Incremental Cross-view Mutual Distillation for Self-supervised Medical CT Synthesis

Chaowei Fang[1] Liang Wang[1] Dingwen Zhang[2] Jun Xu[3] Yixuan Yuan[4] Junwei Han[2]
[1]Xidian University [2]Northwestern Polytechnical University [3]Nankai University [4]City University of Hong Kong

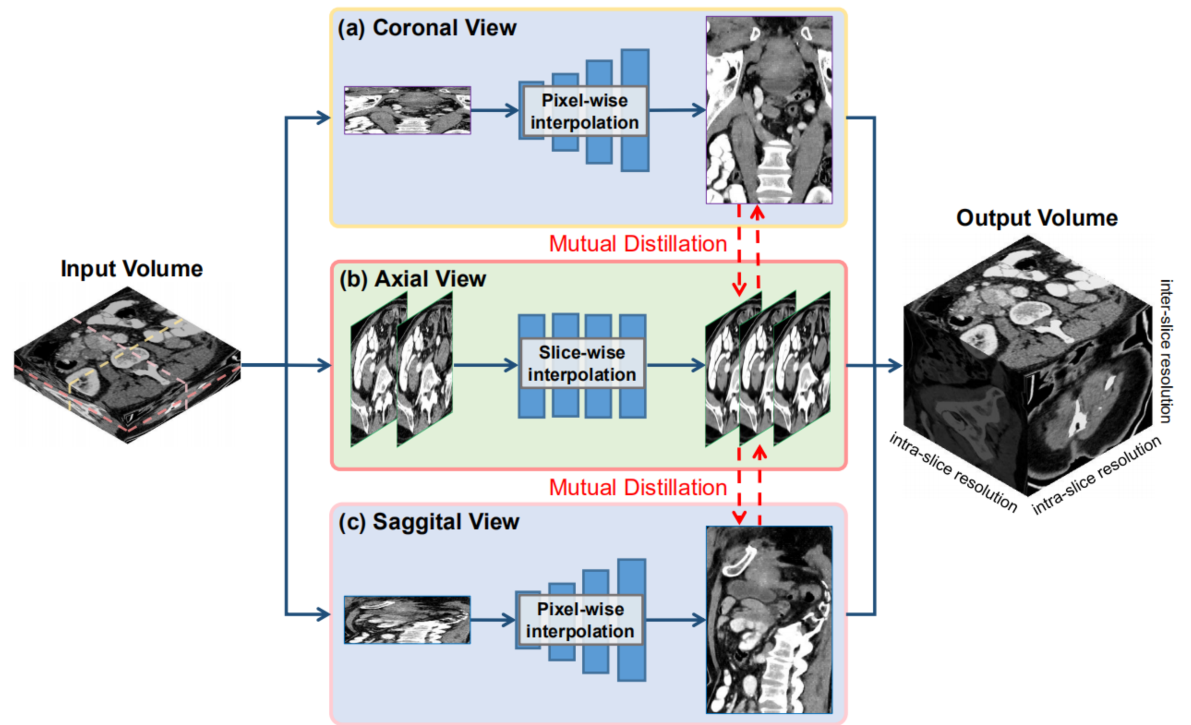
CVPR JUNE 19-24 2022 NEW ORLEANS • LOUISIANA

Background

- Computer tomography scans are usually acquired with low intra-slice resolution and the ground-truth intermediate medical slices are always absent in clinical use.
- Improving the intra-slice resolution is beneficial to the disease diagnosis for both human experts and computer-aided systems.

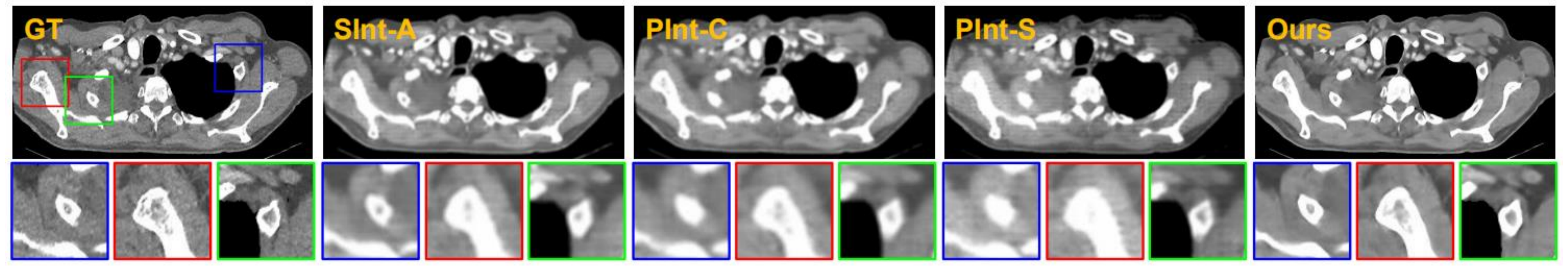
Target

This paper aims at building a medical slice synthesis model to increase the inter-slice resolution of an input 3D volume.



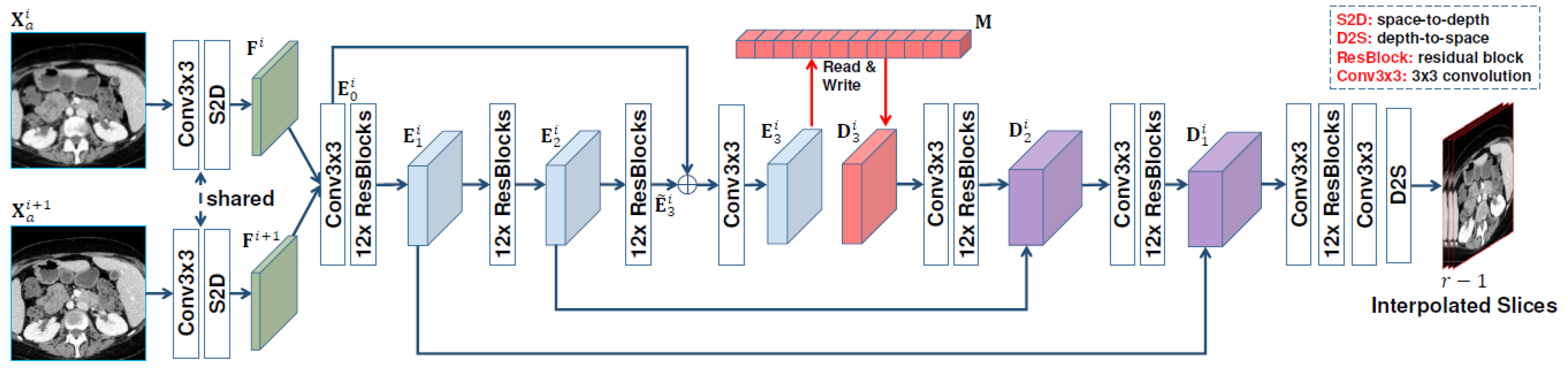
Motivation

- Intermediate slices can be generated by interpolating slices in the axial view or interpolating pixels in the coronal/sagittal view.
- Structural information appears to have different characteristics across views, and models learned from different views have their superiority.

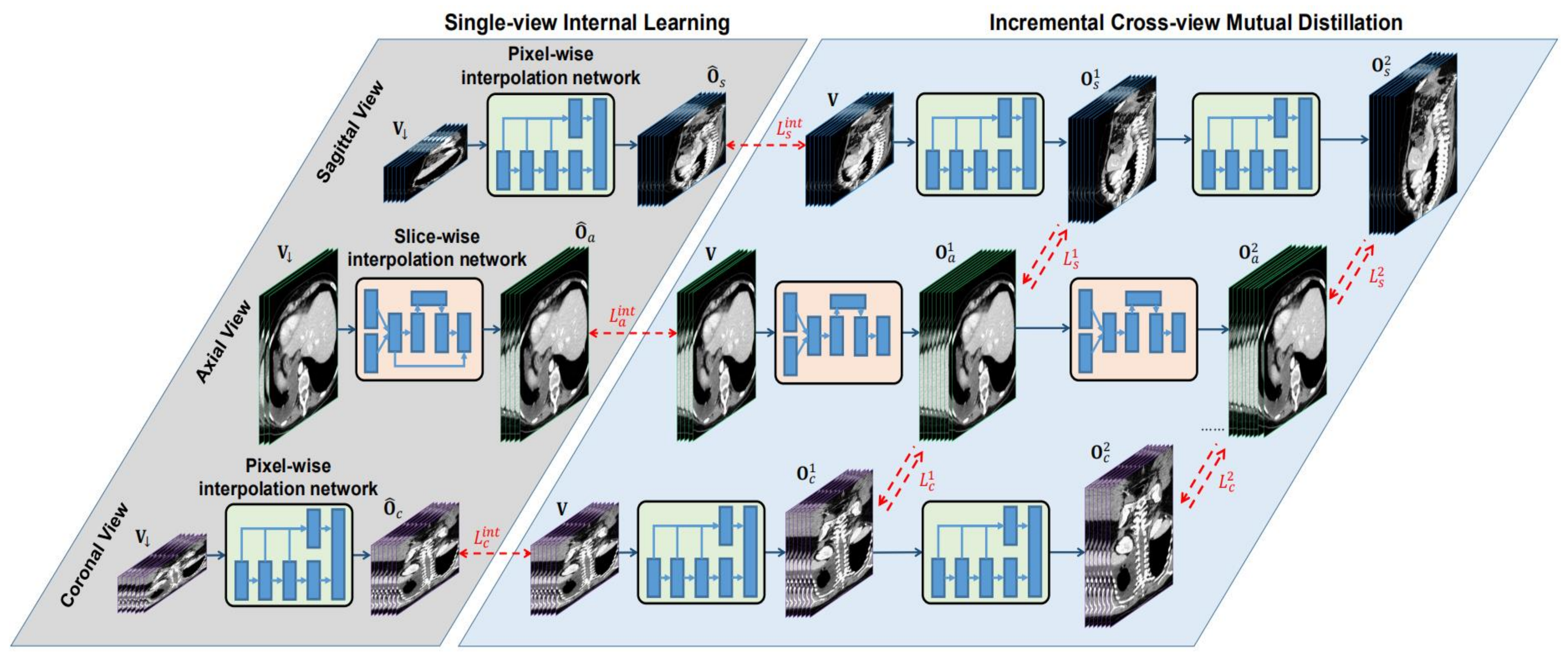


Approach

- Axial-View Slice Interpolation:



- Coronal/Sagittal-View Pixel Interpolation: HAN [*Niu et al.,2020*].
- Training Process:



– Single-view Internal Learning :

- Regarding down-sampled CT and original CT as training samples
- Using mean square error to calculate losses for predicted images and their wavelet coefficients

$$L_a^{int} = \mathrm{MSE}(\hat{\mathbf{O}}_a, \mathbf{V}) + \sum_{t=1}^{3} \mathrm{MSE}(\mathrm{WT}_a^{(t)}(\hat{\mathbf{O}}_a), \mathrm{WT}_a^{(t)}(\mathbf{V}))$$

$$L_c^{int} = \mathrm{MSE}(\hat{\mathbf{O}}_c, \mathbf{V}) + \sum_{t=1}^{3} \mathrm{MSE}(\mathrm{WT}_c^{(t)}(\hat{\mathbf{O}}_c), \mathrm{WT}_c^{(t)}(\mathbf{V}))$$

$$L_s^{int} = \mathrm{MSE}(\hat{\mathbf{O}}_s, \mathbf{V}) + \sum_{t=1}^{3} \mathrm{MSE}(\mathrm{WT}_s^{(t)}(\hat{\mathbf{O}}_s), \mathrm{WT}_s^{(t)}(\mathbf{V}))$$

– Incremental Cross-view Mutual Distillation:

$$L_c^n = \sum\nolimits_{(x,y,z)\in\mathbb{T}_c^n(\gamma)} \frac{(\mathbf{O}_a^n[x,y,z]-\mathbf{O}_c^n[x,y,z])^2}{|\mathbb{T}_c^n(\gamma)|}$$

$$L_s^n = \sum\nolimits_{(x,y,z)\in\mathbb{T}_s^n(\gamma)} \frac{(\mathbf{O}_a^n[x,y,z]-\mathbf{O}_s^n[x,y,z])^2}{|\mathbb{T}_s^n(\gamma)|}$$

– Memory Bank Regularization:

$$L^{com} = \sum_{i=1}^{l-1}\sum_{x=1}^{h/8}\sum_{y=1}^{w/8} \|\mathbf{E}_3^i[x,y] - \mathbf{M}[z_{\text{pos}}^i(x,y)]\|_2,$$

$$s.t.\ \ z_{\text{pos}}^i(x,y) = \operatorname*{argmax}_{z'} p_{x,y,z'}^i;$$

$$L^{sep} = \sum_{i=1}^{l-1}\sum_{x=1}^{h/8}\sum_{y=1}^{w/8} \max(\|\mathbf{E}_3^i[x,y] - \mathbf{M}[z_{\text{pos}}^i(x,y)]\|_2$$

$$- \|\mathbf{E}_3^i[x,y] - \mathbf{M}[z_{\text{neg}}^i(x,y)]\|_2 + \alpha, 0),$$

$$s.t.\ \ z_{\text{neg}}^i(x,y) = \operatorname*{argmax}_{z'\neq z_{\text{pos}}^i(x,y)} p_{x,y,z'}^i.$$

– Overall Training Loss:

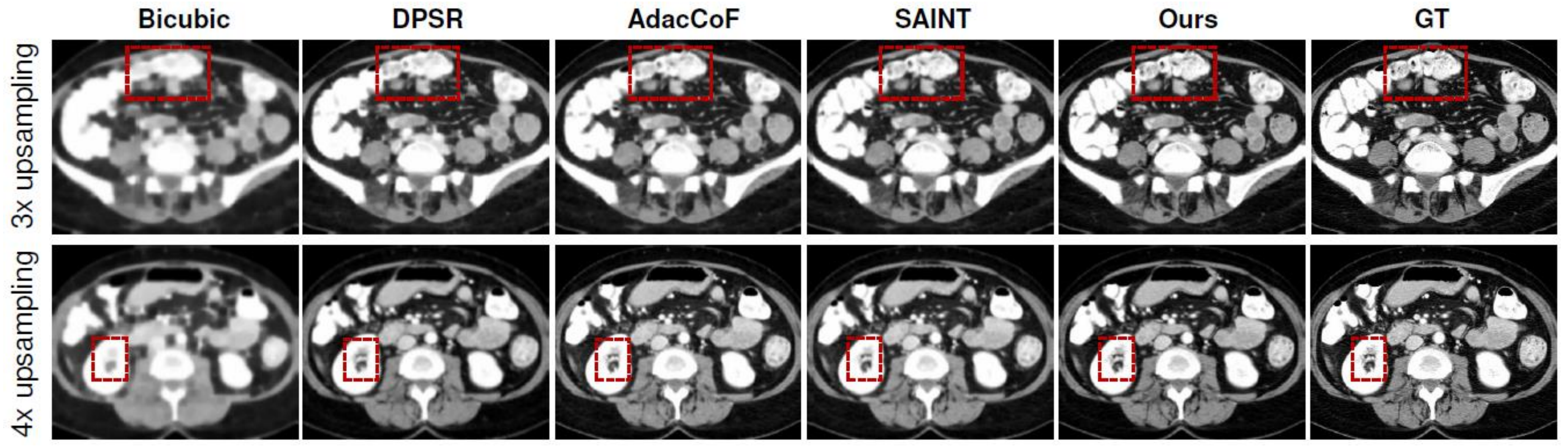
$$L = L_a^{int} + L_c^{int} + L_s^{int} + 0.15*(L_c^{cmd} + L_s^{cmd}) + 0.1*(L^{com} + L^{sep})$$

Experimental Results

- Quantitative Comparison

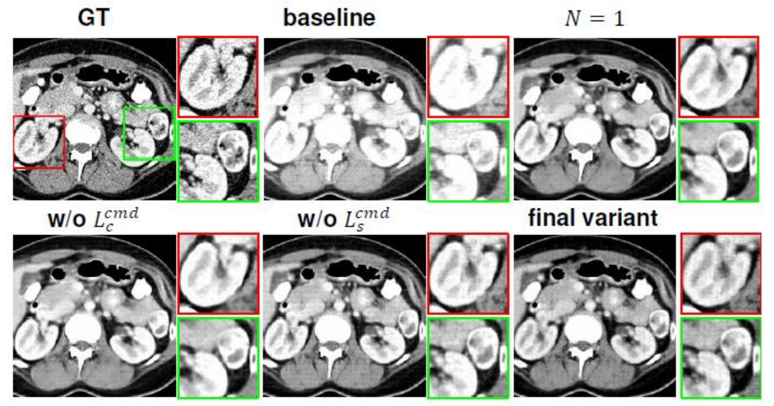
Method	2×				3×				4×			
	PSNR	$SSIM_a$	$SSIM_c$	$SSIM_s$	PSNR	$SSIM_a$	$SSIM_c$	$SSIM_s$	PSNR	$SSIM_a$	$SSIM_c$	$SSIM_s$
RDN [35]	43.51	0.9539	0.9519	0.9512	39.52	0.9402	0.9398	0.9376	37.89	0.9199	0.9210	0.9212
DPSR [34]	43.83	0.9690	0.9691	0.9682	38.82	0.9434	0.9423	0.9424	38.13	0.9166	0.9135	0.9154
MetaSR [8]	43.68	0.9547	0.9549	0.9548	39.90	0.9419	0.9425	0.9414	38.00	0.9211	0.9198	0.9214
RRIN [14]	43.45	0.9688	0.9691	0.9682	38.68	0.9428	0.9424	0.9422	38.10	0.9255	0.9232	0.9252
SRGAN [12]	43.22	0.9524	0.9521	0.9522	38.54	0.9433	0.9429	0.9425	37.91	0.9213	0.9209	0.9207
3D-MDSR [15]	44.31	0.9692	0.9698	0.9689	40.22	0.9489	0.9489	0.9490	38.20	0.9307	0.9302	0.9310
AdaCoF [13]	44.88	0.9749	0.9746	0.9747	40.92	0.9513	0.9498	0.9451	38.23	0.9311	0.9148	0.9150
SAINT [24]	44.43	0.9694	0.9641	0.9632	40.81	0.9448	0.9388	0.9416	38.42	0.9259	0.9175	0.9203
Ours	**46.81**	**0.9792**	**0.9784**	**0.9786**	**42.94**	**0.9631**	**0.9589**	**0.9604**	**41.11**	**0.9404**	**0.9385**	**0.9382**

- Qualitative Comparison



- Ablation Study

Variant	PSNR	$SSIM_a$
w/o L_c^{cmd} or L_s^{cmd}	38.58	0.9286
w/o L_c^{cmd}	40.26	0.9322
w/o L_s^{cmd}	40.24	0.9321
N=1	40.47	0.9325
w/o WT	40.56	0.9334
w/o memory	40.28	0.9324
final variant	41.11	0.9404



Conclusion

- This work proposes an incremental cross-view mutual distillation pipeline to tackle the CT slice synthesis task.
- The mutual distillation between each view and incremental learning process contributes to a slice synthesizer with appealing performance.
- Extensive experiments on the CT dataset demonstrate the superiority of our method against existing slice synthesis methods.