 3.64	95.16 / 4.67	122.47 / 6.56	78.82 / 3.13
		Average	93.72 / 4.22	93.91 / 4.27	126.74 / 7.10	80.64 / 3.24
		Align	99.68 / 4.93	95.52 / 4.75	98.33 / 4.70	92.30 / 4.20
		Self-att.	83.60 / 3.38	90.79 / 3.92	105.62 / 5.42	73.87 / 2.46
Depth+Normal →RGB	**84.33 / 2.70**	Concat	105.17 / 5.15	100.29 / 3.37	116.51 / 5.74	99.08 / 4.28
		Average	109.25 / 5.50	104.95 / 4.98	122.42 / 6.76	99.63 / 4.41
		Align	111.65 / 5.53	108.92 / 5.26	105.85 / 4.98	105.03 / 4.91
		Self-att.	100.70 / 4.47	98.63 / 4.35	108.02 / 5.09	96.73 / 3.95

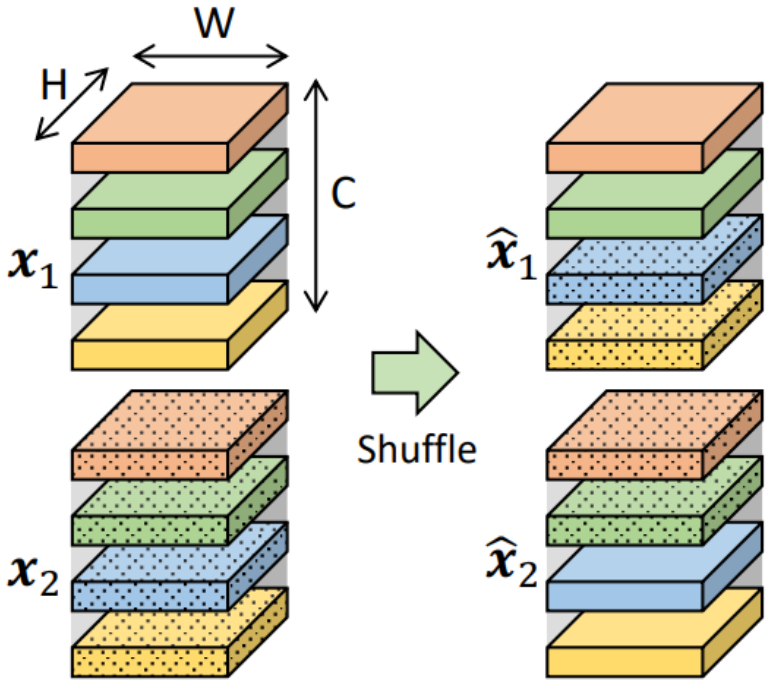
Wang, Yikai, et al. "Deep multimodal fusion by channel exchanging." *NIPS* 2020.

Deep Multimodal Fusion by Channel Exchanging

- The idea of channel exchanging also exists in another work in 2020, showing similar performance on segmentation task.

Modality 1 feature

Modality 2 feature



Method	Data modality	Backbone	Pixel acc.	Mean acc.	IoU	#Params.
RefineNet [21]	RGB	ResNet101	73.8	58.8	46.4	118.10M
RefineNet [21]	RGB	ResNet152	74.4	59.6	47.6	133.74M
CFN [19]	RGB-D	ResNet152	-	-	47.7	-
SCN [20]	RGB-D	ResNet152	-	-	49.6	-
RDFNet [17]	RGB-D	ResNet101	75.6	62.2	49.1	366.71M
RDFNet [17]	RGB-D	ResNet152	76.0	62.8	50.1	398.00M
RefineNet †	RGB	ResNet101	73.8	59.0	46.5	118.10M
RefineNet †	Depth	ResNet101	64.0	45.6	34.3	118.10M
AsymFusion	RGB-D	ResNet101	76.6	63.5	50.8	**118.20M**
AsymFusion	RGB-D	ResNet152	**77.0**	**64.0**	**51.2**	**133.89M**

† indicates our re-implemented results

Wang, Yikai, et al. "Learning Deep Multimodal Feature Representation with Asymmetric Multi-layer Fusion." *Proceedings of the 28th ACM International Conference on Multimedia*. 2020.

Multimodal Model Architectures

Motivation/Overview

- Unified Backbone
  - Can we come up with a unified model backbone for different inputs?

Language	Vision	Audio	Point Cloud
Transformer	2D conv.	1D conv.	Low-res. grid

- Modality Fusion
  - How to design a proper fusion mechanism under such unified backbone model?
  - Early? Late? Something else?

Unified Backbone

- Transformer architecture is widely used in NLP tasks
- However, the space/time complexity is quadratic
  - QKV attention:
  - , is for an image
- Do we really need such a large ?
  - No. There is redundant information in an image for example

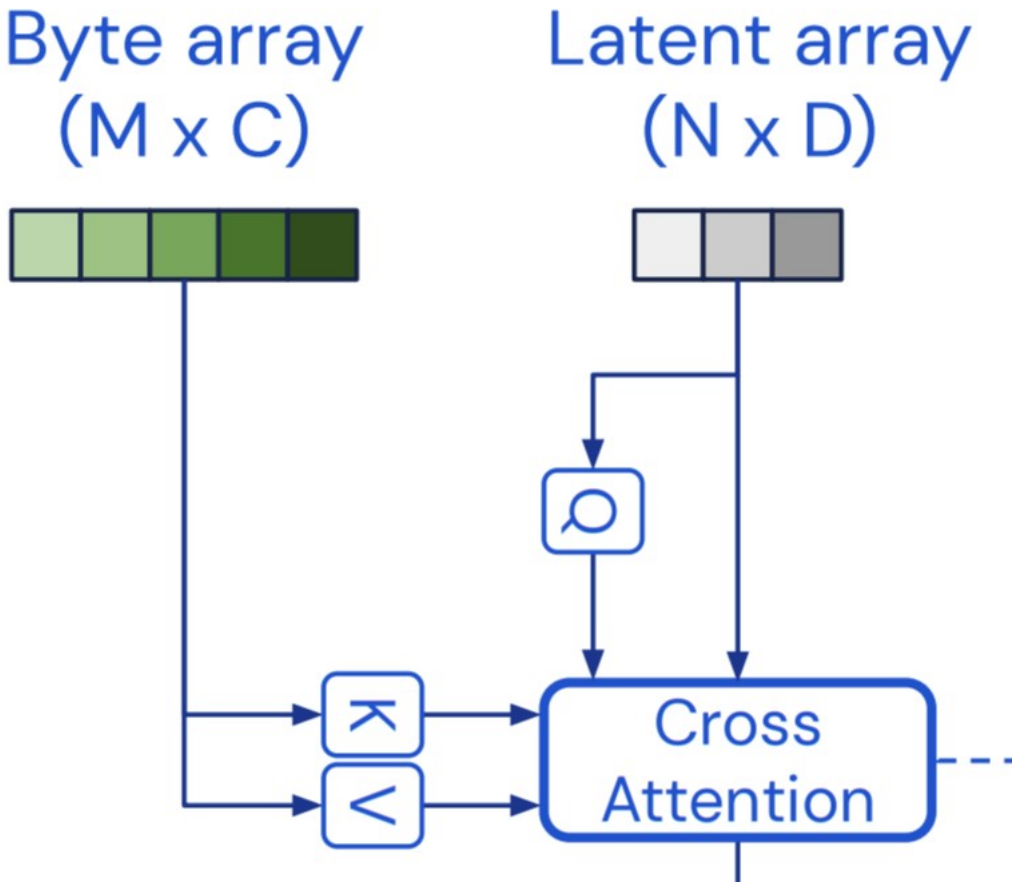
Deepmind, Perceiver: General Perception with Iterative Attention, ICML 2021

Dimension Reduction

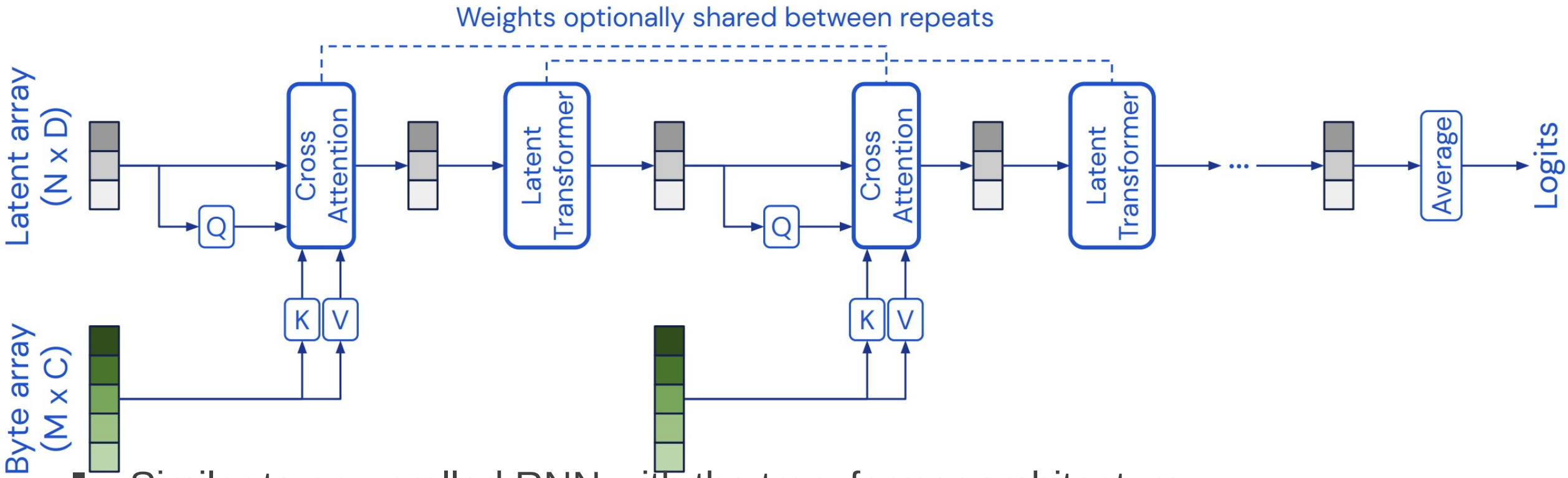
- Change to , where

- Latent array is randomly initialized
- It serves as a **bottleneck** attention layer
- C and D are just #channels

- However, the model becomes less expressive. What should we do?



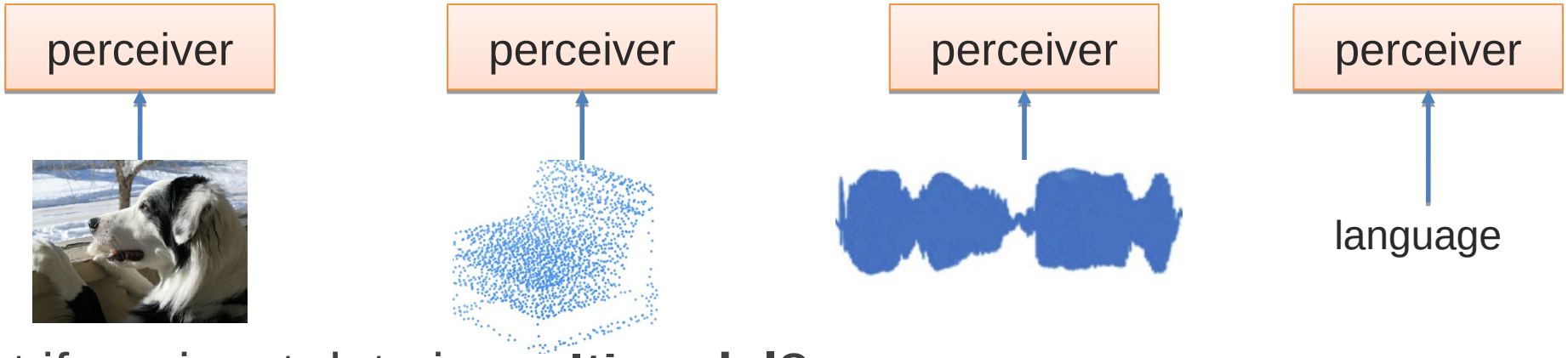
Iterative Attention



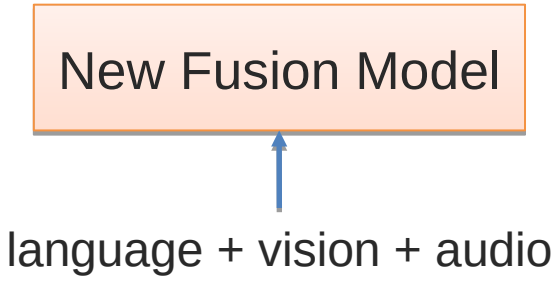
- Similar to an unrolled RNN with the transformer architecture

Modailty Fusion

- We have one backbone that can be applied to each modality **separately**
  - Input is still unimodal in each task

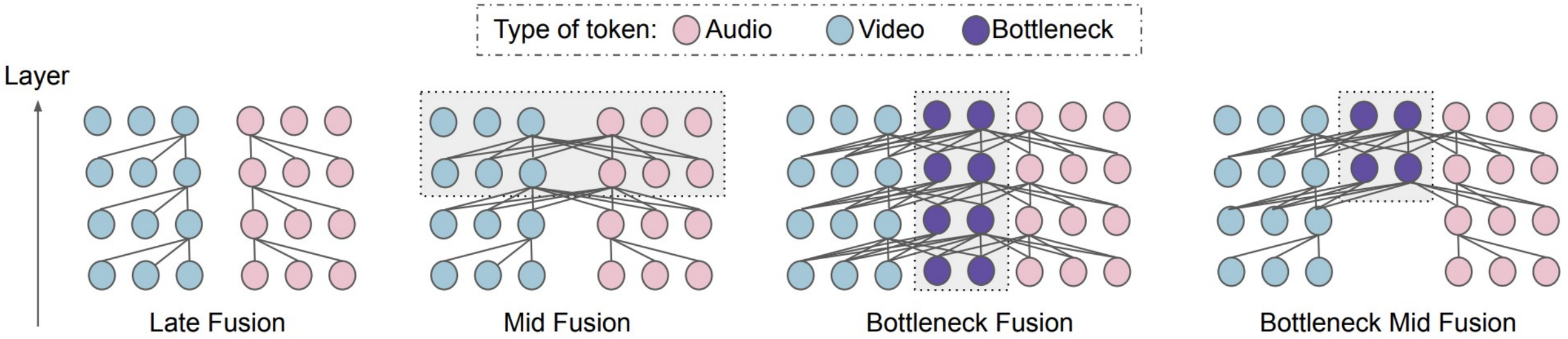


- What if our input data is **multimodal**?



Modality Fusion

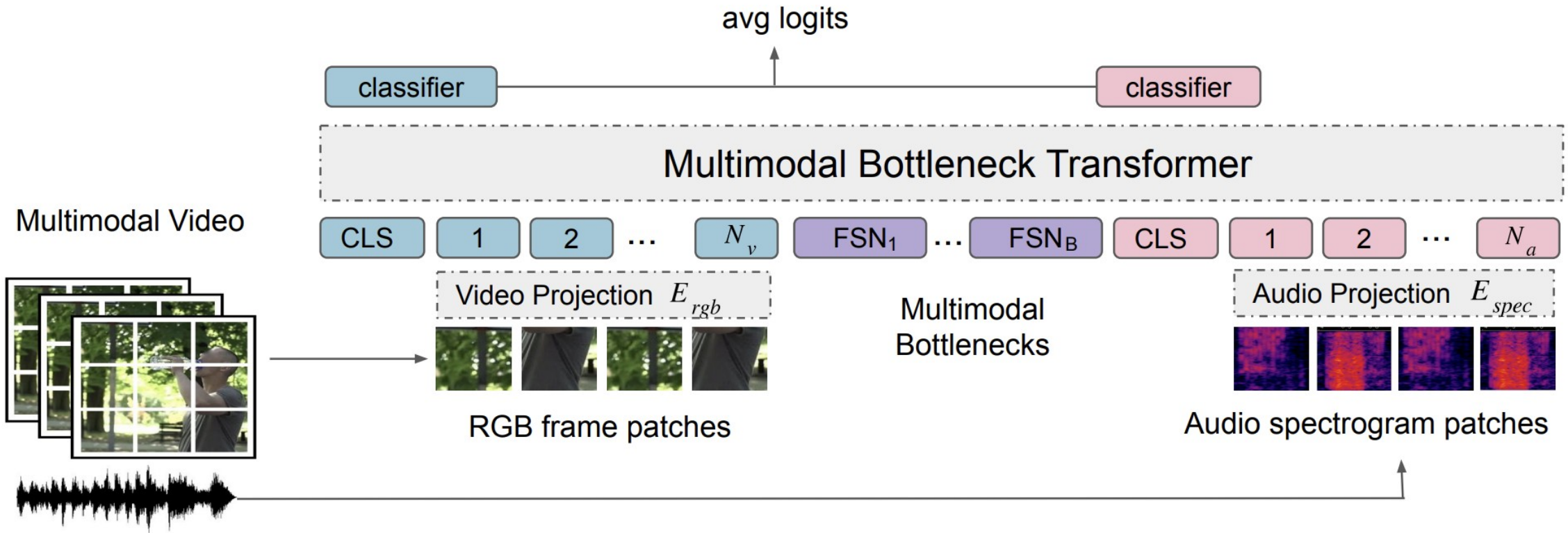
- Apply the same **bottleneck** concept, but this time it's cross-modal
- Pink and green are transformers



Google Research, Attention Bottlenecks for Multimodal Fusion, NIPS 2021

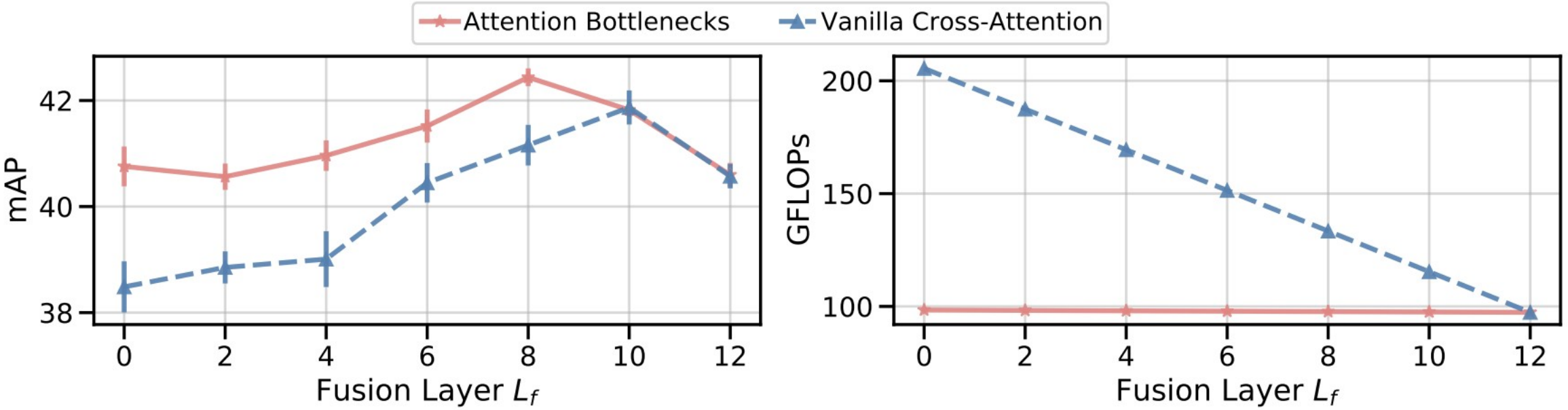
Modality Fusion

- Insert bottleneck FSN tokens between modalities
- All cross-modal attention is retricted to flow via FSNs
- FSNs are updated twice, first with visual, and then with audio information



Practical Impact

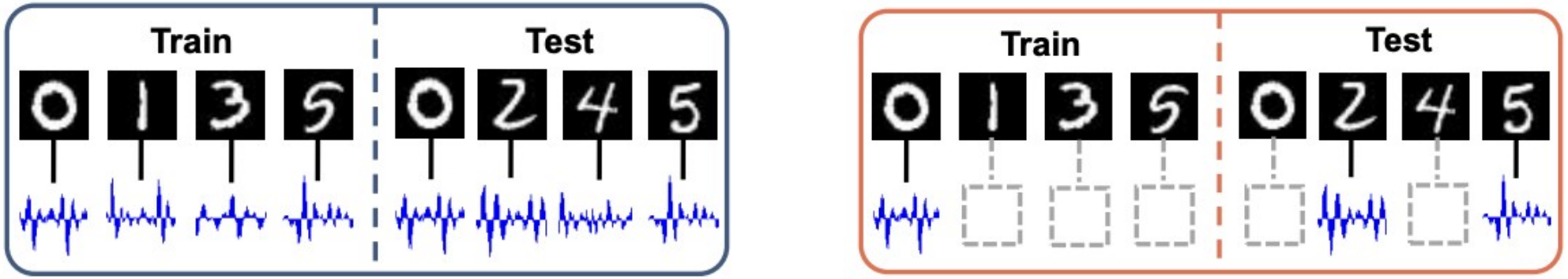
- # of FSN=4 ($FSN_{B=4}$ in the prev. slide) in the experiments
- => only the last 12-x layers are equipped with FSNs
- Mid-Late fusion works the best
- FSN computational cost is almost constant
  - upadated separately with two modalities and B is only 4



Missing Modality

Motivation

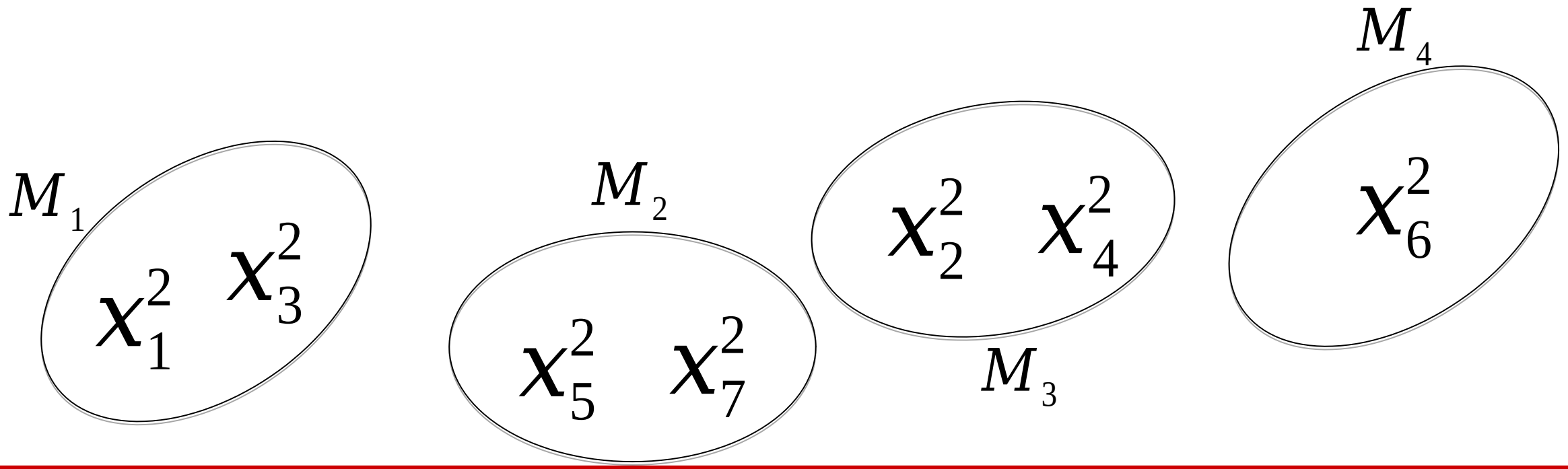
- Usually we have complete modality data (left)
- What if data from a modality is severly missing (90%) during both training **AND** testing time (right)
- Can we generate pseudo data for those missing instances?



Ma et al, SMIL: Multimodal Learning with Severely Missing Modality, AAAI 2021

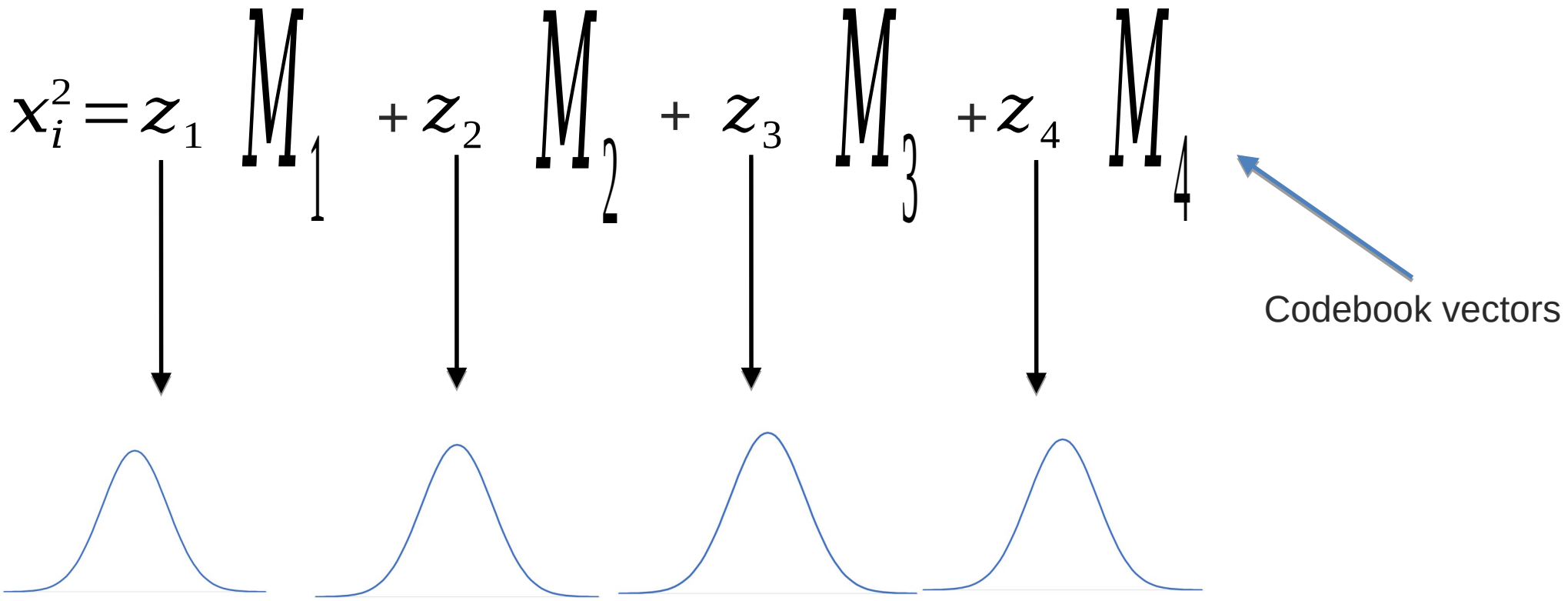
Codebook Learning

- Suppose some portion of the data is complete , and the remaining portion is missing
- We want to learn a codebook from by K-means or PCA
  - 4 vectors in the codebook for example:



Prior Learning

- We collect all the s and model them using a gaussian random variable
  - Modeling the prior , mean field approximation



Variational Inference

- We can use CVAE to model the missing data
  - is the observed variable and is the conditional
  - Sample to optimize the ELBO

learned codebook prior

$$\mathcal{L}_{\boldsymbol{\theta},\boldsymbol{\psi}} = \boldsymbol{E}_{q(\mathbf{z}|\mathbf{X};\boldsymbol{\theta},\boldsymbol{\psi})}[\log p(\mathbf{Y}|\mathbf{X},\mathbf{z};\boldsymbol{\theta})] - \mathrm{KL}[q(\mathbf{z}|\mathbf{X};\boldsymbol{\psi})\|p(\mathbf{z}|\mathbf{X})]$$

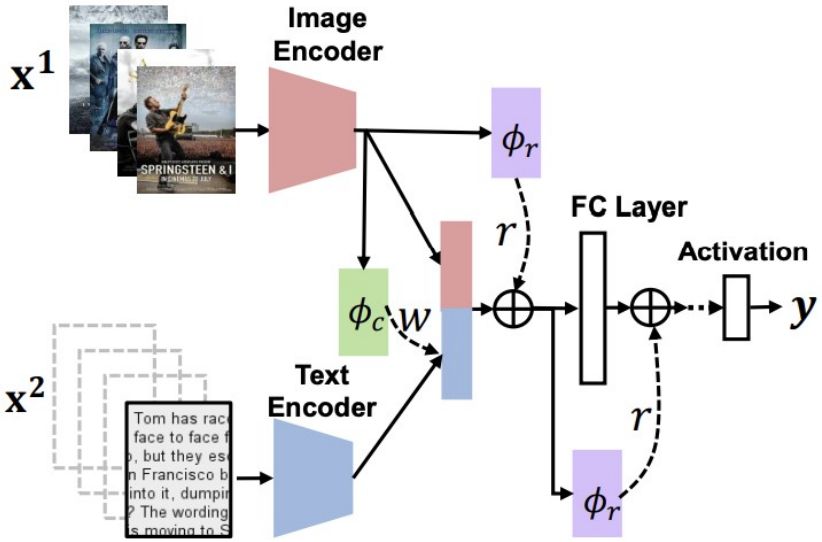
inference network

recognition network

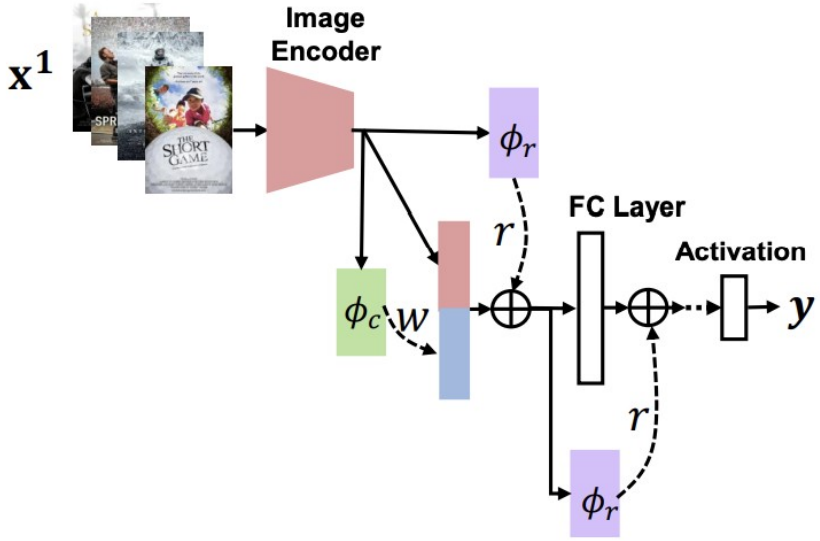
- The original paper also uses a meta learning framework to stabalize the training process
  - Optimize the inference network more frequently

Full Generative Story

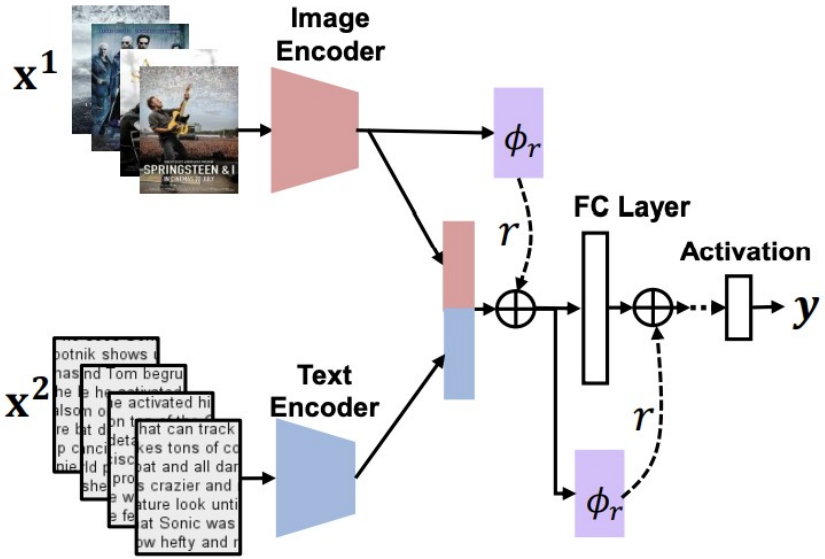
- For complete data, we perform MLE training
- For missing data, we perform variational inference to infer pseudo data
- , then use it to weighted-sum codebook vectors as



(a) Training with severely missing modality



(b) Testing with single modality



(c) Testing with full modality