Eliminating the Dirty Beam Effect in 21cm Foreground Subtraction with Unsupervised Algorithm

Shulei Ni, Yichao Li, Li-Yang Gao, Xin Zhang 2022, ApJ, 934, 83

Shulei Ni(倪书磊)

ZheJiang Lab

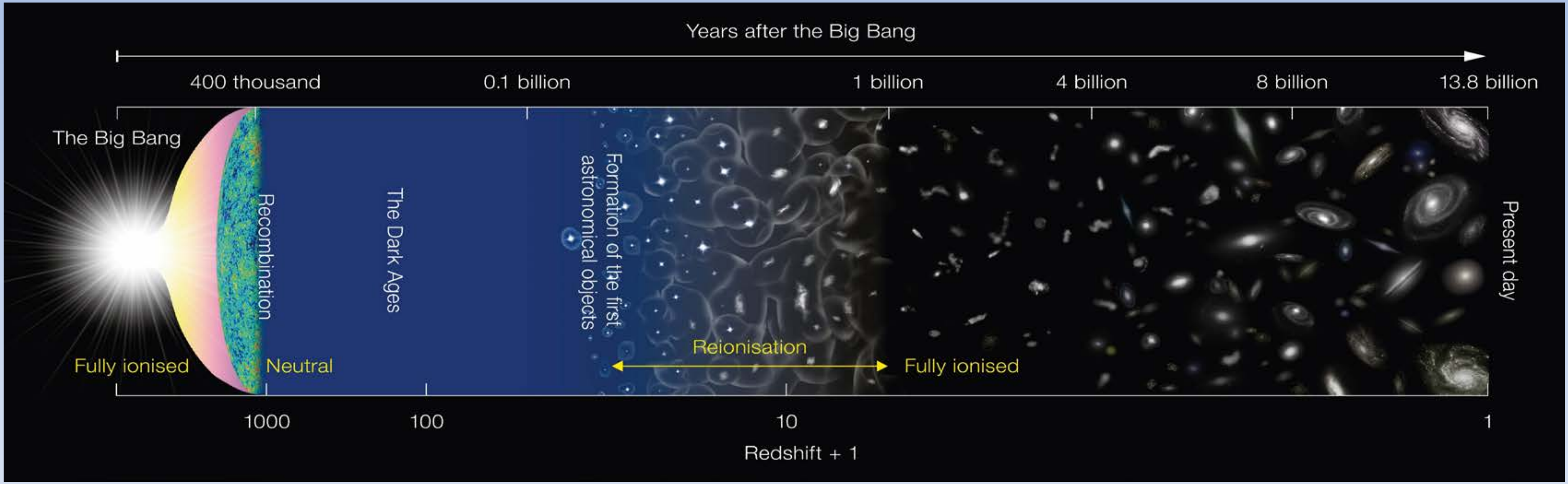
China · ShenYang

2023.07.18

之江实验室 ZHEJIANG LAB 智能计算研究院 RESEARCH INSTITUTE OF INTELLIGENT COMPUTING

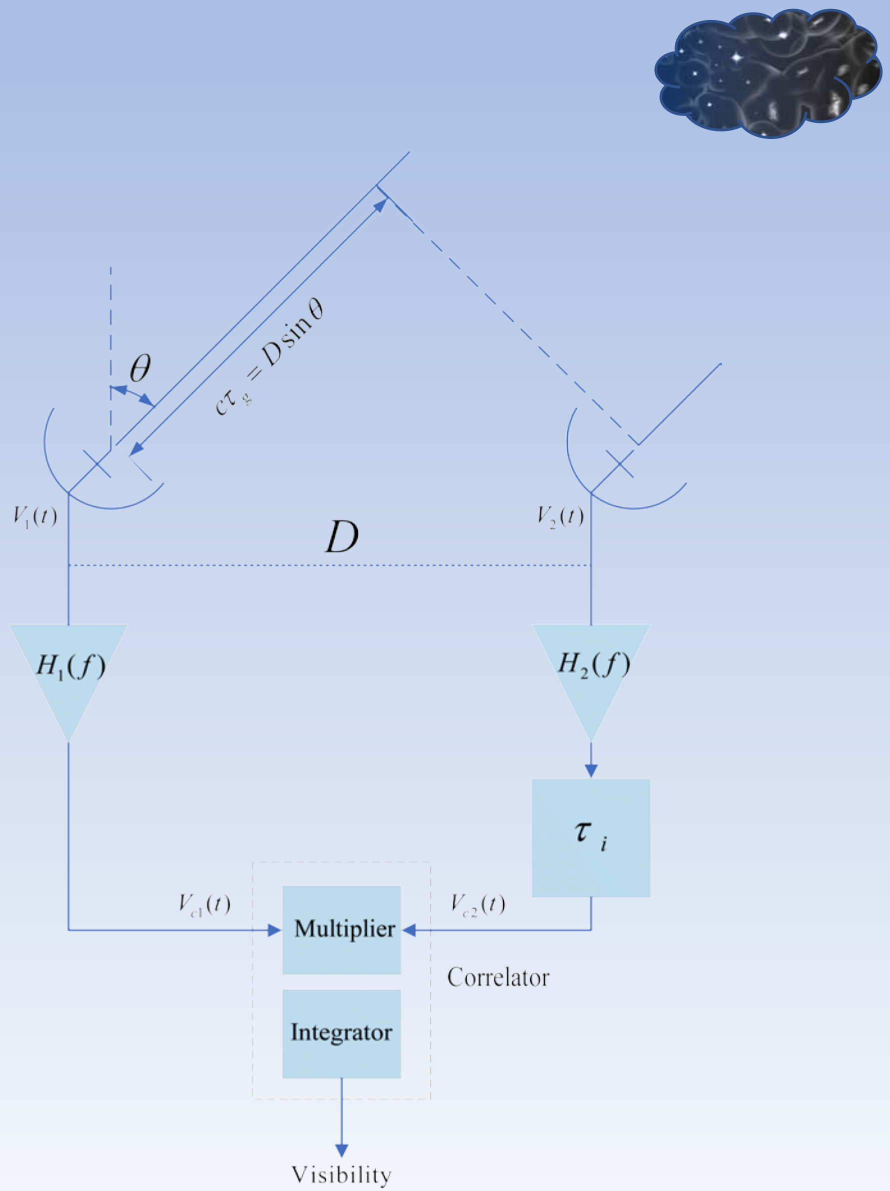
Content

1. 21 cm Cosmology & Data Characteristics



1. Cosmic Dark Ages (30 < z < 1100, 0.38-101 MY)
2. Cosmic Dawn (15 < z < 30, 101-272 MY)
3. Epoch of Reionization (6 < z < 15, 272 – 941 MY)

1. 21 cm Cosmology & Data Characteristics



Visibility(V):

$$V(\mu,\nu)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}A(x,y)I(x,y)e^{i2\pi(\mu x+\nu y)}dxdy$$

FFT:

$$I'(x,y)=A(x,y)I(x,y)=\int_{-\infty}^{\infty}V(\mu,\nu)e^{-i2\pi(\mu x+\nu y)}d\mu d\nu$$

Gridded Visibility Function and DFT, Dirty Image:

$$I_D(x,y)=\sum_k g(\mu_k,\nu_k)V(\mu_k,\nu_k)e^{-i2\pi(\mu_k x+\nu_k y)}$$

FFT ⇋ Convolution :

$$I_D(x,y)=P_D(x,y)\otimes I'(x,y)$$

I' is real image, P_D is the dirty beam (PSF):

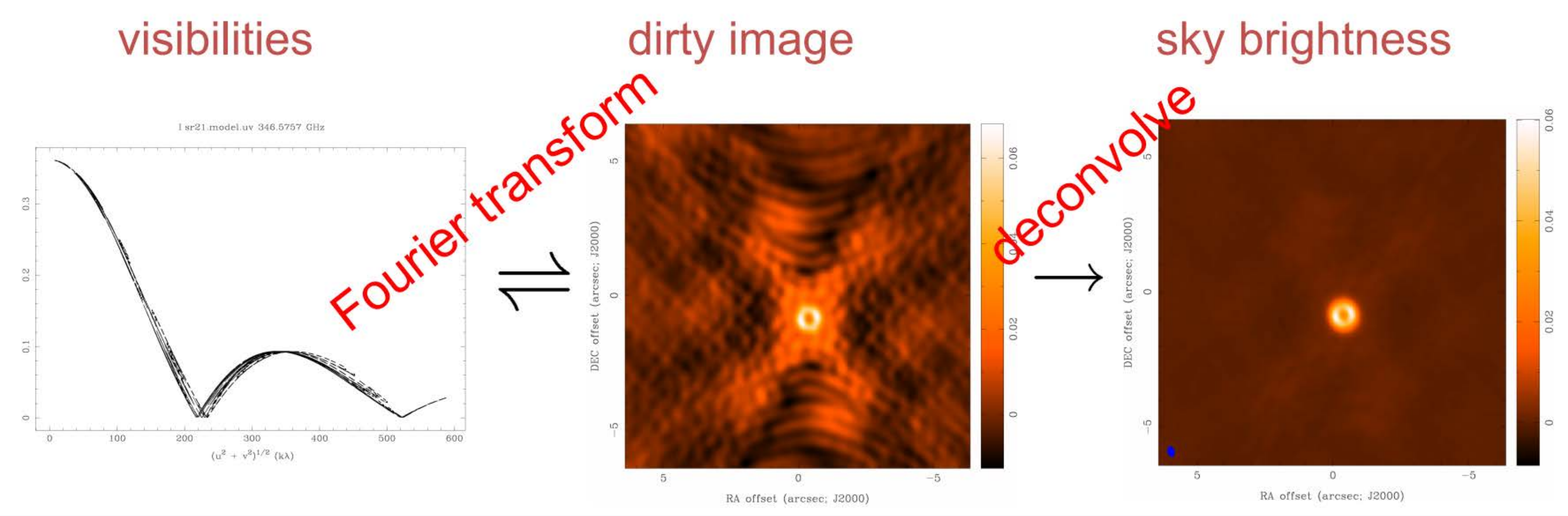
$$P_D=\sum_k g(\mu_k,\nu_k)e^{-2i\pi(\mu_k x+\nu_k y)}$$

Formula(traditional):

$$V(\mu, \nu) \xrightarrow{\text{FFT}} \text{Dirty Image} = \text{Dirty Beam} \otimes \text{Real Image}$$

Flow chart:

$$\text{Clean Beam} \otimes \text{Model} = \text{Clean Image}$$



from casa

1. 21 cm Cosmology & Data Characteristics

Traditional methods:

WSClean
Project ID: 19986999

CASA

SDC3 data examples:

wsclean -reorder -use-wgridder -parallel-gridding 10 **-weight natural** -oversampling 4095 -kernel-size 15 -nwlayers 1000 -grid-mode kb -taper-edge 100 -padding 2 -name out_name -size 2048 2048 -scale 16asec **-niter 0** -pol xx -make-psf in_ms

wsclean -no-update-model-required -use-wgridder -multiscale -parallel-gridding 10 **-weight uniform** -oversampling 4095 -kernel-size 15 -nwlayers 1000 -grid-mode kb -taper-edge 100 -padding 2 -taper-gaussian 60 -super-weight 4 -name out_name -size 2048 2048 -scale 16asec **-niter 1.e6** -auto-threshold 4 -mgain 0.8 -pol xx -make-psf '+out_msfrq

tclean(vis=‘xx.ms’, imagename=‘SNR.MultiScale’, deconvolver=‘multiscale’, scales=[0,6,10,30,60], smallscalebias=0.9, imsize=4095, cell=‘16arcsec', pblimit=-0.01, niter=1000,weighting='briggs', stokes=‘I’, robust=0.0, interactive=False, threshold='0.12mJy', savemodel='modelcolumn')

One problem with both of software:

we don’t know what iteration control(niter) is, and if we set niter as a big value, need to a lot of computing power and computer memory, and spend a long time!

So, We try to use deep learning !!!

2. Deep Learning & Application Scenarios

- **Data Layer:**

  SDSS: total Public 116TB (Data Release 12)
  FAST: more than 6Gbit/s, and 20PB/a
  SKA1: 34Gbit/s, and 1.072EB/a
  - SKA1-low(No Reion): 3Gbit/s, and 94.6PB/a;
  - Reion: 22Gbit/s, and 693.8PB/a;
  - SKA1-mid: 9Gbit/s, and 283.8PB/a

- **Algorithm Layer:**

  Many DP Models:
  - CNN, GAN, U-Net, …

  Applications:
  - Data Simulation, Cluster Analysis, Logistic Regression, Neural Networks, ODE/PDE, PCA, Bayesian,…

2. Deep Learning & Application Scenarios

之江实验室 ZHEJIANG LAB 智能计算研究院 RESEARCH INSTITUTE OF INTELLIGENT COMPUTING

1. Classification and Identification:

   1801.06381(CNN), 1912.04412(RF), 2006.05998(CNN), 2106.06587(RF), 2207.09072(CNN), ···

2. Super-Resolution, Generation and Enhancement:

   2002.07940(GAN), 2004.09206(GAN), 2205.06758(GAN), 2206.15131(GAN), ···

3. Parameter Estimation:

   1812.04631(GP), 1911.08508(BNN), 2205.10881(AE), ···

1. Reionization and 21cm: CNN, U-Net, GAN, ···
2. Gravitational Lensing: CNN, U-Net, ···
3. Large-Scale Structure: V-Net, CNN, GAN, ···
4. CMB: U-Net, GAN, CNN, ···
5. Parameter Estimation: BNN, GAN, ···

Foreground Removal:

- HI Foreground:

  V-Net, GP, VAE, ···

- CMB Foreground:

  GAN, BNN, NN, U-Net, ···

3. De-Conv U-Net Model

Based on a conclusion of our previous work: S. Ni et al. 2022, ApJ, 934, 83

DATA: simulation with CRIME

Success: simple beam model (gaussian beam), PCA can remove the foreground;

Failure: complex beam model (MeerKAT beam, cosine beam), PCA can not work.

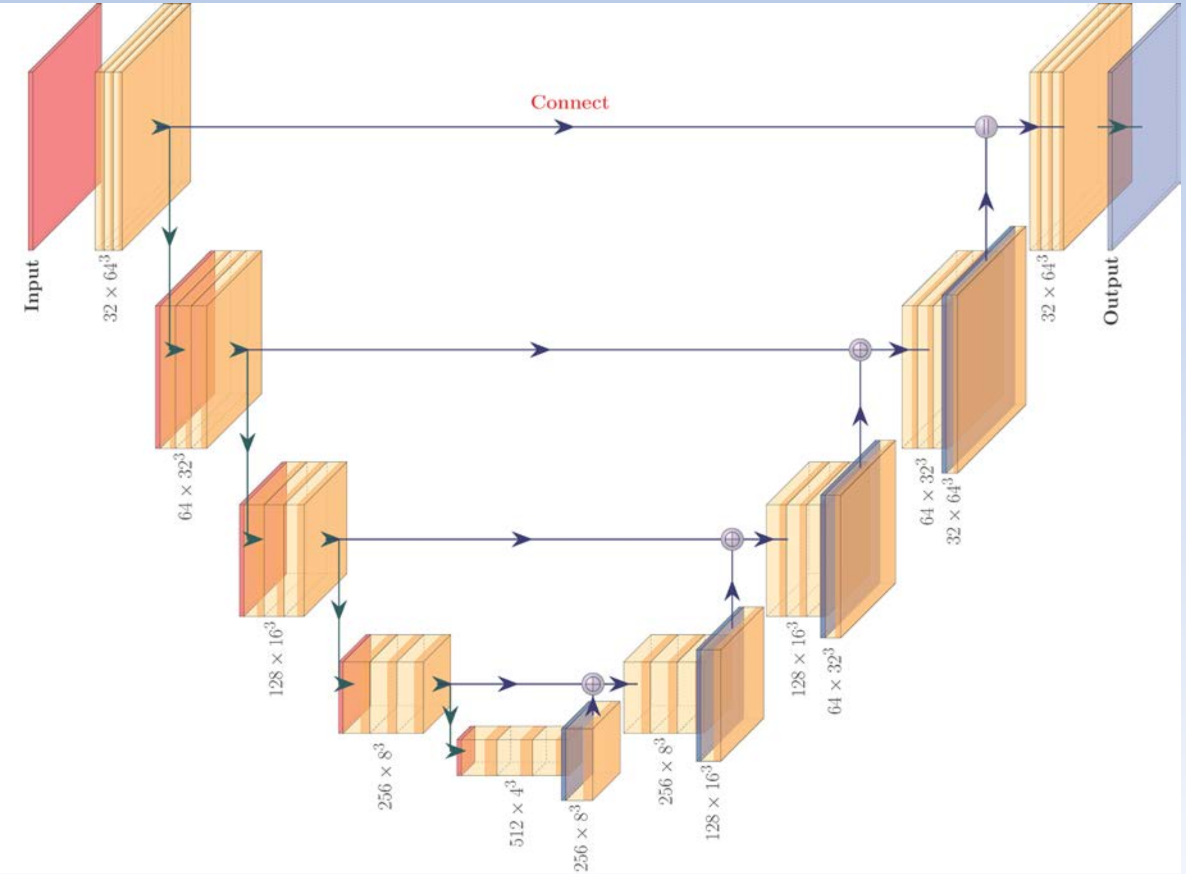
Formula (PCA, Smooth Beam, UNet as an operator**):**

Gaussian: $\begin{cases} \mathrm{PCA}_{res}[\mathrm{Gaussian}_{beam}\ (FG + HI)] \approx HI \\ \mathrm{UNet}\{\mathrm{PCA}_{res}[\mathrm{Gaussian}_{beam}\ (FG + HI)]\} \approx HI \end{cases}$

Cosine: $\begin{cases} \mathrm{PCA}_{res}[\mathrm{Cosine}_{beam}\ (FG + HI)] \neq HI \\ \mathrm{UNet}\{\mathrm{PCA}_{res}[\mathrm{Cosine}_{beam}\ (FG + HI)]\} \approx HI \end{cases}$

Train: $\mathrm{PCA}_{res}(FG + HI)$

Label: HI



3. De-Conv U-Net Model

Based on a conclusion of our previous work: S. Ni et al. 2022, ApJ, 934, 83

Success: simple beam model (gaussian beam), PCA can remove the foreground;

Failure: complex beam model (cosine beam), PCA can not work.

Formula (PCA, Smooth Beam, UNet as an operator**):**

Gaussian:
$$\begin{cases} \mathrm{PCA}_{res}[\mathrm{Gaussian}_{beam}\,(FG+HI)] \approx HI \\ \mathrm{UNet}\{\mathrm{PCA}_{res}[\mathrm{Gaussian}_{beam}\,(FG+HI)]\} \approx HI \end{cases}$$

Cosine:
$$\begin{cases} \mathrm{PCA}_{res}[\mathrm{Cosine}_{beam}\,(FG+HI)] \neq HI \\ \mathrm{UNet}\{\mathrm{PCA}_{res}[\mathrm{Cosine}_{beam}\,(FG+HI)]\} \approx HI \end{cases}$$

Train: $\mathrm{PCA}_{res}(FG+HI)$
Label: HI

⇒

Gaussian:
$$\begin{cases} \mathrm{PCA}_{res}[\mathrm{Gaussian}_{beam}\,(FG+HI)] \approx HI \\ \mathrm{PCA}_{res}\{\mathrm{UNet}[\mathrm{Gaussian}_{beam}\,(FG+HI)]\} \approx HI \end{cases}$$

Cosine:
$$\begin{cases} \mathrm{PCA}_{res}[\mathrm{Cosine}_{beam}\,(FG+HI)] \neq HI \\ \mathrm{PCA}_{res}\{\mathrm{UNet}[\mathrm{Cosine}_{beam}\,(FG+HI)]\} \approx HI \end{cases}$$

Train: $\mathrm{Smooth}_{beam}(FG+HI)$
Label: $(FG+HI)$

3. De-Conv U-Net Model

Based on a conclusion of our previous work: S. Ni et al. 2022, ApJ, 934, 83

Success: simple beam model (gaussian beam), PCA can remove the foreground;

Failure: complex beam model (cosine beam), PCA can not work.

Formula (PCA, Smooth Beam, UNet as an operator**):**

$$\mathrm{DeConv\ U-Net}: \quad (\mathrm{UNet} + \mathrm{Conv}_{Beam})[Beam(FG + HI)] = Beam(FG + HI)$$

Dataset: $\mathrm{Train} = \ \mathrm{Label} : \mathrm{Beam}(FG + HI)$

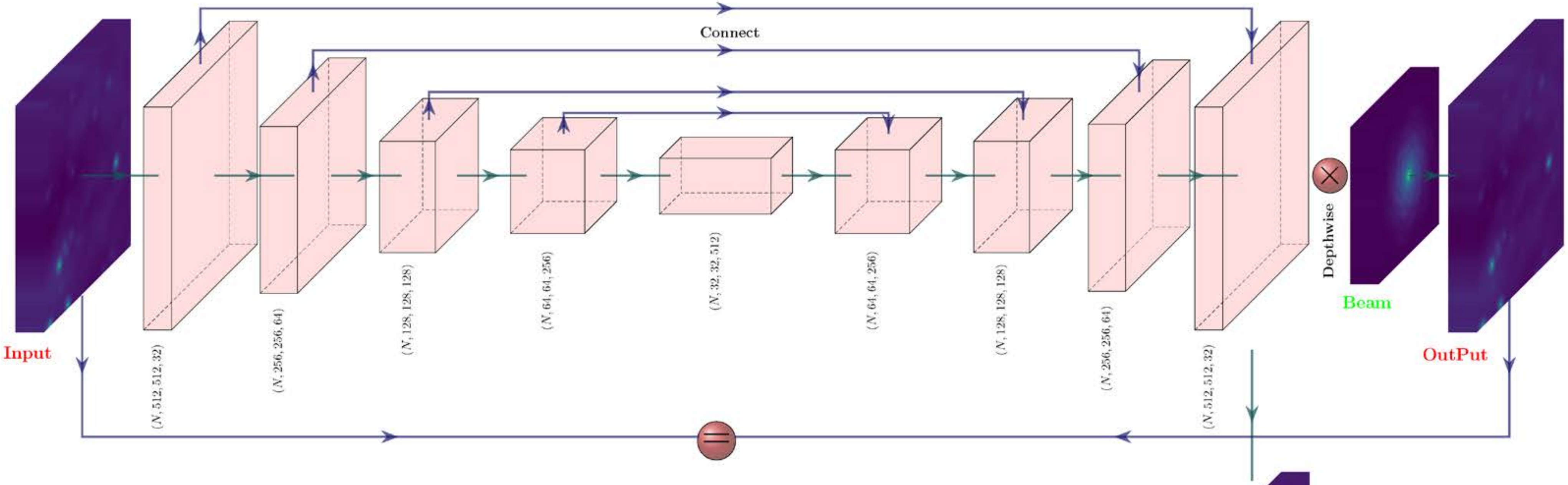
(set Iteration Control as 0(niter=0, SDC3 natural weight), no deconvolution, we use De-Conv U-Net)

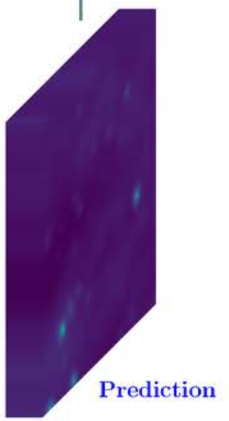
Unet: foreground removal

Conv_{Beam}: beam effect

3. De-Conv U-Net Model

De-Conv U-Net Structure :





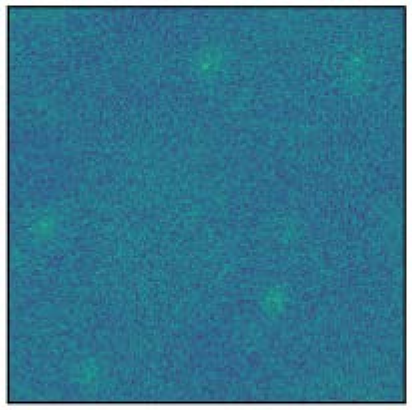
Pink Box: U-Net network

$\text{Input} = \text{Ouput} = \text{Image}_{\text{smoothed}}$ $\text{Beam} = \text{Dirty Beam}$

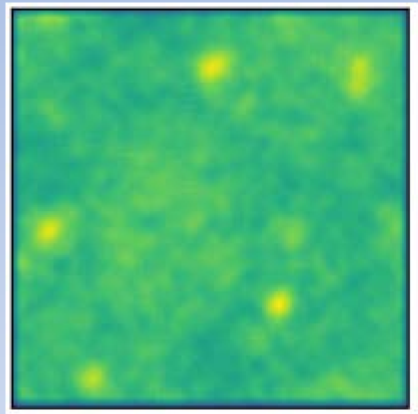
$\text{Prediction} = \text{Image}$(the result we want)

3. De-Conv U-Net Model

Simulation Data Result:



$$\otimes \quad \mathrm{Kernel}_{\mathrm{Random}} \quad =$$



Prediction

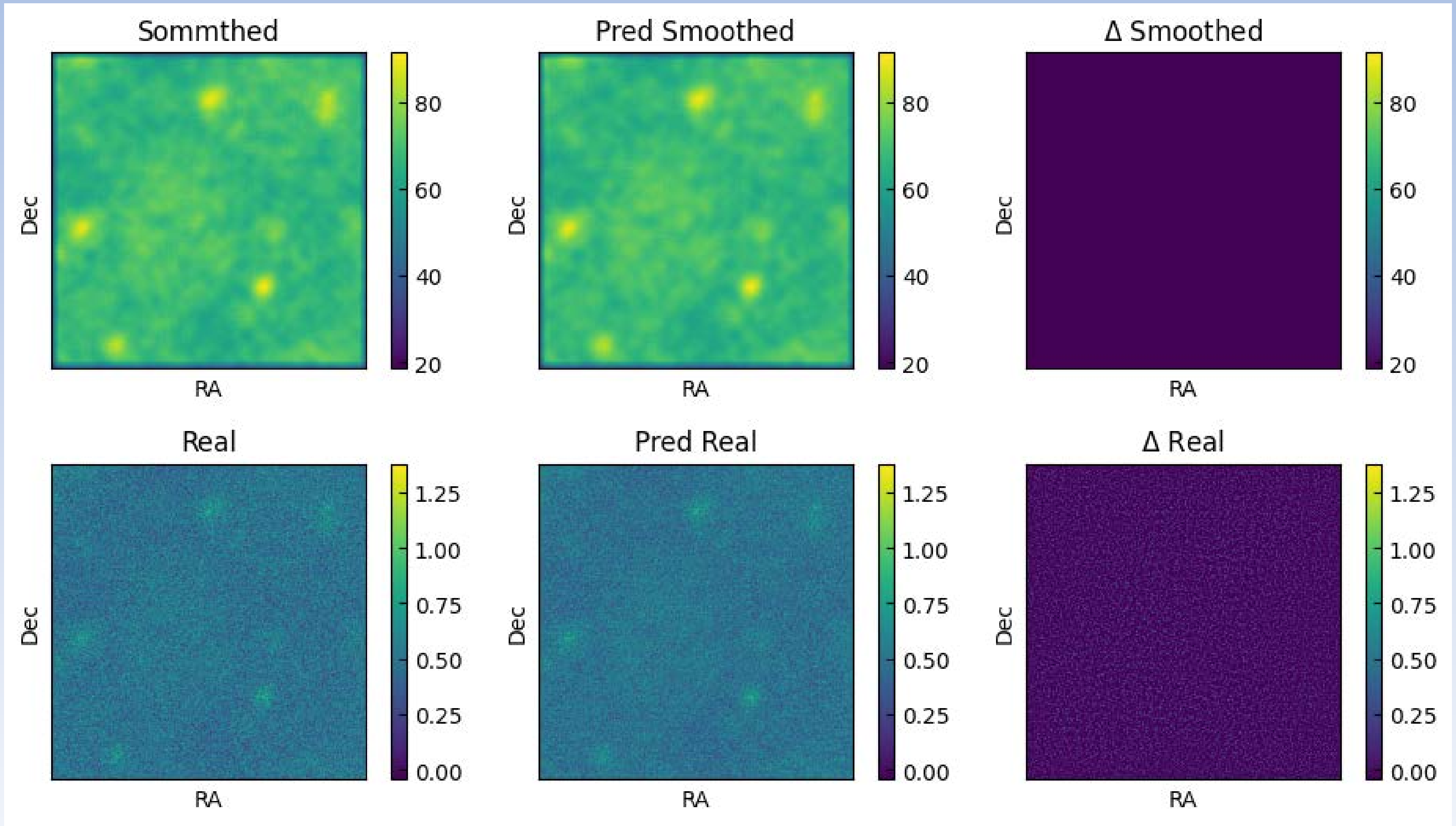
Beam

Input
and
Output

3. De-Conv U-Net Model

之江实验室 ZHEJIANG LAB 智能计算研究院 RESEARCH INSTITUTE OF INTELLIGENT COMPUTING

Simulation Data Result:

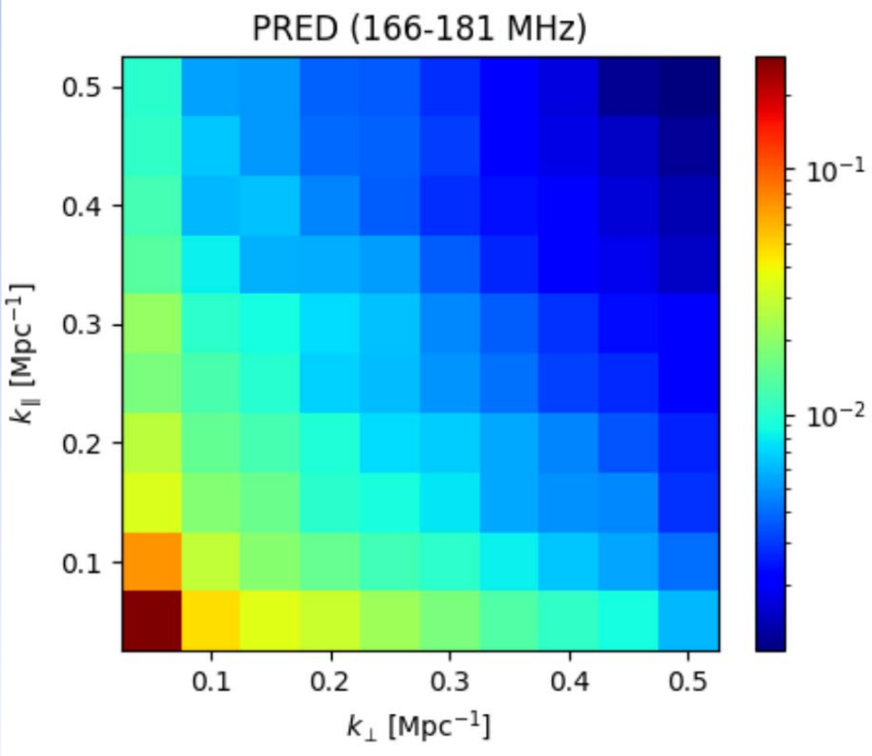


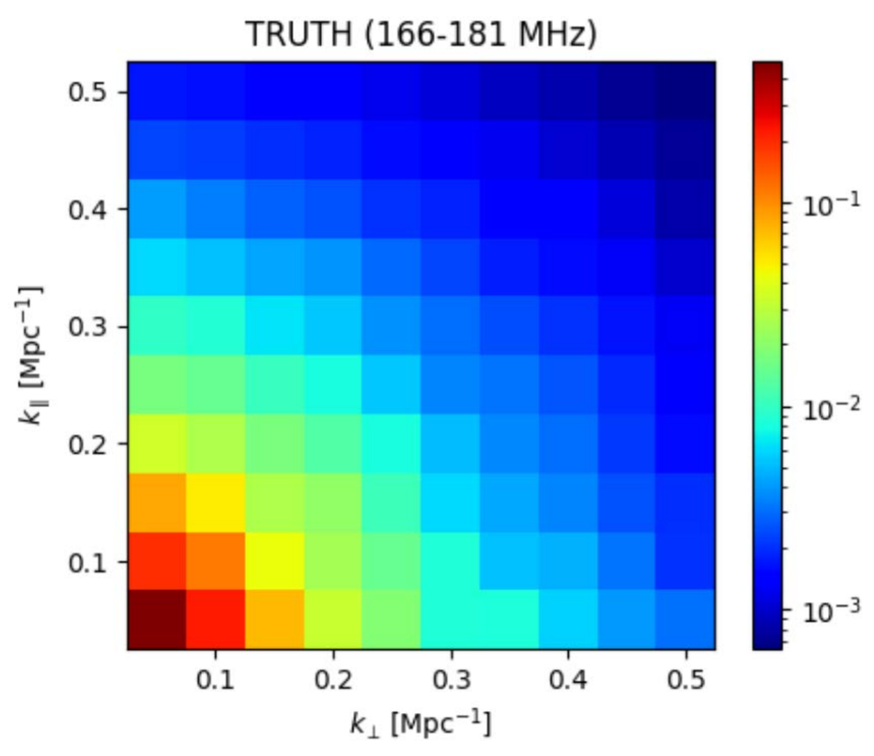
3. De-Conv U-Net Model

SDC3 Test Datasets Results(First Version):

Beam(Noise + HI)

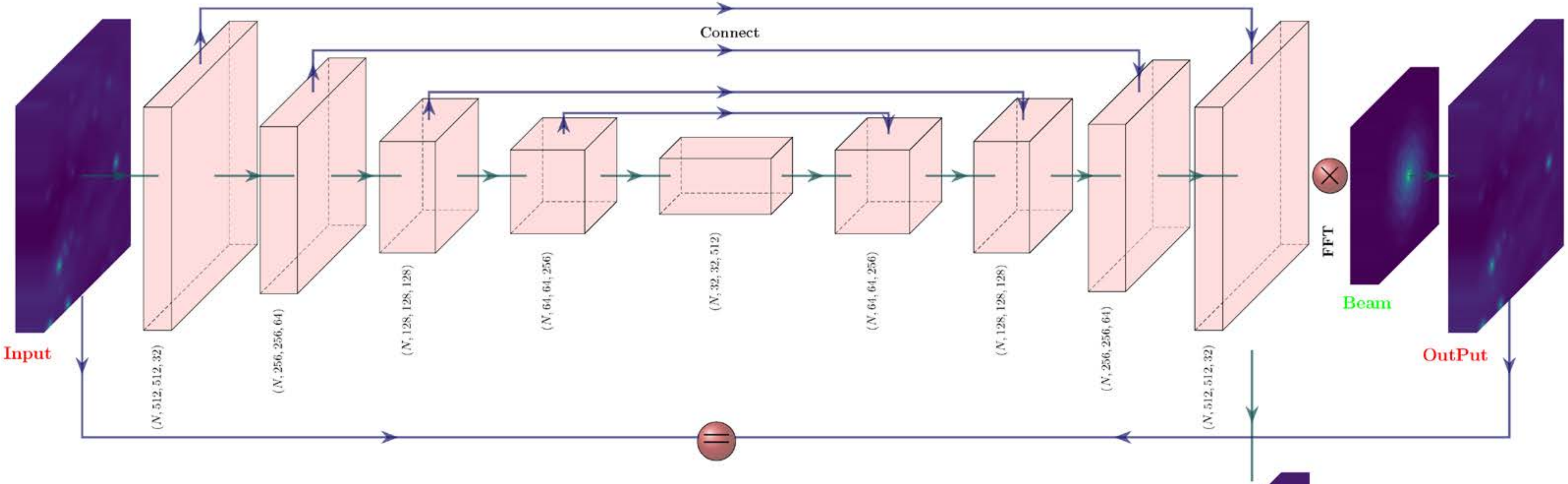
Size: center cube, (151, 512, 512)





3. De-Conv U-Net Model

The beam size is large (2048, 2048), resulting in very slow convolution.



Convolution → FFT

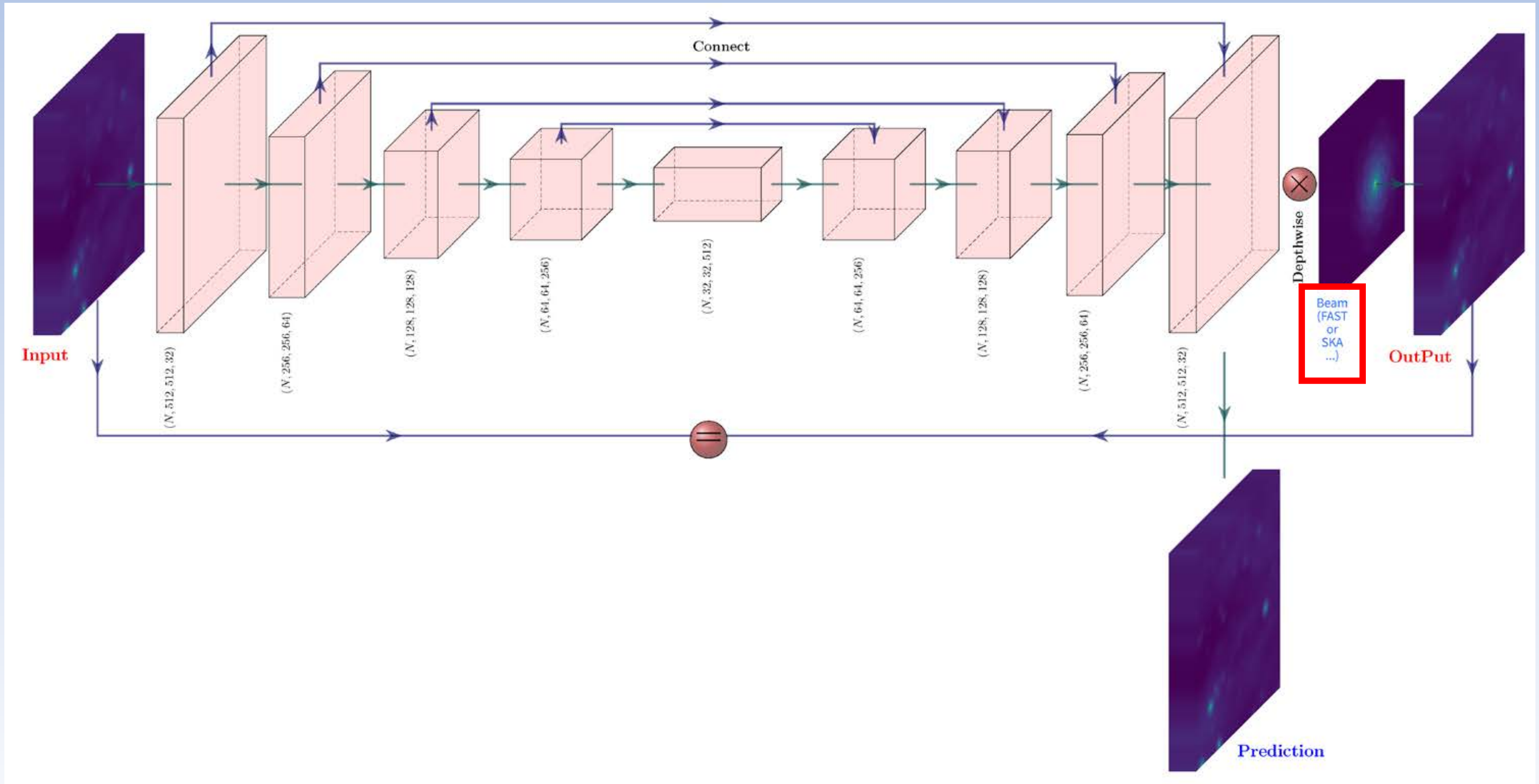
Time Complexity: $O(n^2) \rightarrow O(log(n))$

Faster ~ 10^3 times than the convolution!

4. Applications Extending

Different Beams:

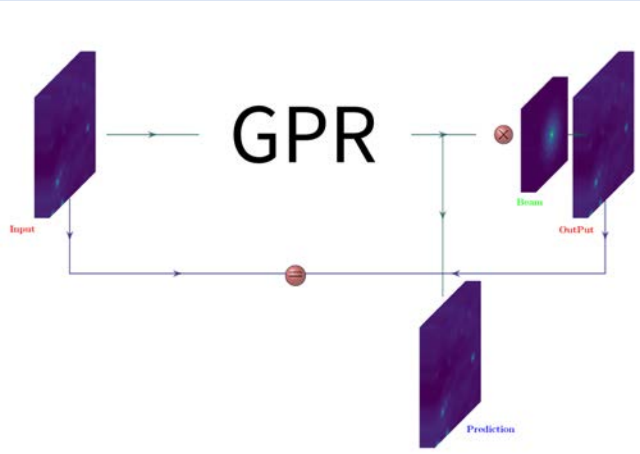
Single Dish (FAST) or Interferometric Array (SKA)

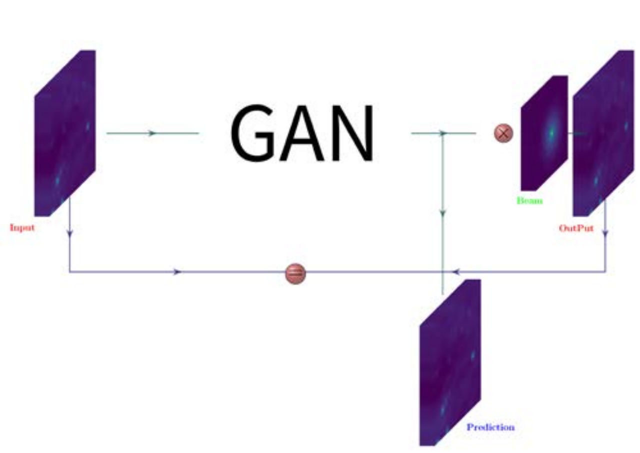


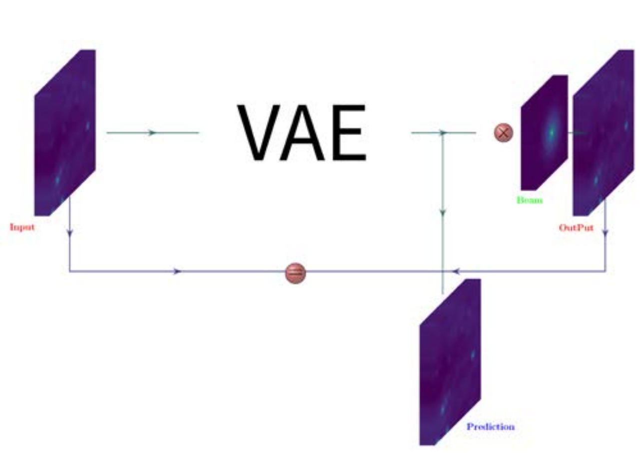
4. Applications Extending

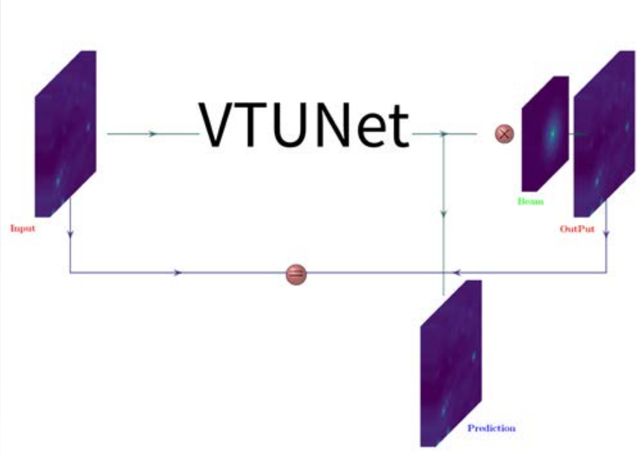
Different AI Methods:

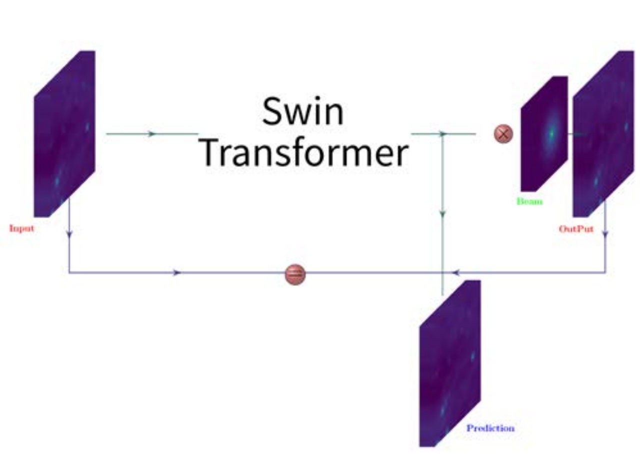
Gaussian Processing Regressive, Generative Adversarial Network, Variational AutoEncoder, Volumetric Transformer Unet, Swin Transformer











Summary

The paper will be pre-published on ArXiv
in the next few days!

Advertising

之江实验室 ZHEJIANG LAB | 智能计算研究院 RESEARCH INSTITUTE OF INTELLIGENT COMPUTING

天文领域AI专家：

之江实验室智能计算天文团队，依托 “中国天眼”FAST，以及之江实验室智能计算数字反应堆，海量的数据和充裕的算力。

博士后：

税前40万+，相关资助综合年收入最高达百万！

方向： FRB, HI,谱线成像及处理, 高性能计算, AI for Astronomy 等相关领域。

Thanks