AlignShift: Bridging the Gap of Imaging Thickness in 3D Anisotropic Volumes

Jiancheng Yang*, Yi He*, Xiaoyang Huang, Jingwei Xu, Xiaodan Ye, Guangyu Tao, Bingbing Ni.

Oct 2020

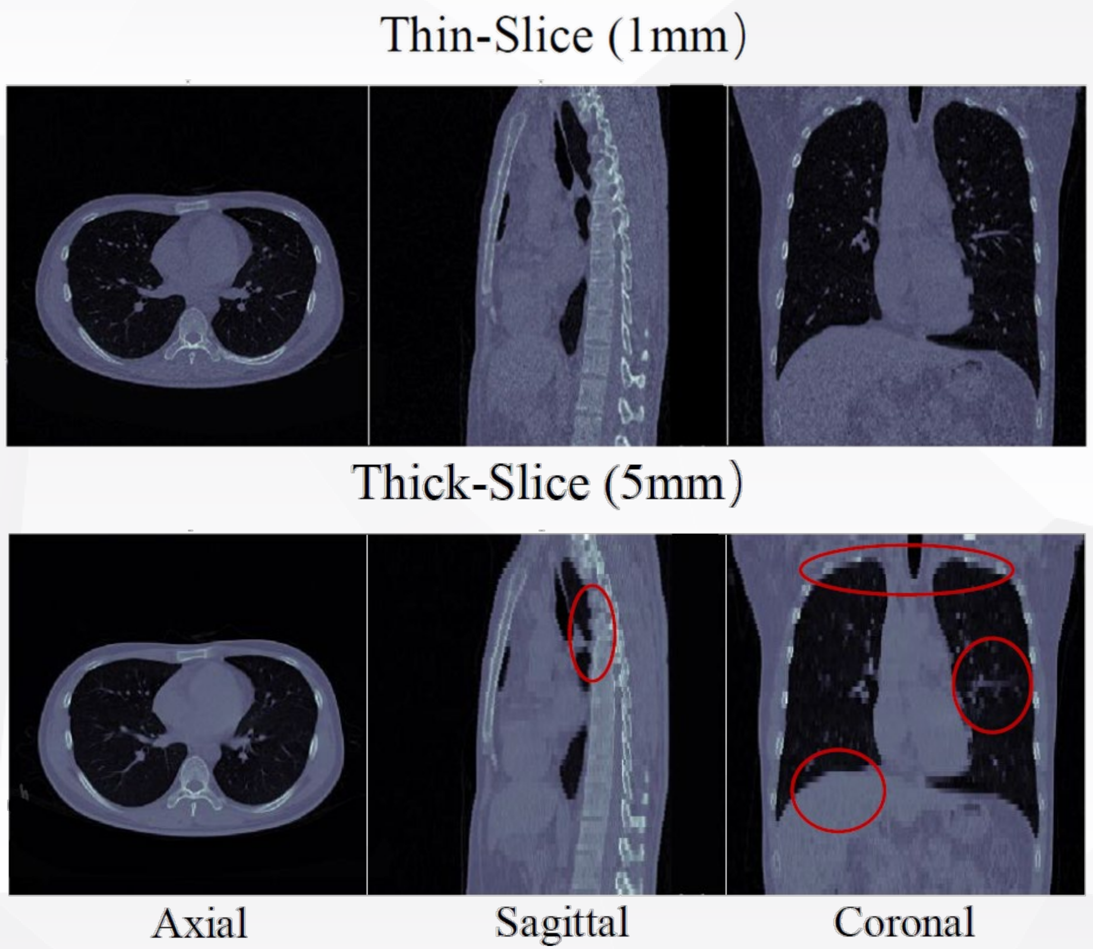
MICCAI 2020 Lima PERU

上海交通大學 Shanghai Jiao Tong University | 人工智能研究院 Artificial Intelligence Institute

点内 DIANEI

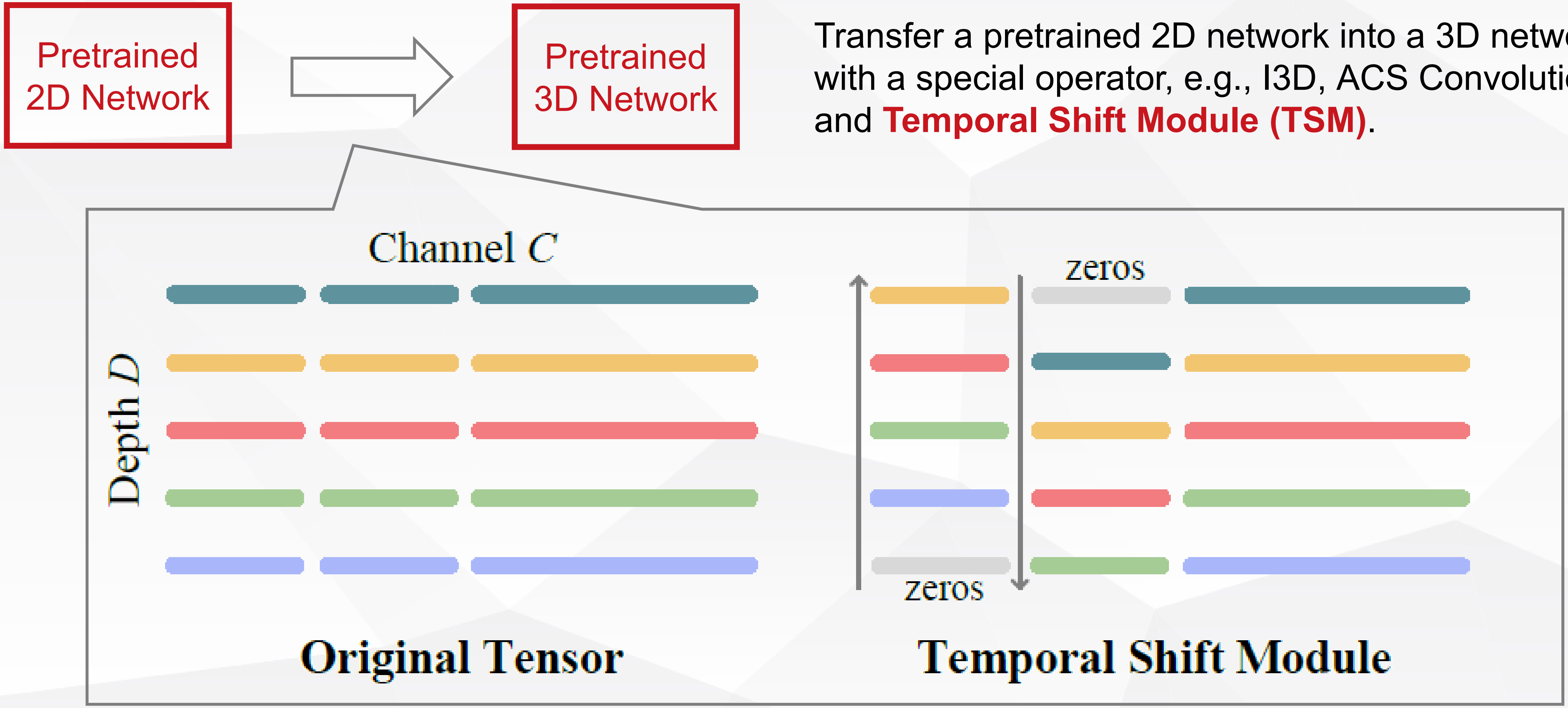
HUAWEI

Introduction



Is there a **unified** approach to bridge the **performance gap** between thin- and thick-slice 3D **anisotropic** volumes?

Preliminaries



Transfer a pretrained 2D network into a 3D network with a special operator, e.g., I3D, ACS Convolution, and **Temporal Shift Module (TSM)**.

TSM: Temporal Shift Module for Efficient Video Understanding. Ji Lin et al. ICCV'19.
Reinventing 2D Convolutions for 3D Images. Jiancheng Yang et al. arXiv,1911.

Methodology: *AlignShift*

Pretrained 2D Network → Pretrained 3D Network

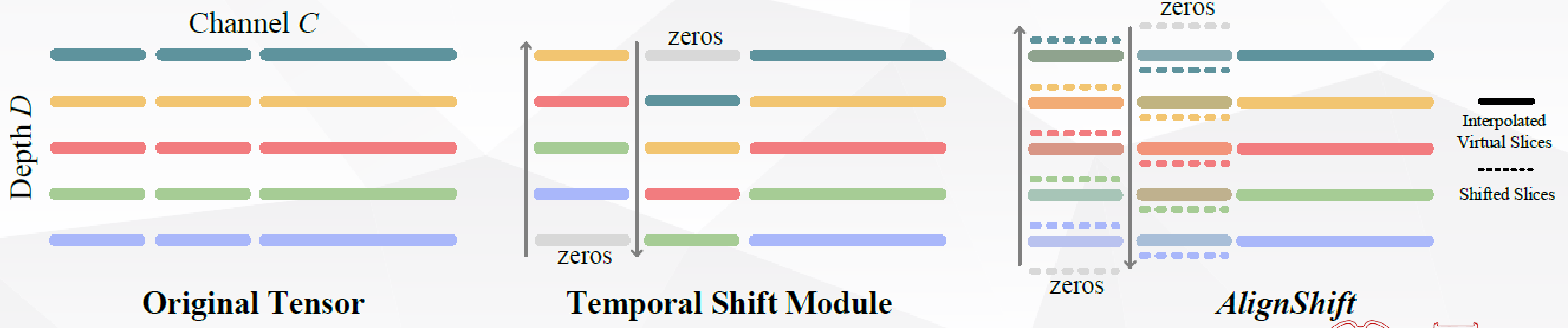
Transfer a pretrained 2D network into a **thickness-aware** 3D network via ***AlignShift***.

TSM

- Inputs are **thickness-normalized** via spatial normalization.

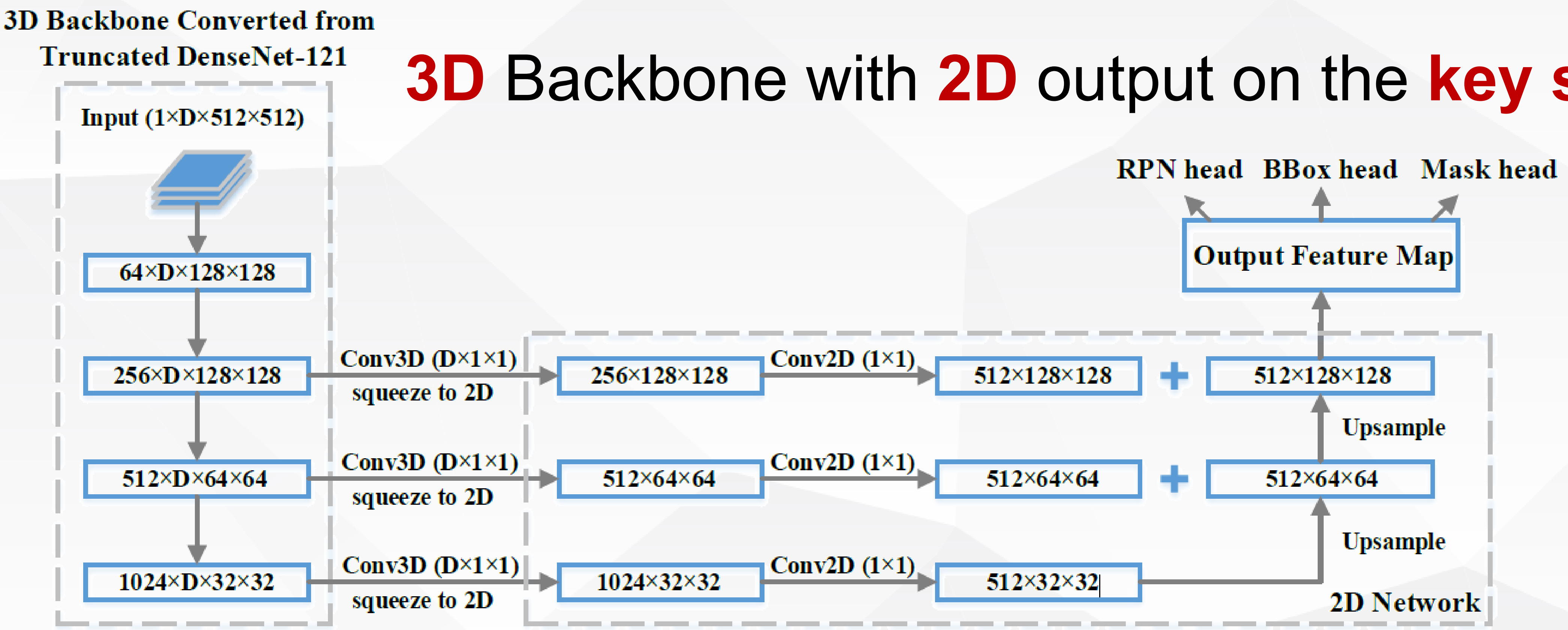
AlignShift

- Inputs are not thickness-normalized.
- The converted networks are thickness-aware, i.e., the thickness is used as input.



MICCAI 2020 Lima PERU

3D Backbone with 2D output on the key slice



2D Backbone	Conv2D $K \times K$	Pool2D $K \times K$	Norm2D
3D Backbone	$(\mathit{AlignShift}+)$Conv3D $1 \times K \times K$	Pool3D $1 \times K \times K$	Norm3D

Experiments: DeepLesion Benchmark

MICCAI2020 Lima PERU

3D context modeling itself is very effective given multiple slices.

Methods	Slices	0.5	1	2	4	8	16	Avg.[0.5,1,2,4]
3DCE [26] MICCAI'18	×27	62.48	73.37	80.70	85.65	89.09	91.06	75.55
ULDor [23] ISBI'19	×1	52.86	64.8	74.84	84.38	87.17	91.8	69.22
V.Attn [24] MICCAI'19	×3	69.10	77.90	83.80	-	-	-	-
Retina. [35] MICCAI'19	×3	72.15	80.07	86.40	90.77	94.09	96.32	82.35
MVP [13] MICCAI'19	×3	70.01	78.77	84.71	89.03	-	-	80.63
MVP [13] MICCAI'19	×9	73.83	81.82	87.60	91.30	-	-	83.64
MULAN [27] MICCAI'19	×9	76.12	83.69	88.76	92.30	94.71	95.64	$\boxed{85.22}$
w/o SRL [27] MICCAI'19	×9	-	-	-	-	-	-	$\boxed{84.22}$
Ours 2.5D	×3	71.27	79.82	86.30	90.61	93.75	95.70	82.00
Ours 2.5D	×7	72.66	81.45	87.07	90.98	93.40	95.30	83.04
Ours TSM	×3	70.24	79.52	86.28	90.90	94.06	96.09	81.73
Ours TSM	×7	75.98	83.65	88.44	92.14	94.89	96.50	$\boxed{85.05}$
Ours *AlignShift*	×3	72.90	80.74	87.15	91.92	94.85	96.48	83.18
Ours *AlignShift*	×7	**79.40**	**85.50**	**90.09**	**93.26**	**95.24**	**96.66**	**87.06**

previous state of the art

SJTU

Experiments: DeepLesion Benchmark

MICCAI2020 Lima PERU

AlignShift improves the performance further, which significantly outperforms previous SOTA.

Methods	Slices	0.5	1	2	4	8	16	Avg.[0.5,1,2,4]
3DCE [26] MICCAI'18	×27	62.48	73.37	80.70	85.65	89.09	91.06	75.55
ULDor [23] ISBI'19	×1	52.86	64.8	74.84	84.38	87.17	91.8	69.22
V.Attn [24] MICCAI'19	×3	69.10	77.90	83.80	-	-	-	-
Retina. [35] MICCAI'19	×3	72.15	80.07	86.40	90.77	94.09	96.32	82.35
MVP [13] MICCAI'19	×3	70.01	78.77	84.71	89.03	-	-	80.63
MVP [13] MICCAI'19	×9	73.83	81.82	87.60	91.30	-	-	83.64
MULAN [27] MICCAI'19	×9	76.12	83.69	88.76	92.30	94.71	95.64	85.22
w/o SRL [27] MICCAI'19	×9	-	-	-	-	-	-	84.22
Ours 2.5D	×3	71.27	79.82	86.30	90.61	93.75	95.70	82.00
Ours 2.5D	×7	72.66	81.45	87.07	90.98	93.40	95.30	83.04
Ours TSM	×3	70.24	79.52	86.28	90.90	94.06	96.09	81.73
Ours TSM	×7	75.98	83.65	88.44	92.14	94.89	96.50	85.05
Ours *AlignShift*	×3	72.90	80.74	87.15	91.92	94.85	96.48	83.18
Ours *AlignShift*	×7	**79.40**	**85.50**	**90.09**	**93.26**	**95.24**	**96.66**	**87.06**

previous state of the art

Experiments: Analysis on Thickness

AlignShift bridges the performance gap between thin- and thick-slice volumes.

Methods	Thinkness	0.5	1	2	4	8	16	Avg.[0.5,1,2,4]	diff.
2.5D $\times3$	All	71.27	79.82	86.30	90.61	93.75	95.70	82.00	-
	Thin	72.78	80.65	87.21	90.94	93.97	95.86	82.89	+0.89
	Thick	69.88	79.16	85.51	90.48	93.65	95.65	81.26	−0.74
2.5D $\times7$	All	72.66	81.45	87.07	90.98	93.40	95.30	83.04	-
	Thin	75.77	83.93	88.85	92.37	94.26	95.78	85.23	+2.19
	Thick	69.76	78.96	85.75	90.03	92.67	94.99	81.13	−1.91
TSM $\times3$	All	70.24	79.52	86.28	90.90	94.06	96.09	81.73	-
	Thin	73.74	81.84	87.54	92.21	94.92	96.72	83.83	+2.10
	Thick	67.03	77.37	84.98	89.95	93.16	95.44	79.83	−1.90
TSM $\times7$	All	75.98	83.65	88.44	92.14	94.89	96.50	85.05	-
	Thin	78.76	85.53	89.67	93.48	95.61	96.68	86.86	+1.81
	Thick	73.26	81.97	87.10	90.96	94.14	96.42	83.32	−1.73
AlignShift $\times3$	All	72.90	80.74	87.15	91.92	94.85	96.48	83.18	-
	Thin	73.51	81.59	87.62	92.37	94.87	96.51	83.77	+0.61
	Thick	72.85	80.18	87.10	91.94	94.91	96.54	83.02	−0.16
AlignShift $\times7$	All	79.40	85.50	90.09	93.26	95.24	96.66	87.06	-
	Thin	80.73	86.43	91.02	93.97	95.78	96.97	88.04	+0.98
	Thick	78.27	84.74	88.89	92.47	94.67	96.38	86.09	−0.97

Convert 2D model into 3D with a **single line of code**.

```
# m is a standard pytorch model
m = torchvision.models.resnet18(True)

m = Converter(m,
              alignshift_conv_cfg,
              additional_forward_fts=['thickness'],
              skip_first_conv=True,
              first_conv_input_channles=1)
# after converted, m is using AlignShiftConv and capable of processing 3D volumes
x = torch.rand(batch_size, in_channels, D, H, W)
thickness = torch.rand(batch_size, 1)
out = m(x, thickness)
```

Reproducible **code** on DeepLesion with trained model **snapshots**
https://github.com/M3DV/AlignShift/

Conclusion

- We **challenge spatial normalization** as a standard pre-processing approach to solve thickness issues.
- We introduce operators to transfer any pretrained 2D network into 3D.
- We further propose a novel operator ***AlignShift*** to convert any 2D pretrained network into **thickness-aware 3D network**, which bridges the gap between thin- and thick-slice data.
- Without whistles and bells, we establish a new state of the art on DeepLesion, which surpasses prior arts by considerable margins.

Thanks for Listening

Check out my home page for materials on this study

https://jiancheng-yang.com/

WeChat

jekyll4168