



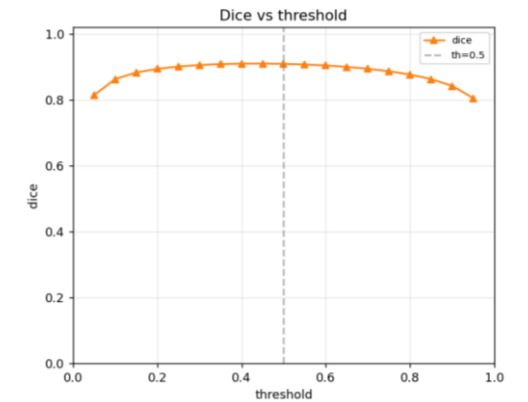
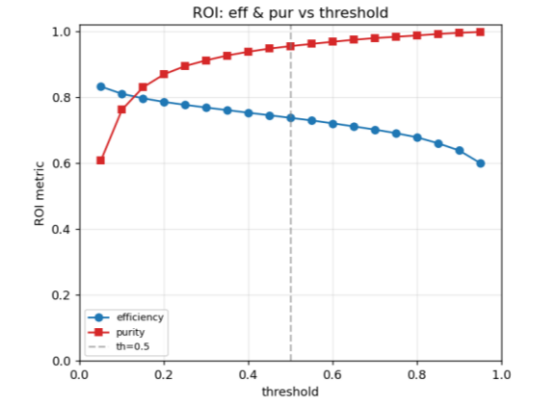
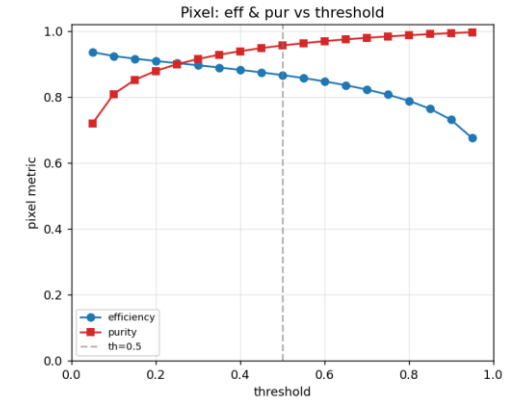
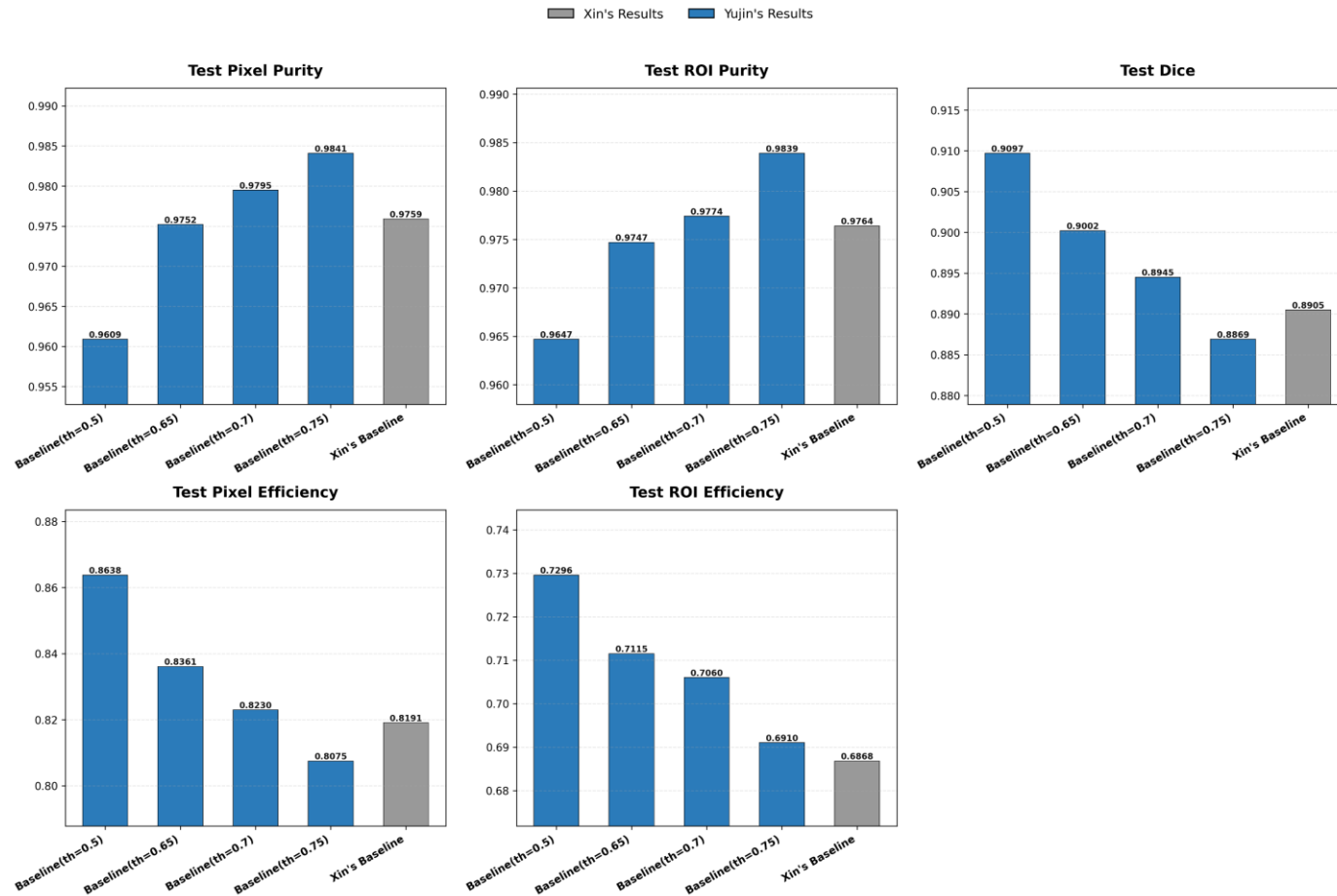
Wire-Cell ProtoDUNE Local Meeting

Yujin Park
Chung-Ang University
Jul08, 2026 (Wed)

Wire-Cell DNN Signal Processing

PDHD Teacher Model Validation (Before KD)

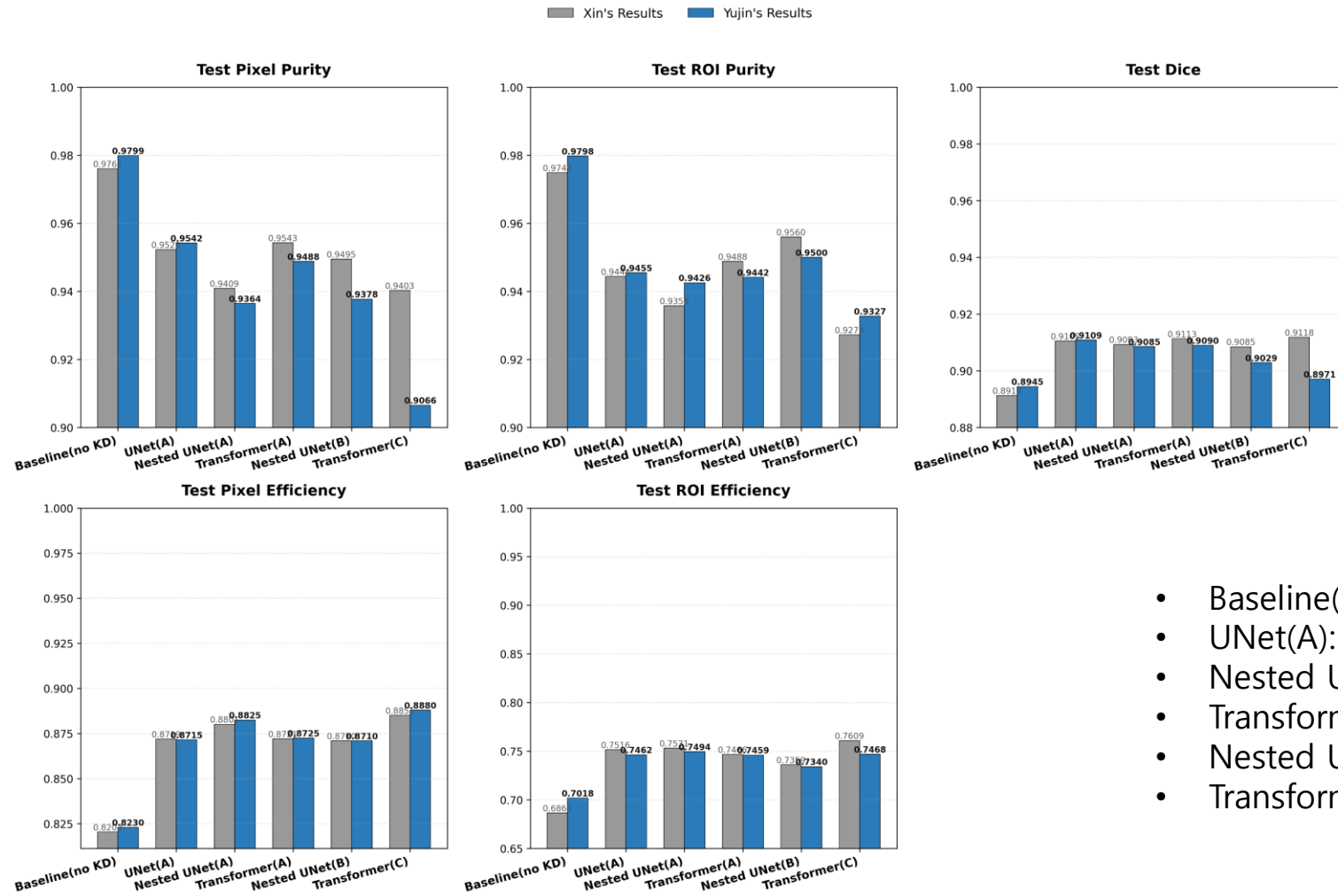
- Val/Test metrics values mismatch between My and Xin's results
- Changing thresholds(0.5, 0.65, 0.7, 0.75) of baseline model
- My results w/ th 0.7 $\approx$ Xin's results
- My model predicts more aggressively.



Wire-Cell DNN Signal Processing

PDHD KD Validation - Evaluation with Held-out Test set

- Depending on the model, the level of aggressiveness in training varies.



- Baseline(no KD): th=0.7000
- UNet(A): th=0.6000
- Nested UNet(A): th=0.4000
- Transformer(A): th=0.6500
- Nested UNet(B): th=0.6000
- Transformer(C): th=0.2500

Wire-Cell DNN Signal Processing

PDHD Quantization Validation

Xin's Results

Yujin's Results

Category	Variant	Weight Mem (MB)	CPU lat med (ms)	CPU lat p95 (ms)	Peak RSS (MB)	speedup vs FP32 of same row	peak RSS Decrement
Baseline Model (small-training)	FP32	19.9	890.9	919.3	4331	-	-
		19.9	409.4	417.8	4272	-	-
	INT8 PTQ	5.0	89.8	90.4	3053	9.92×	29.5% ↓
		5.0	57.4	59.6	3067	7.13×	28.2% ↓
KD (Strategy B) Baseline <- Nested UNet	FP32	19.9	851.1	898.4	6170	-	-
		19.9	408.2	417.6	6296	-	-
	INT8 PTQ	5.0	85.0	91.3	4886	10.0×	20.8% ↓
		5.0	60.1	61.8	5045	6.79×	19.9% ↓
	INT8 QAT	10.03	84.6	90.9	4377	10.1×	29.1% ↓
		10.03	59.6	62.1	4397	6.85×	30.2% ↓
KD (Strategy C) Baseline <- Transformer	FP32	19.9	834.7	880.4	6193	-	-
		19.9	409.1	421.2	6166	-	-
	INT8 PTQ	5.0	84.0	89.5	4975	9.94×	19.7% ↓
		5.0	63.4	66.6	4887	6.46×	20.7% ↓
	INT8 QAT	10.03	86.6	88.4	4366	9.64×	29.5% ↓
		10.03	46.8	49.3	4397	8.74×	28.7% ↓

Measured in wcgpu1

- Weight memories are all same, and peak RSS decrements are similar(~29%).
 - ~ 6.8 to 8.7 times speedup compared to FP32, while ~ 10 times speedup in Xin's results.
- (Speed is affected by its running environment)

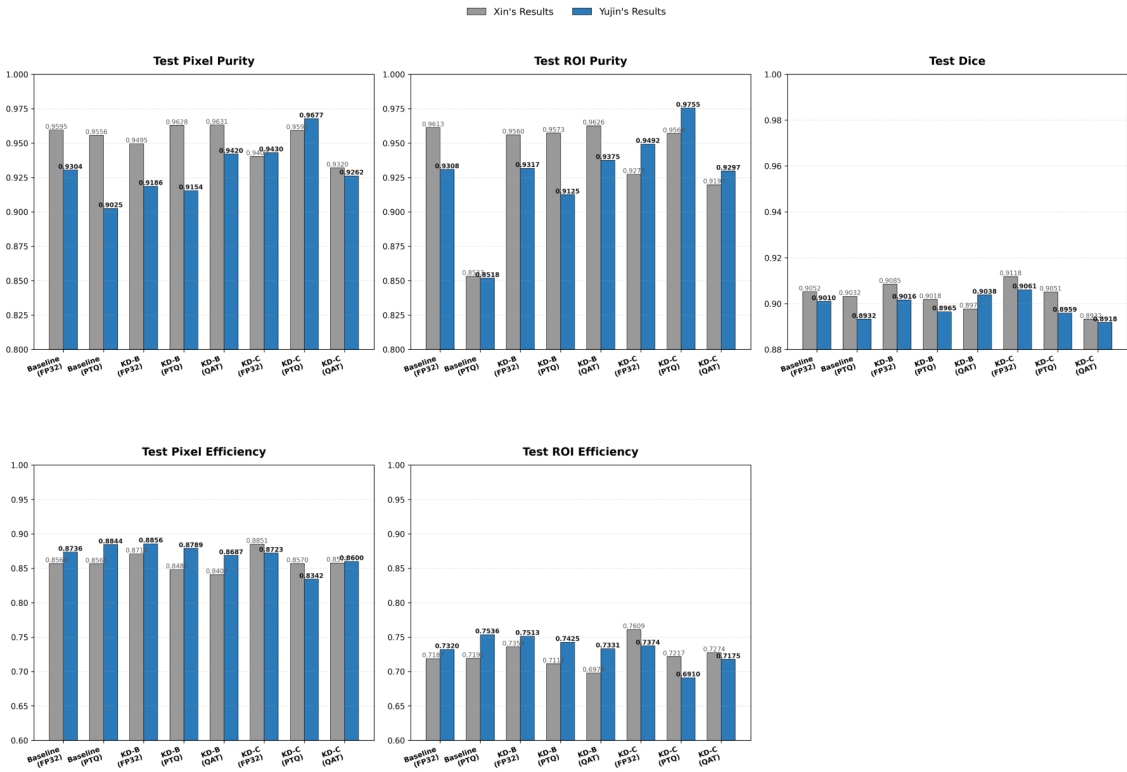
Wire-Cell DNN Signal Processing

PDHD Quantization Validation

		Xin's Results		Yujin's Results		
Category	variant	dice	eff_roi	pur_roi	eff_pix	pur_pix
Baseline Model (small-training)	FP32	0.9052	0.7187	0.9613	0.8569	0.9595
		0.9010	0.7320	0.9308	0.8736	0.9304
	INT8 PTQ	0.9032	0.7191	0.8533	0.8565	0.9556
		0.8932	0.7536	0.8518	0.8844	0.9025
KD (Strategy B) Baseline <- Nested UNet	FP32	0.9085	0.7359	0.9560	0.8710	0.9495
		0.9016	0.7513	0.9317	0.8856	0.9186
	INT8 PTQ	0.9018	0.7112	0.9573	0.8482	0.9628
		0.8965	0.7425	0.9125	0.8789	0.9154
	INT8 QAT	0.8977	0.6978	0.9626	0.8407	0.9631
		0.9038	0.7331	0.9375	0.8687	0.9420
KD (Strategy C) Baseline <- Transformer	FP32	0.9118	0.7609	0.9273	0.8851	0.9403
		0.9061	0.7374	0.9492	0.8723	0.9430
	INT8 PTQ	0.9051	0.7217	0.9569	0.8570	0.9592
		0.8959	0.6910	0.9755	0.8342	0.9677
	INT8 QAT	0.8932	0.7274	0.9197	0.8577	0.9320
		0.8918	0.7175	0.9297	0.8600	0.9262

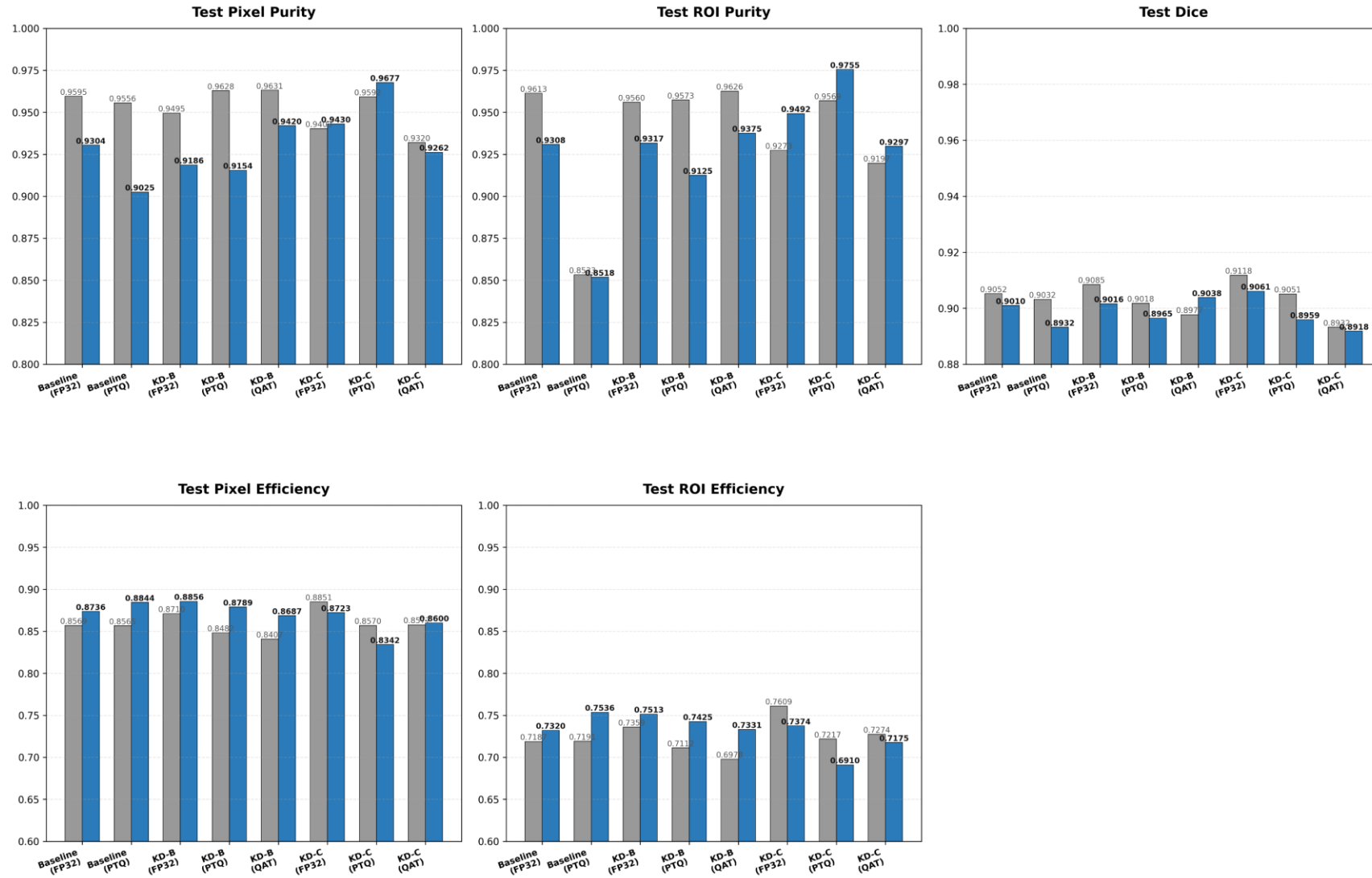
Ref

Metrics (both threshold=0.5)



Backup

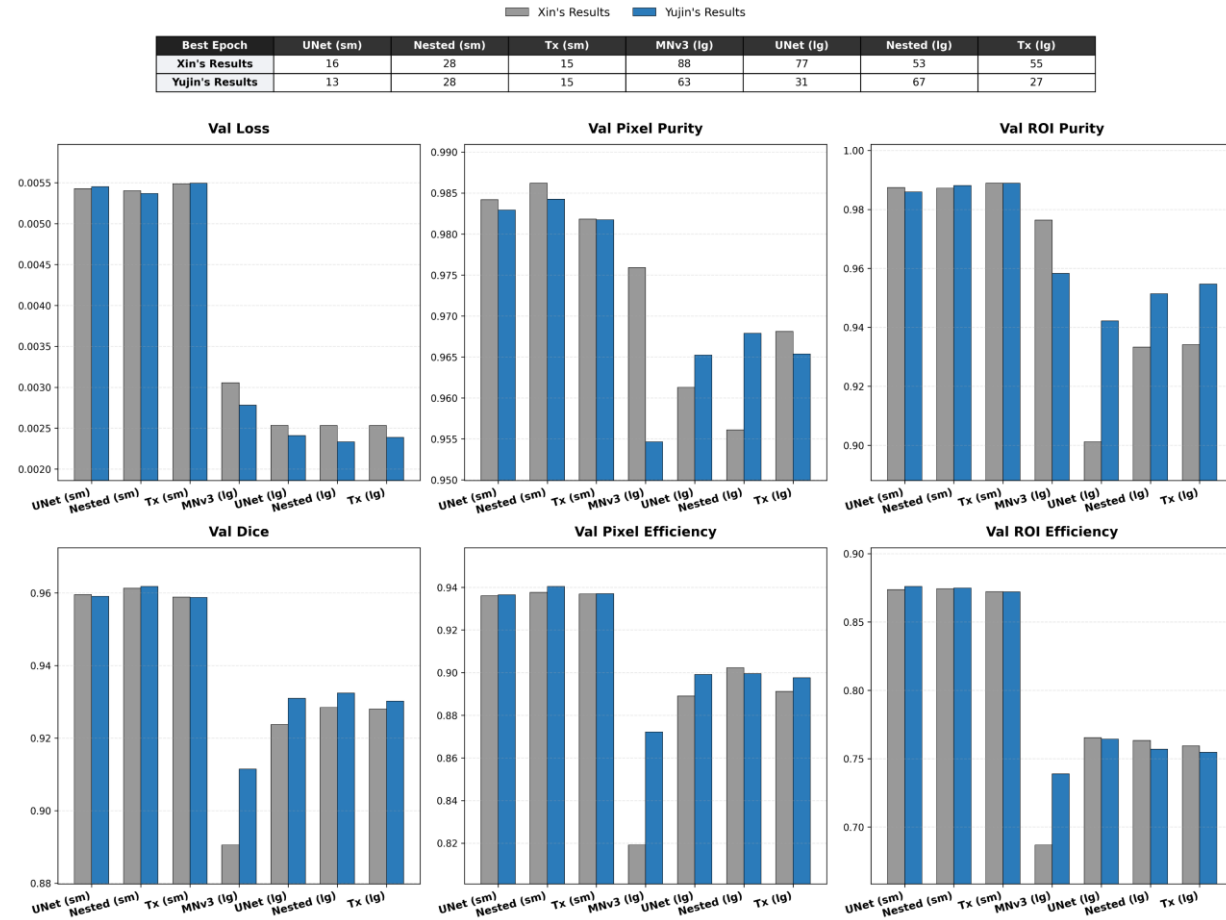
■ Xin's Results ■ Yujin's Results



Wire-Cell DNN Signal Processing

PDHD Teacher Model Validation (Before KD)

Validation Metrics



Comparison (only large dataset results)

Loss/Dice :

- MNv3 and teachers show similar trends. *(consisted)*
- Slightly improved results, specifically for MNv3 Dice in my result.

In Xin's study, MNv3 reached high purities and low efficiencies.

- MNv3 is conservative.
- These weakness should be improved by KD

In my result, MNv3 is less conservative.

- MNv3's efficiencies are improved.
- Purities are worse

Metric	Xin	Yujin (Mine)	Difference (Mine - Xin)
loss	0.003055	0.002780	-0.000275
pur_pix	0.975900	0.954656	-2.1244%
pur_roi	0.976400	0.958333	-1.8067%
dice	0.890500	0.911437	+2.0937%
eff_pix	0.819100	0.872108	+5.3008%
eff_roi	0.686800	0.738917	+5.2117%

Could some randomness affect these difference?

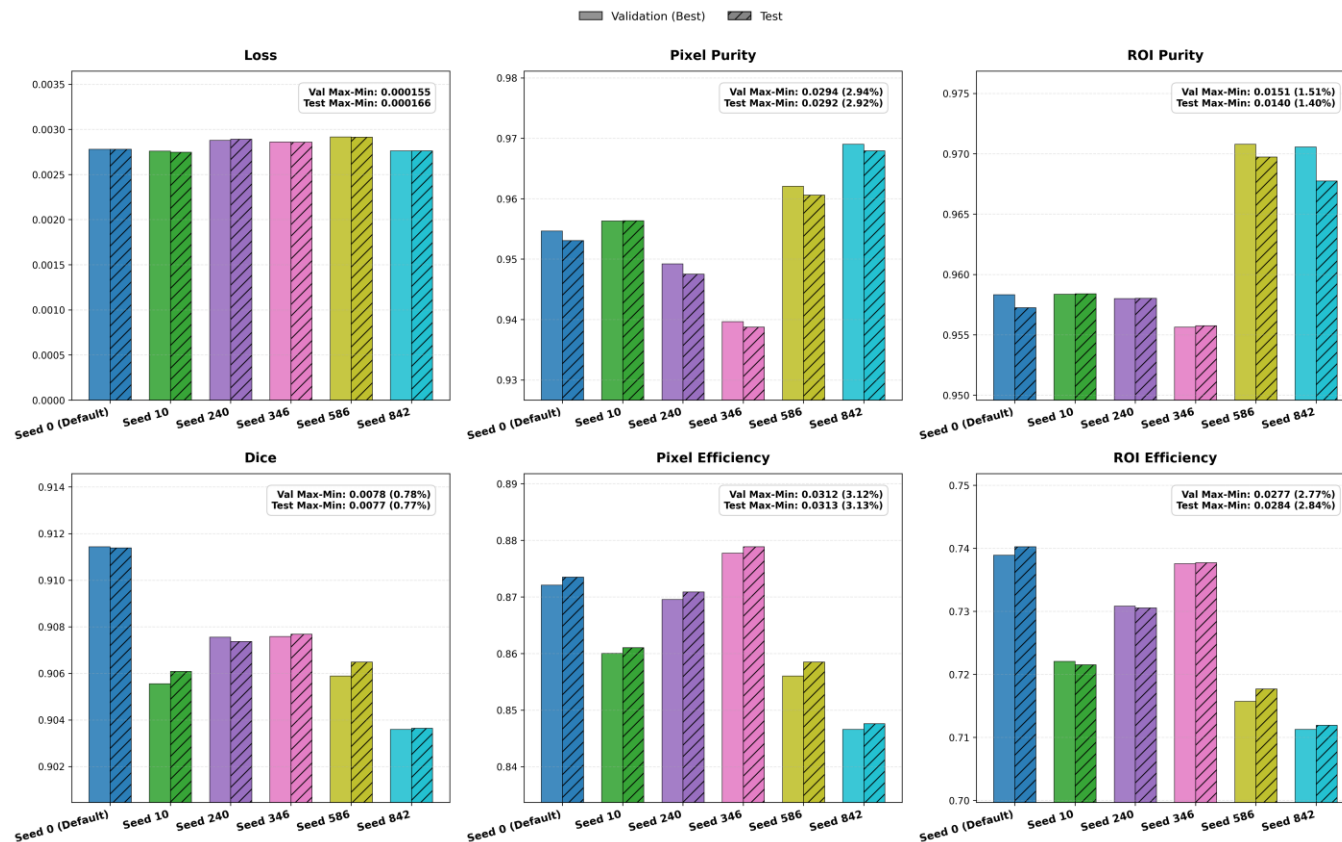
-> Comparing with different random seed(Nest slice)

Wire-Cell DNN Signal Processing

PDHD Teacher Model Validation (Before KD)

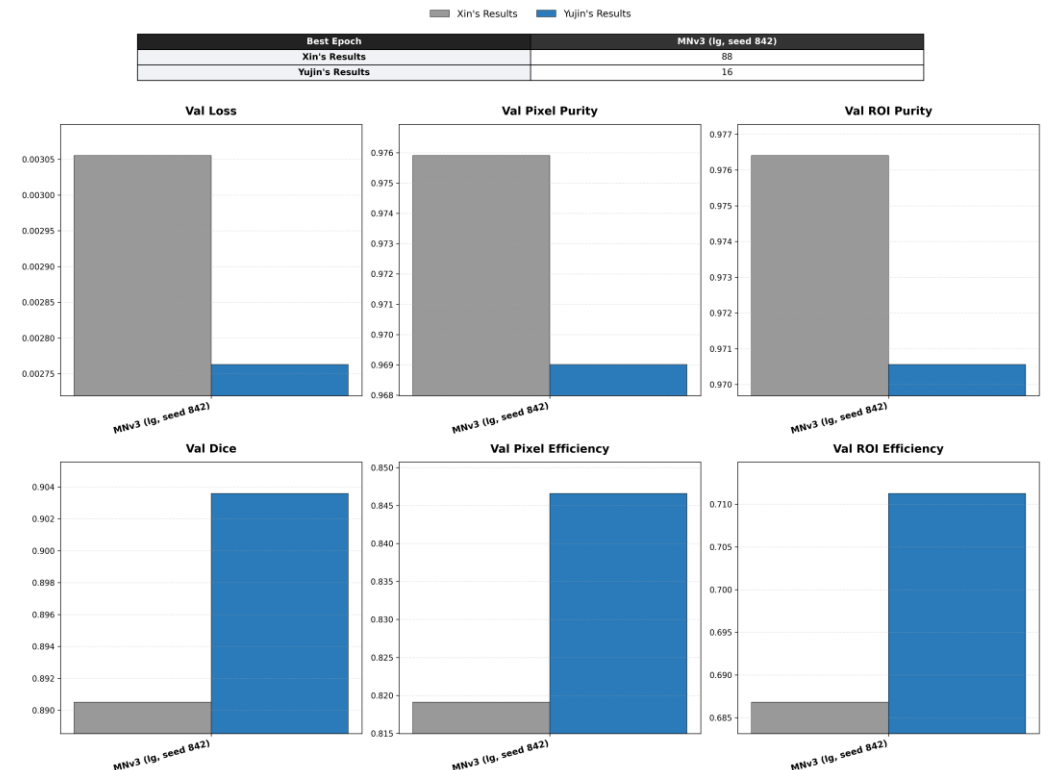
- The default seed is `0`
- Choosing arbitrary 5 seeds

	Seed 0 (Default)	Seed 10	Seed 240	Seed 346	Seed 586	Seed 842
Best Val Epoch	63	18	92	71	77	16



Different seed give slightly different trends.

- About 1~3% differences
- Seed 586, 842 are more conservative.
- Still, less conservative than Xin's.



Wire-Cell DNN Signal Processing

PDHD 3 channels vs 6 channels Validation

https://github.com/lastgeorge/DNN_ROI_SP/blob/main/docs/training_6ch_vs_3ch.md

- **PDHD Raw Dataset:** /nfs/data/1/hnam/train_data_PDHD_fixedbug_separateWC
- No dead-channel Augmentation

Data Pre-processing (`scripts/build\_training\_dataset\_direct.py`)

run id	data file	input ch	truth_th	GPUs
old3ch_th100_ep90	pdhd_rebin4_th100.h5	3	100	1
new6ch_slice012_th150_ep90	pdhd_direct_6ch_rebin4_th150_pad1.h5 *	3	150	1
new6ch_all_th150_ep90_ddp2	pdhd_direct_6ch_rebin4_th150_pad1.h5	6	150	2 (DDP)

- Xin's processed datasets:
 - /nfs/data/1/xqian/toolkit-dev/DNN_ROI_SP/data/pdhd/processed/pdhd_rebin4_th100.h5
 - /nfs/data/1/xqian/toolkit-dev/DNN_ROI_SP/data/pdhd/pdhd_direct_6ch_rebin4_th150_pad1.h5
- Yujin's processed datasets:
 - /nfs/data/1/yujin/DNN_ROI_SP/data/pdhd/processed/pdhd_rebin4_th100.h5
 - /nfs/data/1/yujin/DNN_ROI_SP/data/pdhd/processed/pdhd_direct_6ch_rebin4_th150_pad1.h5

Training (current setting in the draft)

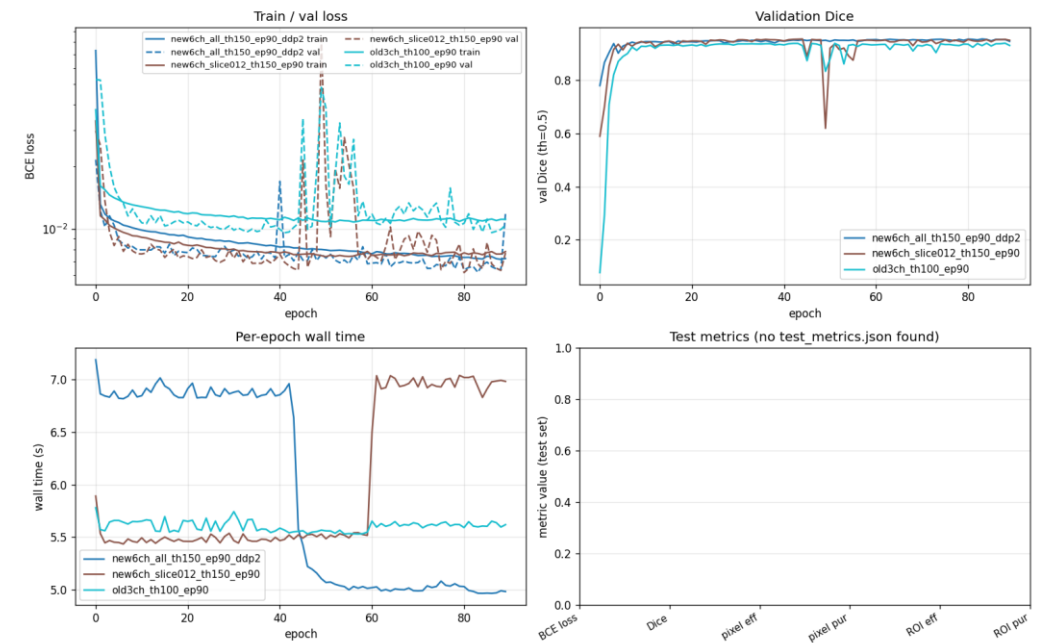
- model: MobileUNetV3
- 90 epochs, seed=0, SGD lr=0.1, momentum=0.9, weight_decay=5e-4, batch_size=1
- 100 events (90 train + 10 val)

Wire-Cell DNN Signal Processing

PDHD 3 channels vs 6 channels validation

- Xin's result

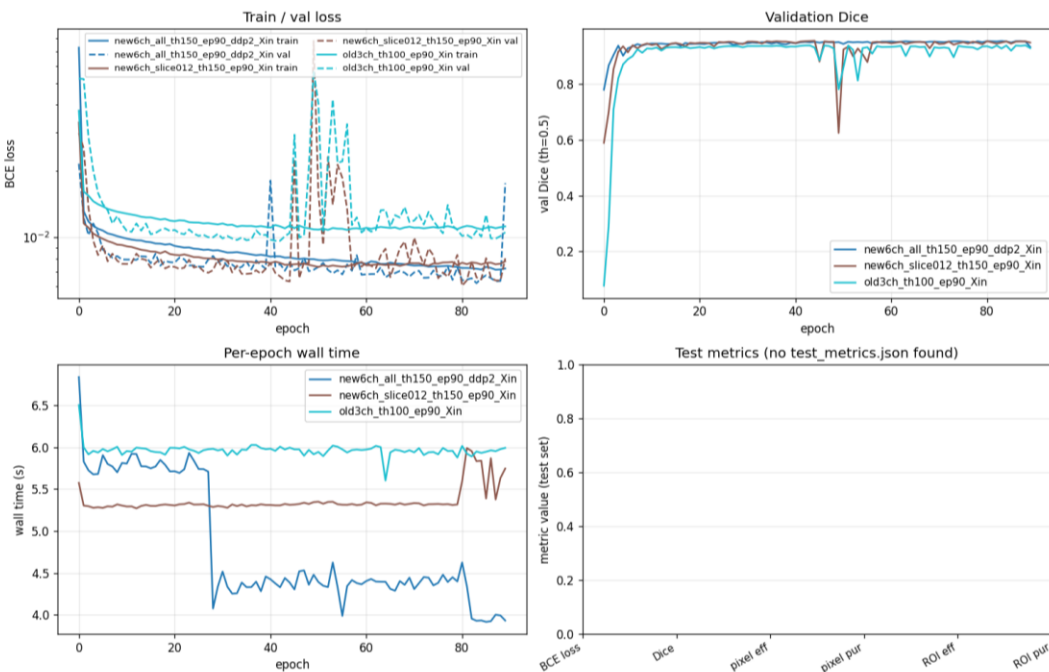
Run comparison — six_channel_three_way



metric	old3ch_th100	new6ch_slice012 (Δ vs old3ch)	new6ch_all_ddp2 (Δ vs slice012)
best epoch	41	80	83
val BCE loss	0.00961	0.00617 (-36 %)	0.00619 ($\approx$)
val Dice	0.9358	0.9519 (+0.016)	0.9536 (+0.002)
pixel efficiency	0.9037	0.9227 (+0.019)	0.9269 (+0.004)
pixel purity	0.9705	0.9832 (+0.013)	0.9821 ($\approx$)
ROI efficiency	0.8158	0.8631 (+0.047)	0.8662 (+0.003)
ROI purity	0.9899	0.9922 (+0.002)	0.9909 ($\approx$)

Conclusion: Consistent Results!

- My result with Xin's datasets



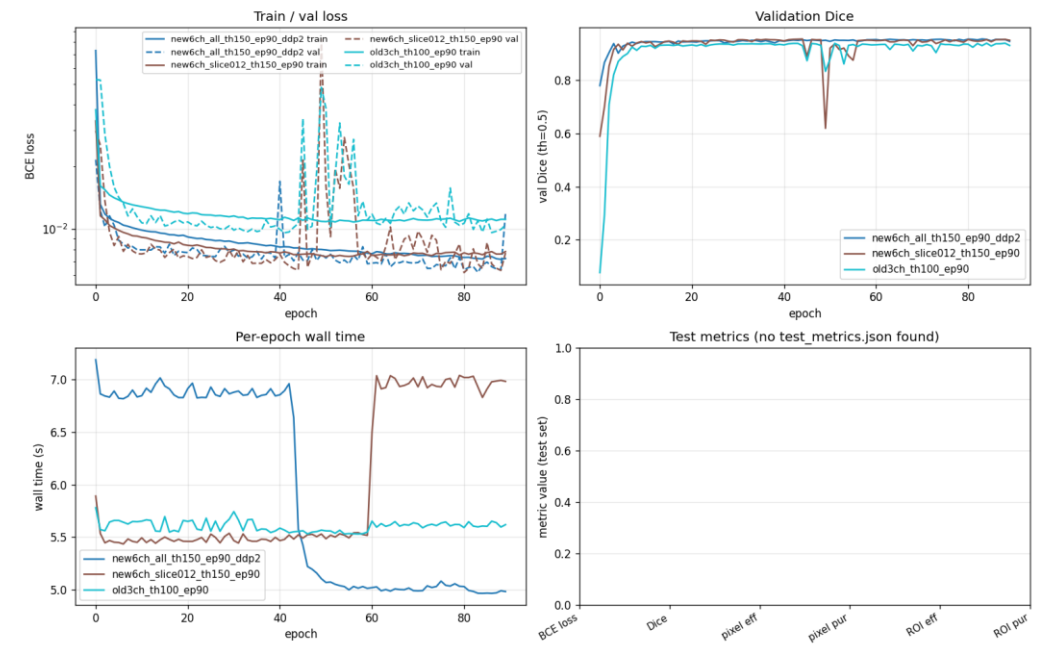
Metric	old3ch_th100	new6ch_slice	new6ch_all
best epoch	41	80	83
val BCE loss	0.009673	0.006052 (-37.4%)	0.006215 (+2.7%)
Val Dice	0.935769	0.952568 (+0.0168)	0.952854 (+0.0003)
pixel efficiency	0.904812	0.924560 (+0.0197)	0.924105 (-0.0005)
pixel purity	0.969148	0.982453 (+0.0133)	0.983640 (+0.0112)
ROI efficiency	0.816906	0.866782 (+0.0499)	0.864017 (-0.0028)
Roi purity	0.988843	0.989778 (+0.0009)	0.992107 (+0.0023)

Wire-Cell DNN Signal Processing

PDHD 3 channels vs 6 channels validation

- Xin's result

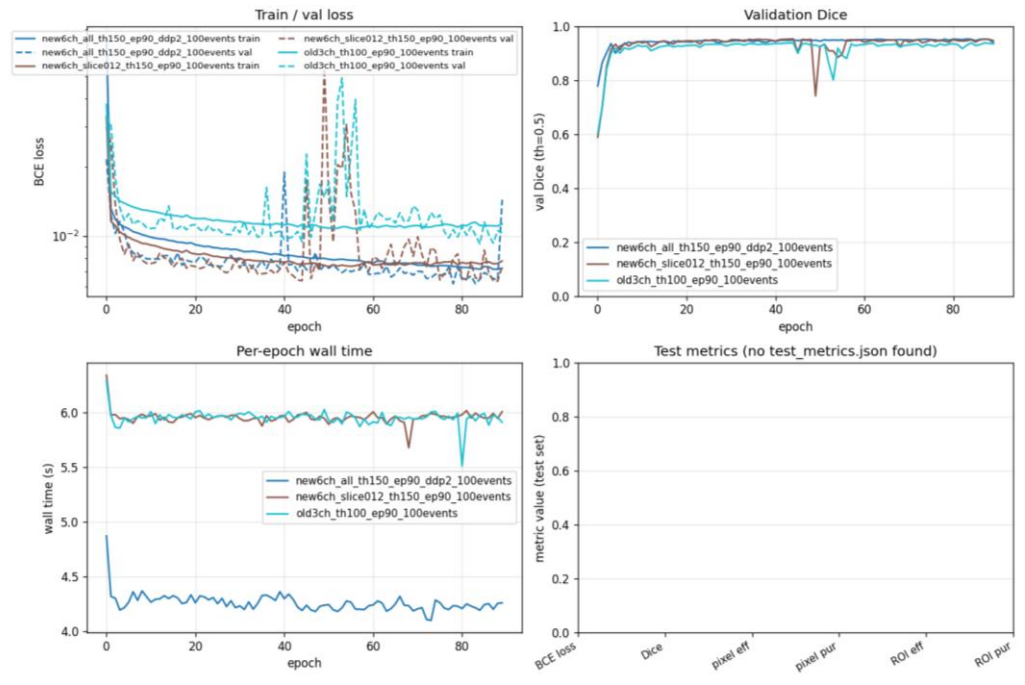
Run comparison — six_channel_three_way



metric	old3ch_th100	new6ch_slice012 (Δ vs old3ch)	new6ch_all_ddp2 (Δ vs slice012)
best epoch	41	80	83
val BCE loss	0.00961	0.00617 (-36 %)	0.00619 ($\approx$)
val Dice	0.9358	0.9519 (+0.016)	0.9536 (+0.002)
pixel efficiency	0.9037	0.9227 (+0.019)	0.9269 (+0.004)
pixel purity	0.9705	0.9832 (+0.013)	0.9821 ($\approx$)
ROI efficiency	0.8158	0.8631 (+0.047)	0.8662 (+0.003)
ROI purity	0.9899	0.9922 (+0.002)	0.9909 ($\approx$)

Conclusion: Consistent Results!

- My result with Yujin's datasets



Metric	old3ch_th100	new6ch_slice	new6ch_all
best epoch	84	88	78
val BCE loss	0.009248	0.006358 (-31.3%)	0.006190 (-2.6%)
Val Dice	0.938748	0.951877 (+0.0131)	0.952780 (+0.0009)
pixel efficiency	0.901140	0.930523 (+0.0294)	0.930871 (+0.0003)
pixel purity	0.979935	0.974416 (-0.0055)	0.975908 (+0.0149)
ROI efficiency	0.820036	0.860076 (+0.0400)	0.865425 (+0.0053)
Roi purity	0.990242	0.994176 (+0.0039)	0.991427 (-0.0027)

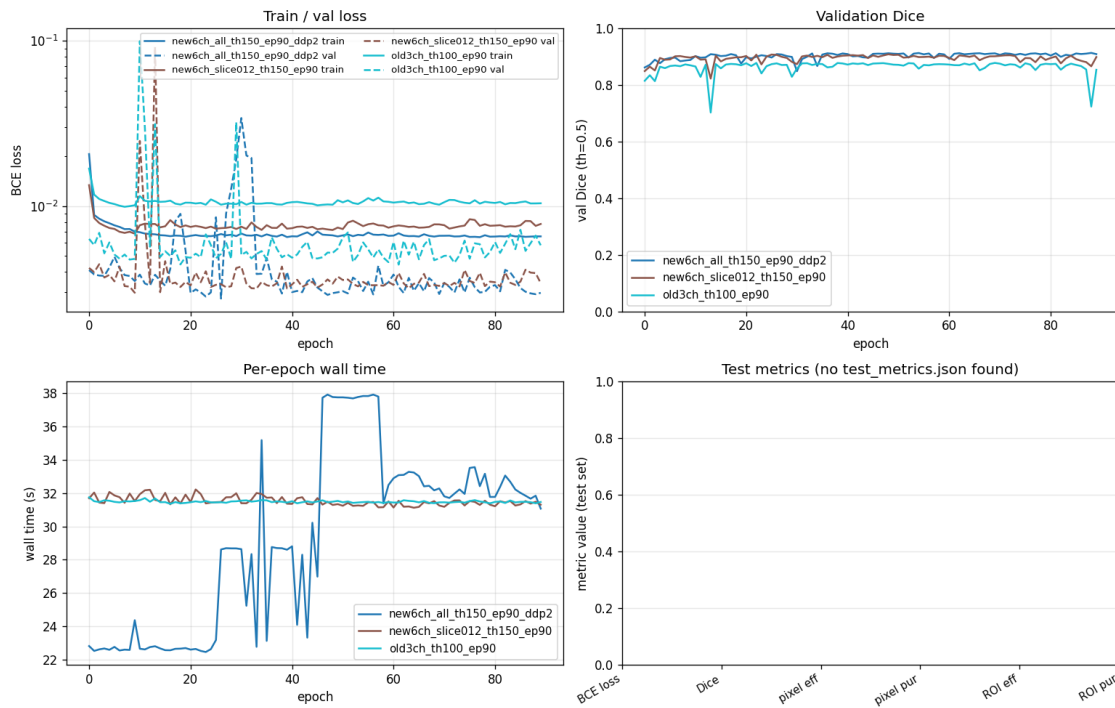
Wire-Cell DNN Signal Processing

PDHD 3 channels vs 6 channels validation

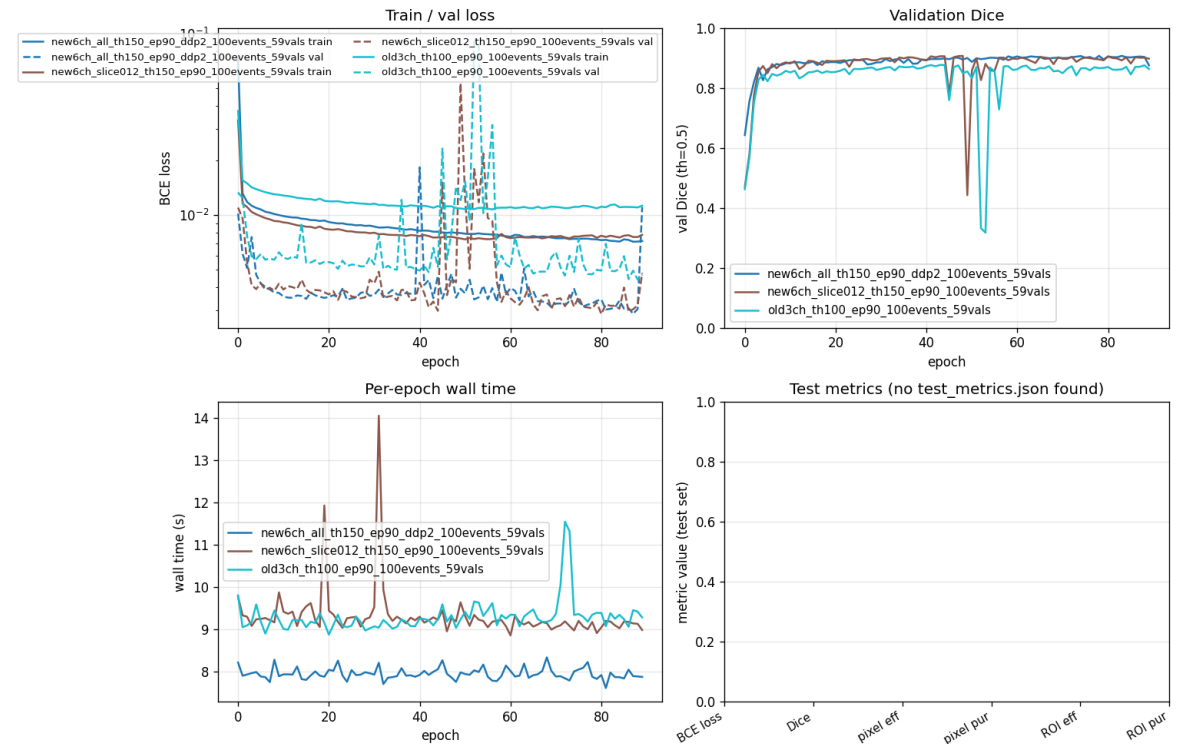
Test with 590events

- 531 train events and 59 validation events in Yujin's dataset.
- To investigate the impact of training sample size, models trained on 531 and 90 events respectively were evaluated against the same 53 validation samples

• 531 events



• 90 events



Wire-Cell DNN Signal Processing

3 channels vs 6 channels validation

- Xin's result(90train events, 10 val events)

metric	old3ch_th100	new6ch_slice012 (Δ vs old3ch)	new6ch_all_ddp2 (Δ vs slice012)
best epoch	41	80	83
val BCE loss	0.00961	0.00617 (-36 %)	0.00619 ($\approx$)
val Dice	0.9358	0.9519 (+0.016)	0.9536 (+0.002)
pixel efficiency	0.9037	0.9227 (+0.019)	0.9269 (+0.004)
pixel purity	0.9705	0.9832 (+0.013)	0.9821 ($\approx$)
ROI efficiency	0.8158	0.8631 (+0.047)	0.8662 (+0.003)
ROI purity	0.9899	0.9922 (+0.002)	0.9909 ($\approx$)

- The Model becomes more generalized.(from val loss, Dice)
- Efficiencies/Purities are become slightly worse.
- 531 events and 90 events give similar numbers.
- Size of training sample doesn't effect much
- In Xin's results, it looks like to give more positive results due to some bias in validation samples.

- 531 events

Metric	old3ch	new6ch_slice	new6ch_all
best epoch	61	9	26
val BCE loss	0.004418	0.002984 (-32.45%)	0.002752 (-37.71%)
Val Dice	0.869741	0.896515 (+0.0268)	0.908686 (+0.0389)
pixel efficiency	0.797166	0.829350 (+0.0322)	0.871504 (+0.0743)
pixel purity	0.957410	0.975883 (+0.0185)	0.949446 (-0.0080)
ROI efficiency	0.634105	0.688518 (+0.0544)	0.733924 (+0.0998)
Roi purity	0.956780	0.977191 (+0.0204)	0.950377 (-0.0064)

- 90 events

Metric	old3ch	new6ch_slice	new6ch_all
best epoch	72	86	87
val BCE loss	0.004648	0.002945 (-36.64%)	0.002860 (-38.47%)
Val Dice	0.872986	0.902867 (+0.0299)	0.905214 (+0.0322)
pixel efficiency	0.808594	0.848910 (+0.0403)	0.862133 (+0.0535)
pixel purity	0.949106	0.964508 (+0.0154)	0.953117 (+0.0040)
ROI efficiency	0.632260	0.708018 (+0.0758)	0.722403 (+0.0901)
Roi purity	0.958279	0.968210 (+0.0099)	0.955497 (-0.0028)

Wire-Cell DNN Signal Processing

3 channels vs 6 channels validation

Conclusion

- The reproduced results are consistent with Xin's results
- 590events rather than 100events gives more generalized results.
- All results show the improvement by using 6channels.

Wire-Cell DNN Signal Processing

PDHD Teacher Model Validation (Before KD)

https://github.com/lastgeorge/DNN_ROI_SP/blob/main/docs/full_model_training.md

https://github.com/lastgeorge/DNN_ROI_SP/blob/main/docs/full_model_production.md

https://github.com/lastgeorge/DNN_ROI_SP/blob/main/docs/full_model_summary.md

Processed Dataset:

- `/nfs/data/1/yujin/DNN_ROI_SP/data/pdhd/processed/pdhd_direct_6ch_rebin4_th150_pad1.h5`
- 6 channel dataset generated by myself
- Small dataset: 90(train) + 10(val)
- Large dataset: 470(train) + 60(val)

Models

run id	model	params	events (train+val)	GPUs	hardware
<code>full_unet_6ch_th150_ep90_l40s_ddp2</code>	unet	13.4 M	90 + 10	2	L40S
<code>full_nestedunet_6ch_th150_ep90_l40s_ddp2</code>	nestedunet	9.2 M	90 + 10	2	L40S
<code>full_transformer_6ch_th150_ep90_l40s_ddp2</code>	transformer	26.0 M	90 + 10	2	L40S
<code>prod_unet_6ch_th150_ep100_l40s_ddp6</code>	unet	13.4 M	470 + 60	6	L40S
<code>prod_nestedunet_6ch_th150_ep100_l40s_ddp6</code>	nestedunet	9.2 M	470 + 60	6	L40S
<code>prod_transformer_6ch_th150_ep100_l40s_ddp6</code>	transformer	26.0 M	470 + 60	6	L40S
<code>prod_mobilenetv3_6ch_th150_ep100_l40s_ddp8</code>	mobilenetv3-L	5.2 M	470 + 60	8	L40S

Instead sdcc, currently ran in wcgpu1 with 2 GPUs

Other hyperparameters

- seed=0, SGD lr=0.1, momentum=0.9, weight_decay=5e-4, batch_size=1
- 90 epochs for small dataset
- 100 epochs for large dataset

Wire-Cell DNN Signal Processing

PDHD Teacher Model Validation (Before KD)

- Xin's result

Best epoch (lowest val_loss)							
metric	UNet (sm)	Nested (sm)	Tx (sm)	UNet (lg)	Nested (lg)	Tx (lg)	MNv3 (lg)
best epoch	16	28	15	77	53	55	88
val BCE loss	0.005426	0.005401	0.005487	0.002535	0.002533	0.002533	0.003055
val Dice	0.9595	0.9613	0.9588	0.9237	0.9284	0.9280	0.8905
pixel eff	0.9360	0.9377	0.9369	0.8890	0.9023	0.8912	0.8191
pixel pur	0.9842	0.9862	0.9818	0.9613	0.9561	0.9681	0.9759
ROI eff	0.8737	0.8744	0.8721	0.7654	0.7634	0.7594	0.6868
ROI pur	0.9874	0.9872	0.9889	0.9012	0.9333	0.9341	0.9764

- My result(large dataset)

metric	UNet (lg)	Nested (lg)	Tx (lg)	MobileNetV3 (lg)
best epoch	31	67	27	63
val BCE loss	0.002409	0.002334	0.002386	0.002780
val Dice	0.930937	0.932417	0.930217	0.911437
pixel eff	0.899062	0.899529	0.897616	0.872108
pixel pur	0.965238	0.967886	0.965370	0.954656
ROI eff	0.764363	0.757057	0.754574	0.738917
ROI pur	0.942198	0.951363	0.954670	0.958333