



UNIVERSITÀ DEGLI STUDI DI NAPOLI
FEDERICO II



**MACHINE LEARNING
FOR ASTROPHYSICS**
3RD EDITION VALLETTA, 31 AUG - 4 SEP, 2026

BRIDGING DIFFERENCE IMAGING AND SUPERNOVA PHYSICS USING TIMEGAN

MARIA ZAMPELLA

Phd student in Computational Intelligence
Dept. of Physics 'E. Pancini'
University of Naples 'Federico II'

In collaboration with:

Dott. Fabio Ragosta, University of Napoli "Federico II"

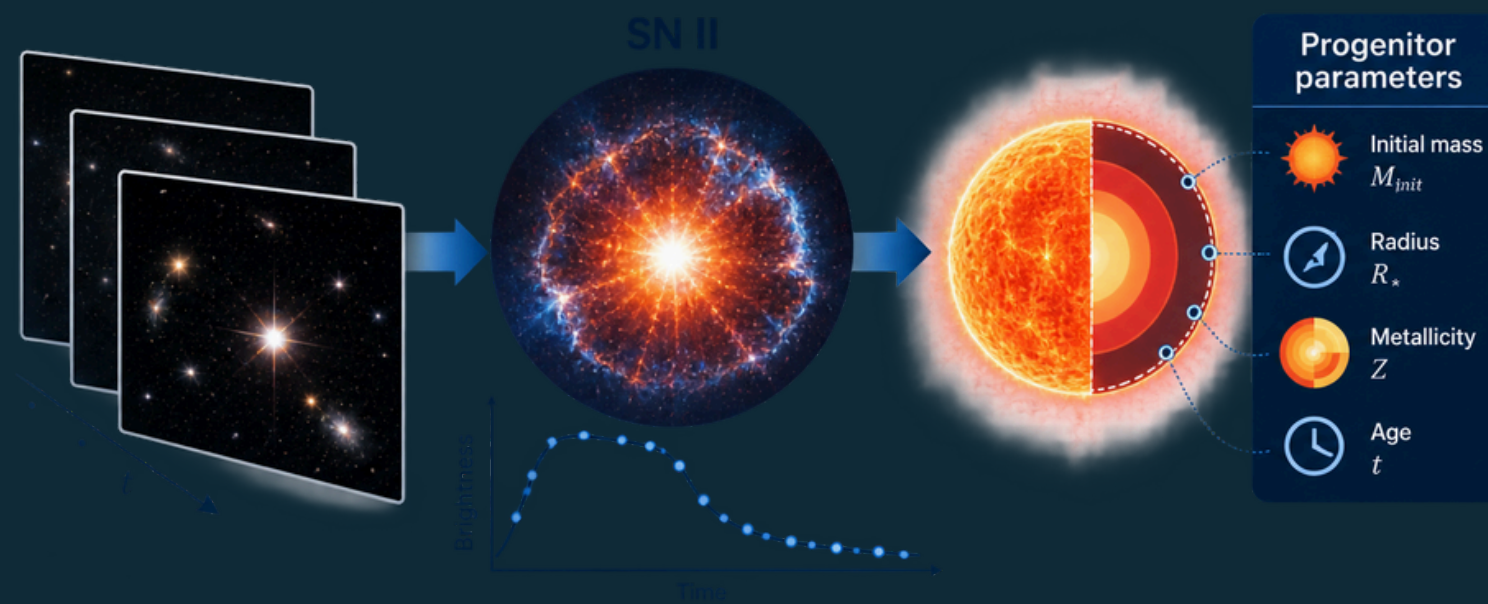
Prof. Massimo Brescia, University of Napoli "Federico II"

ML4ASTRO3 — University of Malta

31 August – 4 September 2026

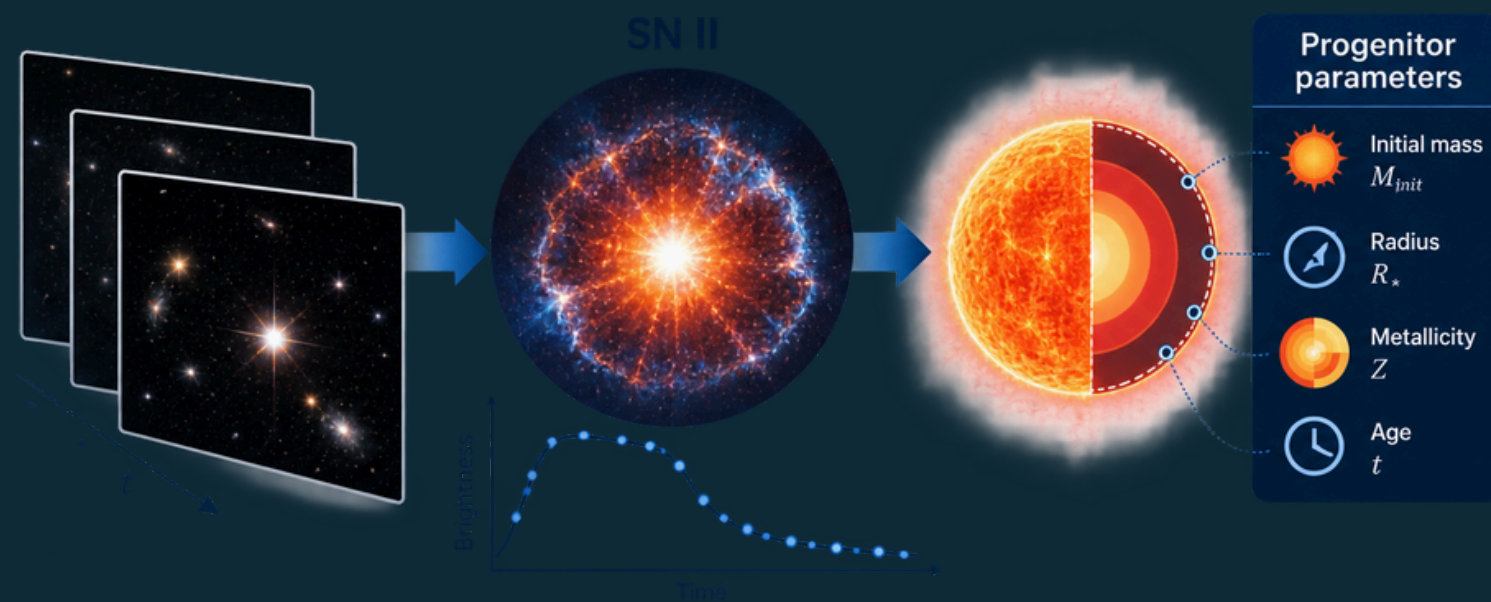
GOAL

- Learn a compact representation of supernova observations from their difference-image time series.
- Infer the physical parameters of a Type II supernova progenitor from the observed evolution of the event.



GOAL

- Learn a compact representation of supernova observations from their difference-image time series.
- Infer the physical parameters of a Type II supernova progenitor from the observed evolution of the event.



WORKFLOW

Obtain a significative embedding for the whole event by training a model for reconstructing the event.

01

- AE
- TIMEGAN

02

Use the embeddings for calculating the progenitor parameters

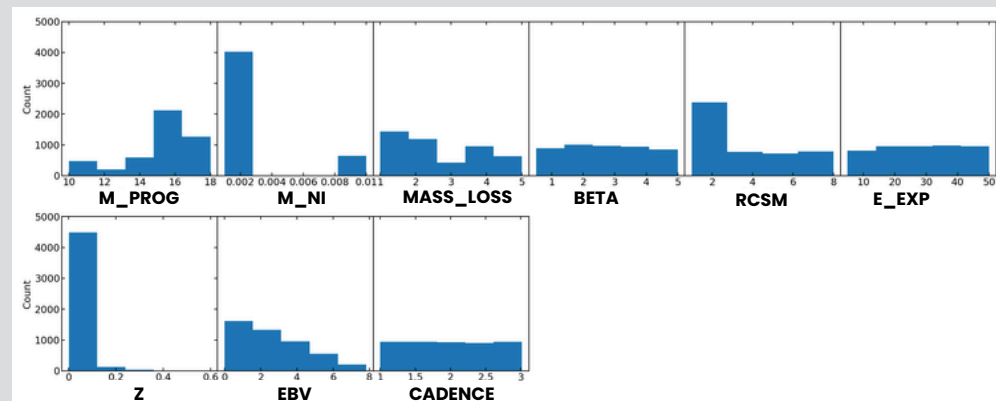
Difference-image time series → Learned embedding → Transformer regression → Progenitor parameters

DATA

SIMULATED DATASET

01

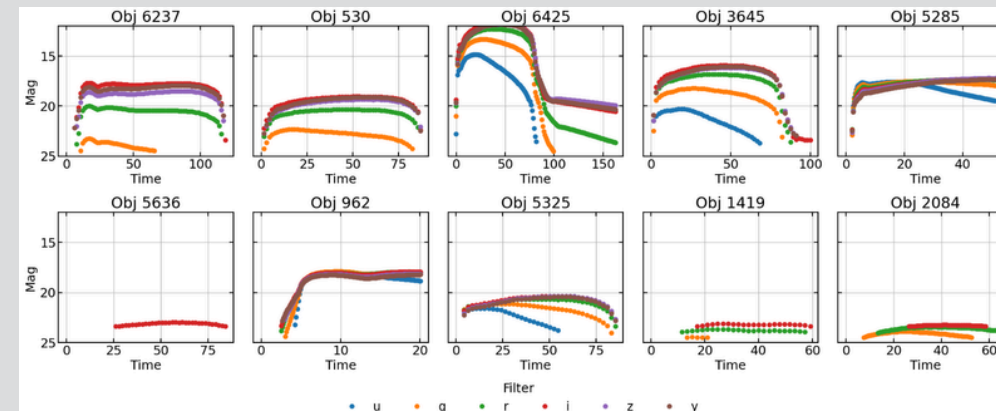
Sampling from parameter space distributions



Takashi J Moriya et al., Publications of the
Astronomical Society of Japan, 75, 3 (2023)

02

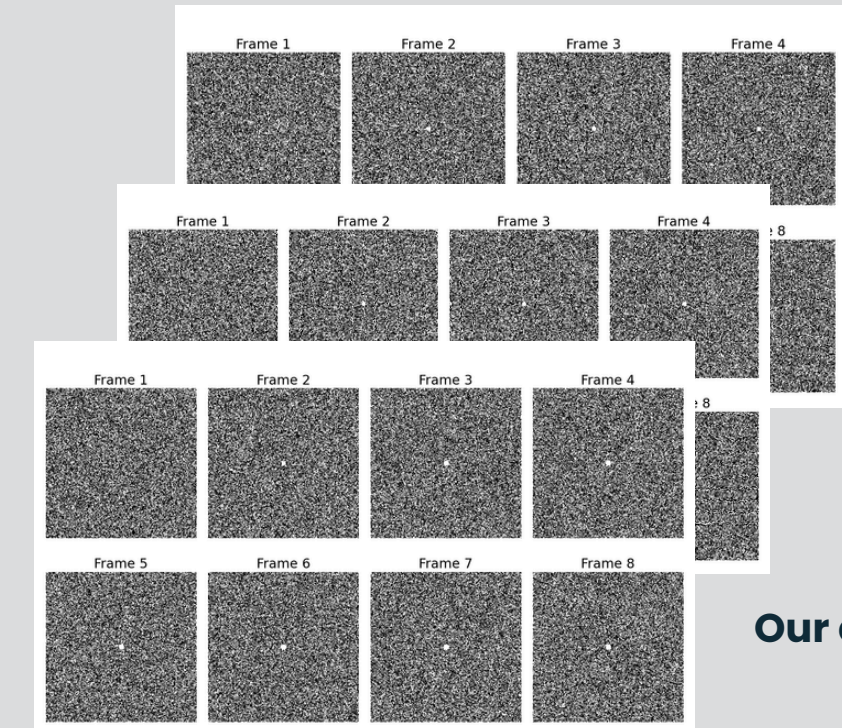
Simulate lightcurves



KEPLER
Weaver, Zimmerman, and Woosley, ApJ, 225, 1021
(1978)

03

Create the difference images



Our code

DATA

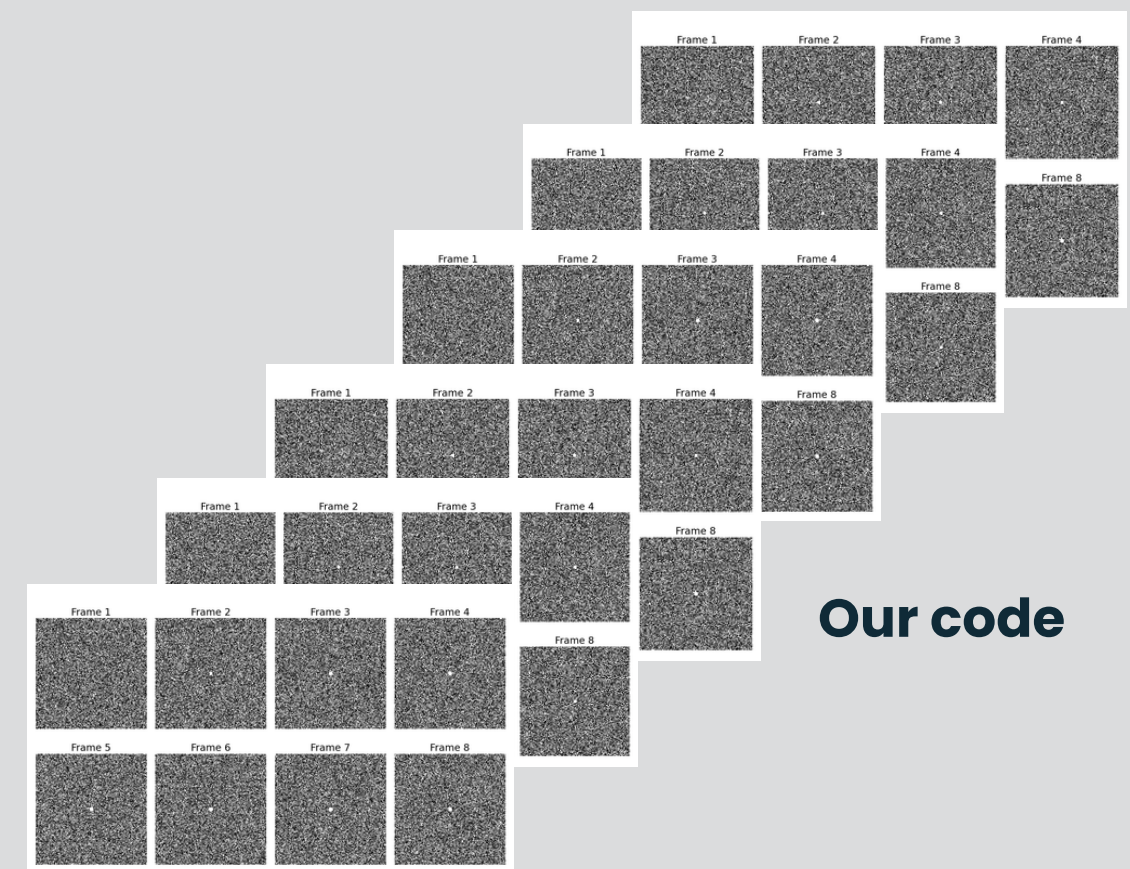
SIMULATED DATASET

We created a mock LSST Universe of Core-Collapse SNe (CCSNe) from which **we developed the expected *image difference*** to produce the light curves.

We **simulated the difference images** using the nominal LSST PSF. For each detection in the light curve, the PSF was obtained so that its integral is equal the flux of the simulated detection. Eventually, we embedded the PSF in a Gaussian background to obtain realistic difference images

03

Create the difference images for the six LSST filters (u,g,r,i,z,y)



n_CCSNe = 10,000, with at least 5 detections in 1 filter above the limiting magnitude

DATA

PREPROCESSING

01

Time step reduction

Cut all simulations up to 30 epochs



02

Normalization

Reduce the dynamic range while preserving the supernova–background contrast.

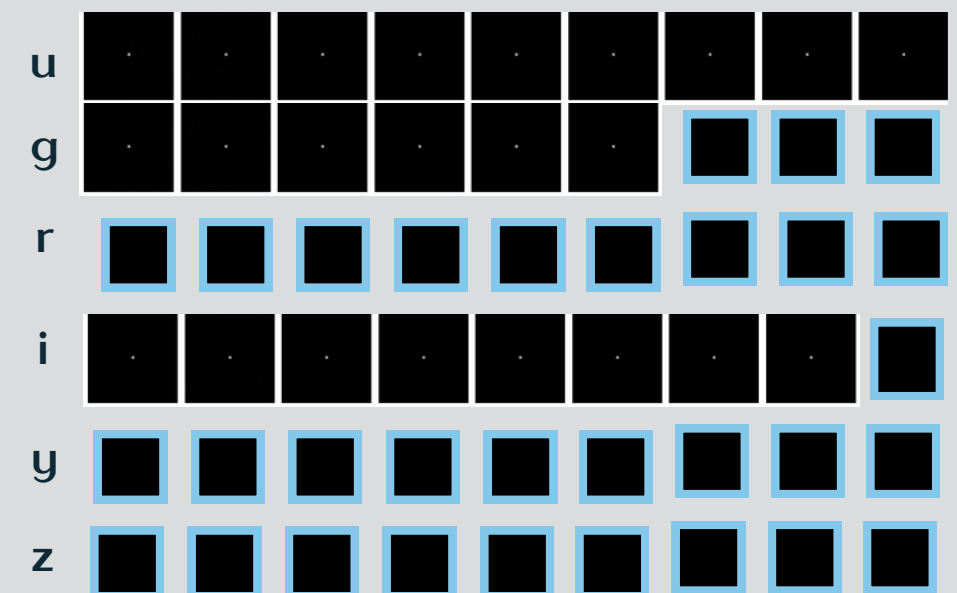
$$I_{norm} = asinh\left(\frac{I - \mu}{\sigma}\right)$$

I = input image
 μ = $mean(I_{border})$
 σ = $std(I_{border})$

03

Padding

Add padding for missing time steps and missing channels



DUAL-TASK ARCHITECTURE

Decompose the parameters' prediction task into two co-dependent subtasks

01. RECONSTRUCTION TASK (REPRESENTATION LEARNING)

Model training is driven by a reconstruction objective to learn a meaningful and compact latent representation of the supernova event, suitable for downstream tasks.

02. REGRESSION TASK (PHYSICAL PARAMETER PREDICTION)

Leverages the pre-trained, high-fidelity embeddings as input to a secondary predictor to estimate the physical properties of the supernova progenitor.

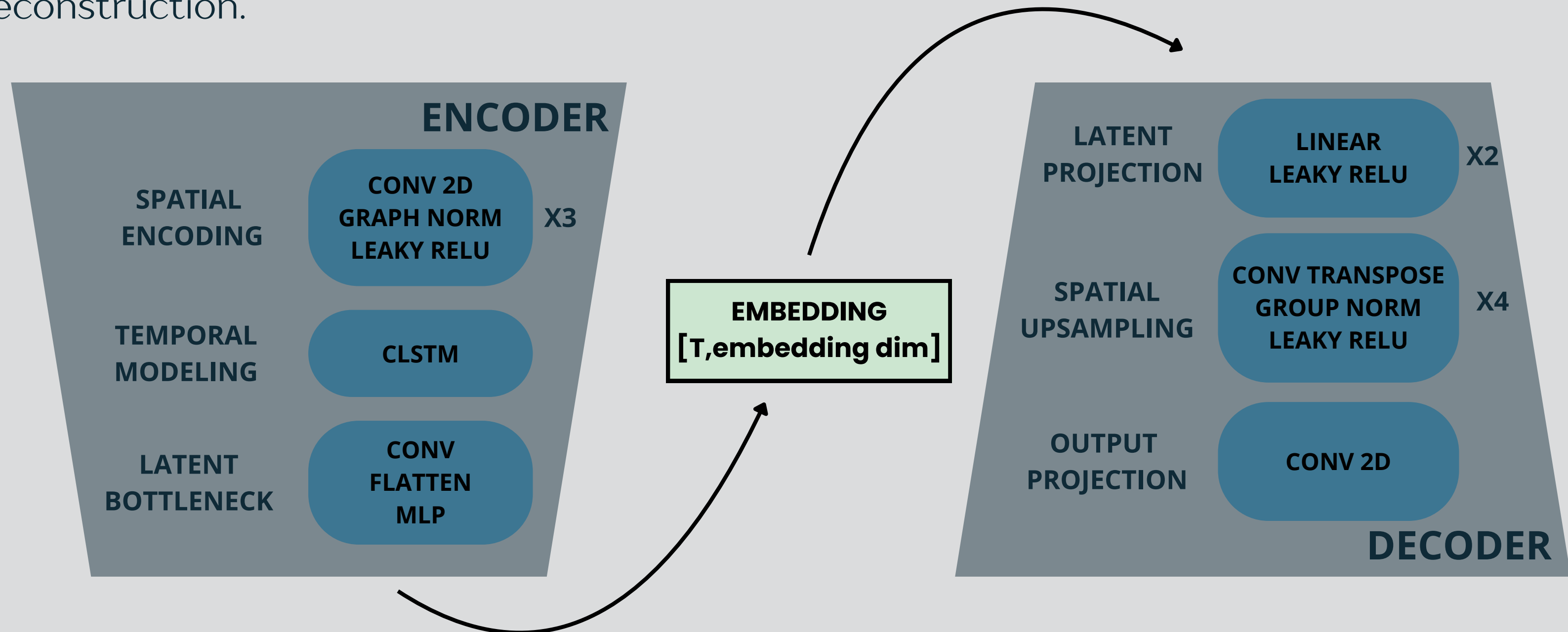
PART I: RECONSTRUCTION



AE

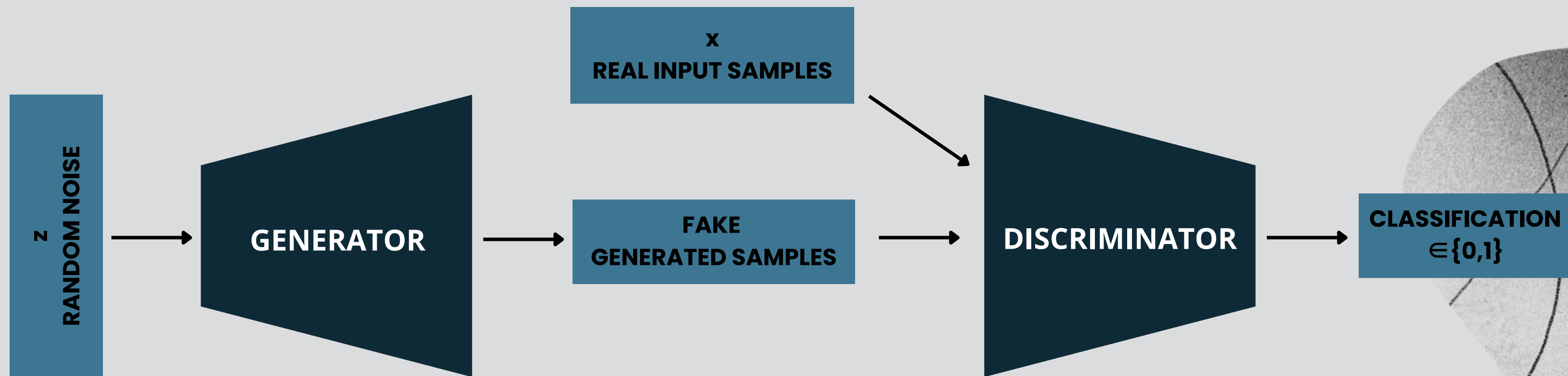
The Encoder maps the input into a lower-dimensional latent space that preserves the essential sequence information required for accurate reconstruction.

Spatial features are extracted by CNNs; **temporal dependencies** are modeled by ConvLSTM, and the resulting representation is compressed into a **low-dimensional latent space** for reconstruction.



GAN

A GAN consists of two separate neural networks — a **Generator** that creates synthetic data, and a **Discriminator** that evaluates it. They are **trained in opposition**: the Generator tries to fool the Discriminator, while the Discriminator learns to detect fakes, resulting in increasingly realistic data generation.



GAN loss: $E_x [\log (D (x))] + E_z [\log (1 - D (G (z)))]$

The generator tries to minimize this function, while the Discriminator tries to maximize it

TIMEGAN

1. Reconstruction:

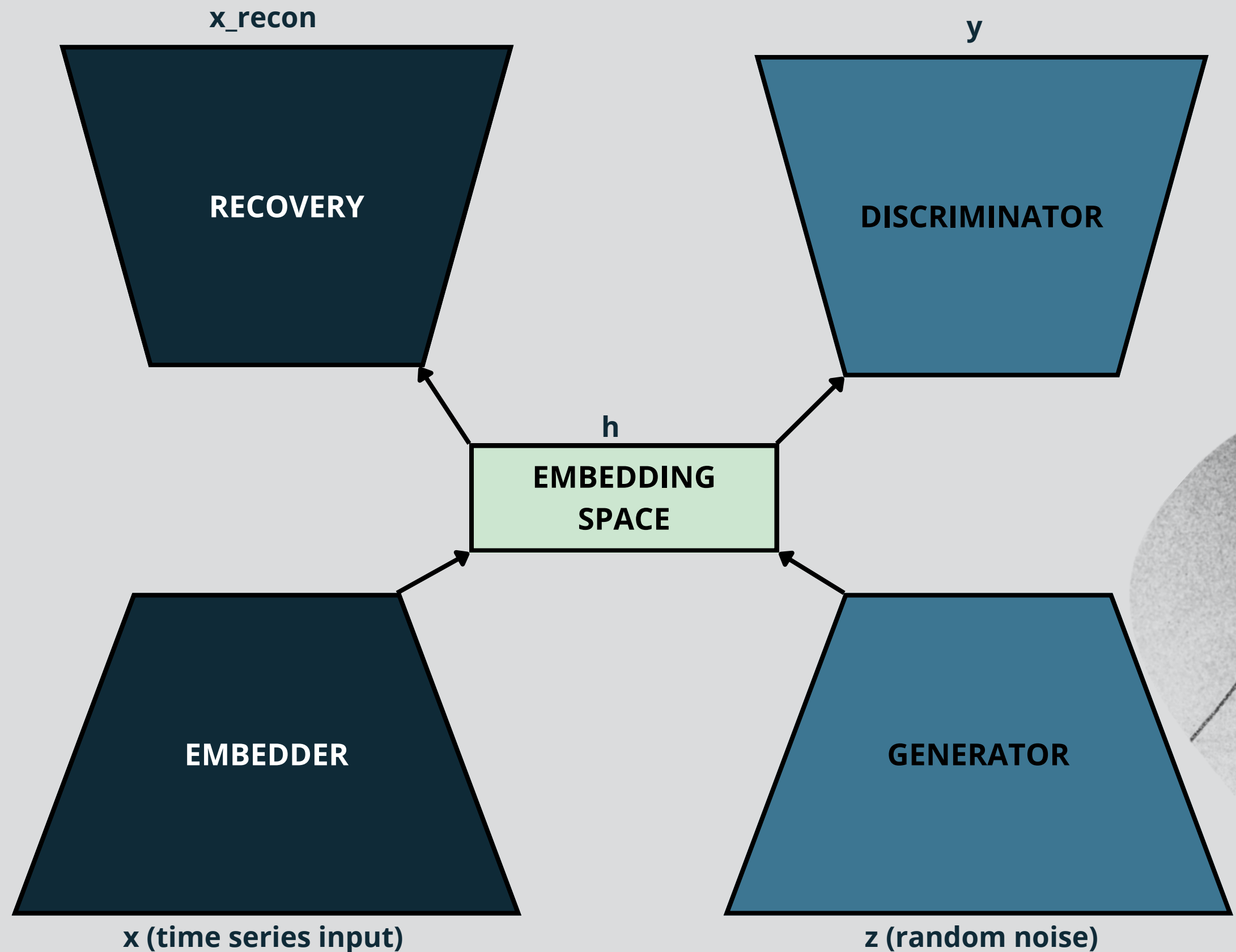
Learn a meaningful latent representation of the real time series.

2. Supervised temporal learning:

Capture the temporal relationships between consecutive observations.

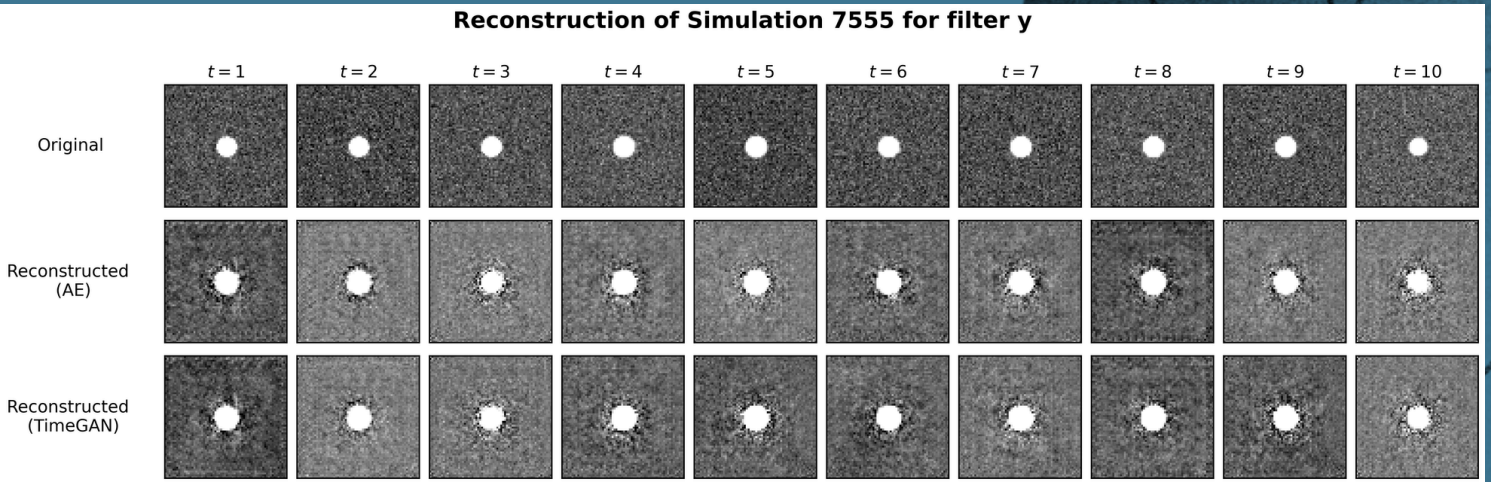
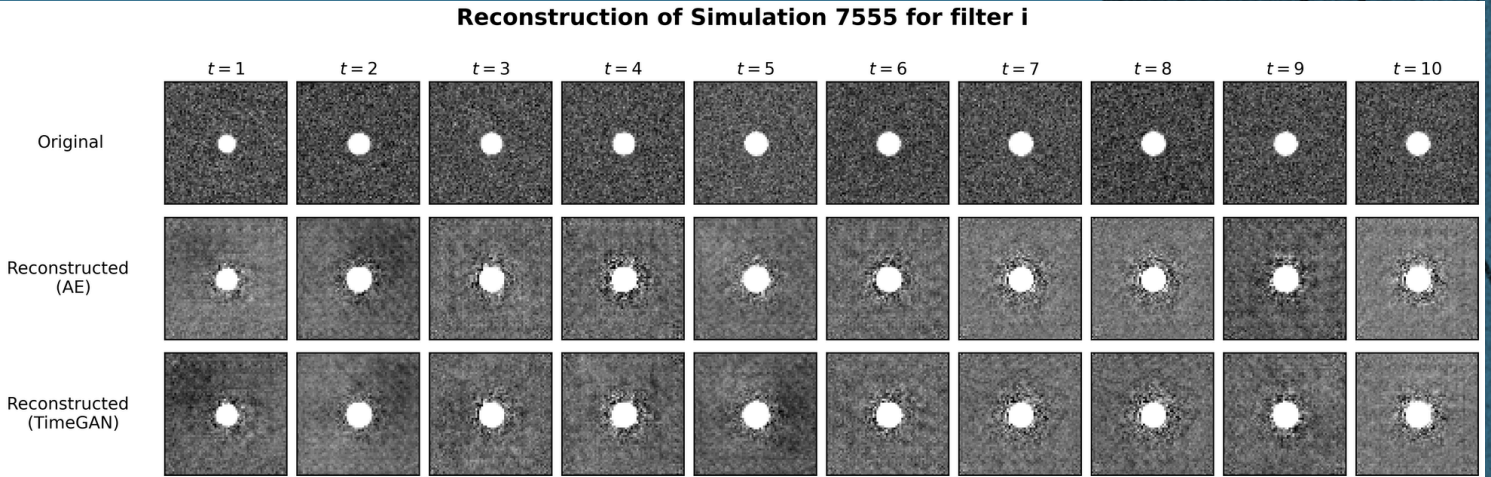
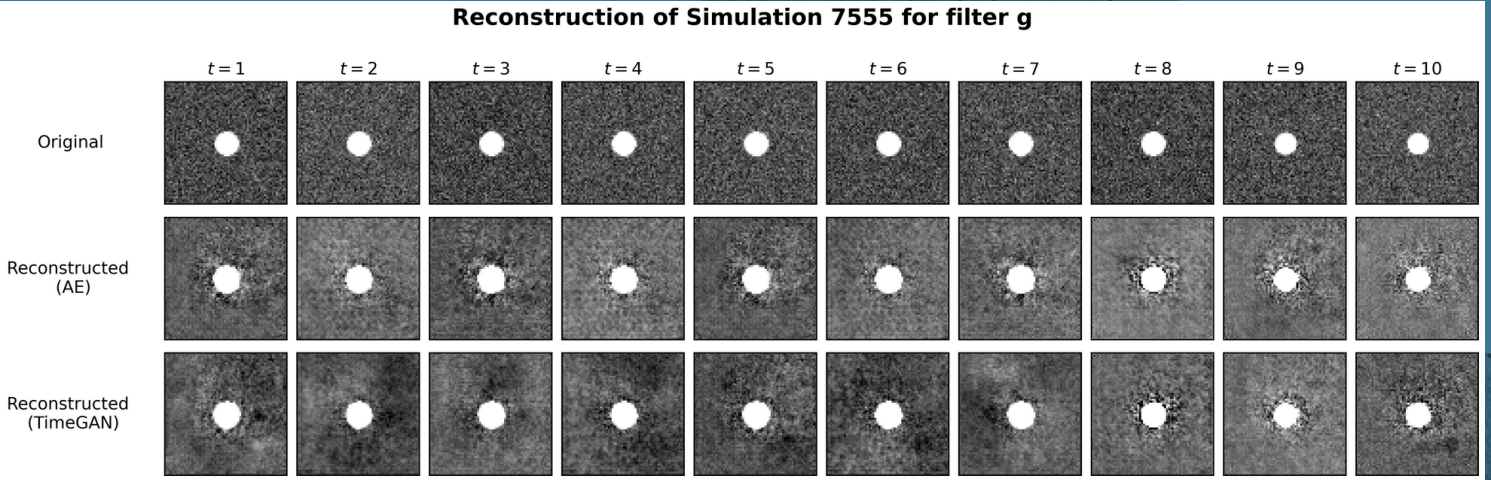
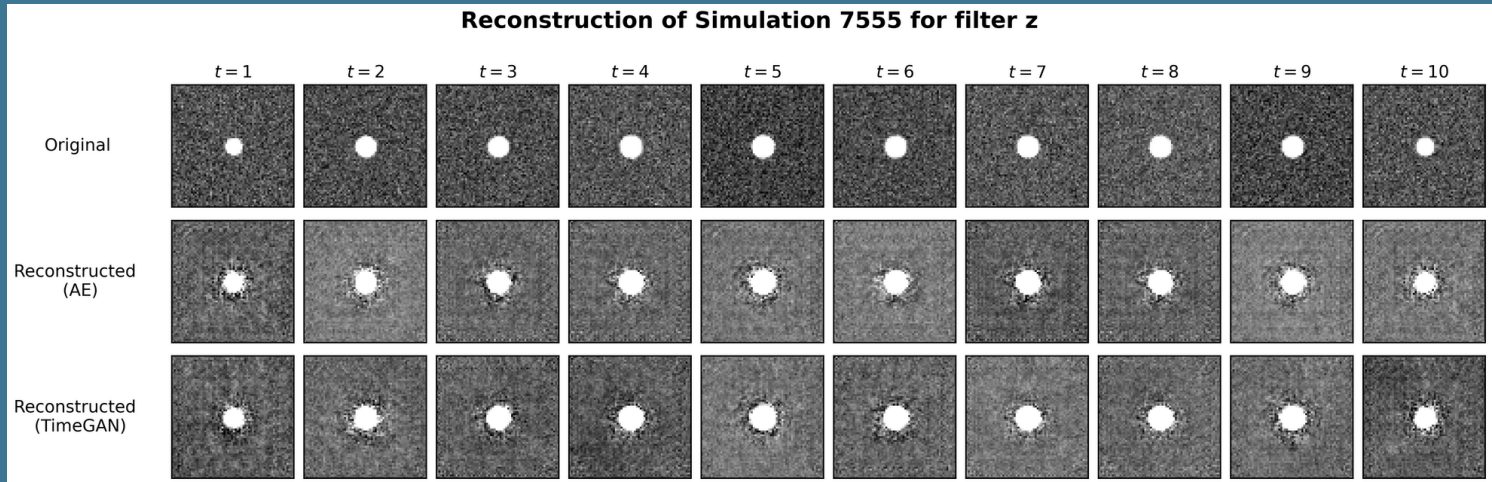
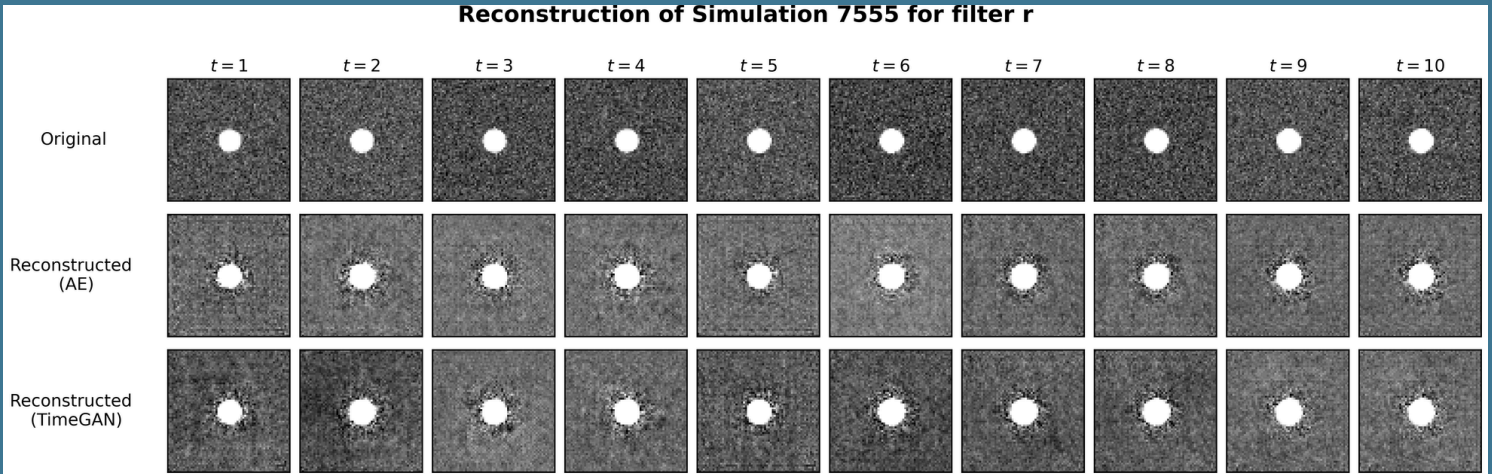
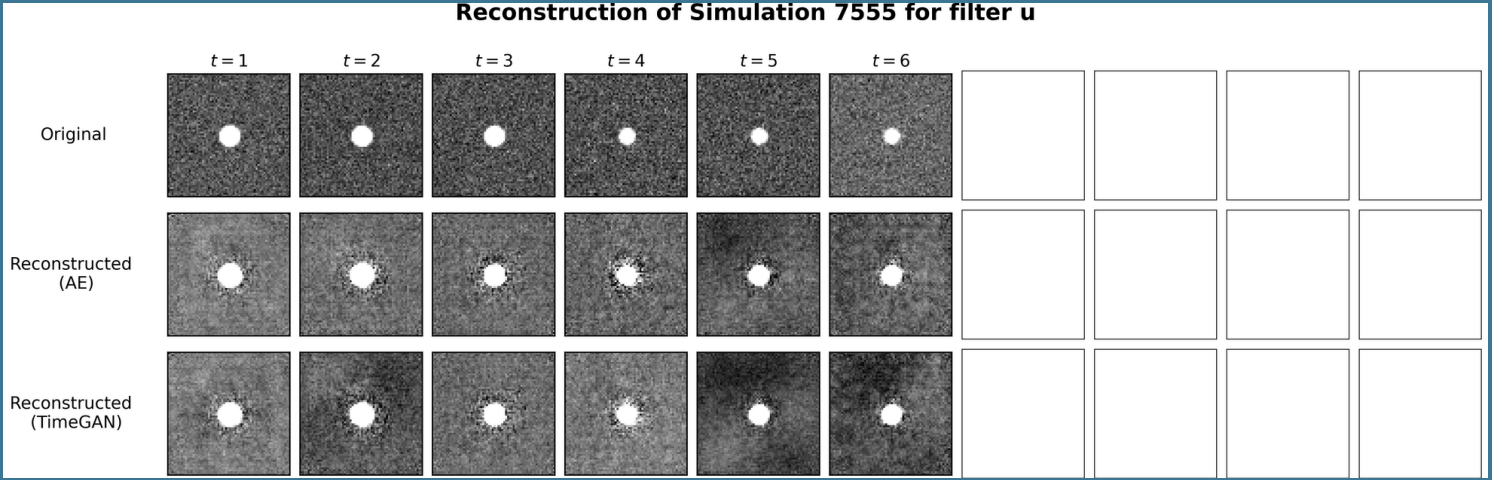
3. Adversarial learning:

Generate synthetic sequences that are difficult for the discriminator to distinguish from real sequences.



VISUALIZATION

RECONSTRUCTION RESULTS



STATISTICS

RECONSTRUCTION RESULTS

Metrics are computed over the complete temporal sequence for each sample and filter, then averaged across the dataset.

METRICS/ MODEL	MSE ↓	RMSE ↓	MAE ↓
AE	0.50	0.65	42.74
TIMEGAN	0.01	0.09	3.02

$$MSE = \frac{1}{n} \sum_{i=1}^n (x_i - x_i^{recon})^2$$
$$RMSE = \sqrt{MSE}$$
$$MAE = \frac{1}{n} \sum_{i=1}^n (|x_i - x_i^{recon}|)$$

MSE: Mean squared Error, Average squared reconstruction error
RMSE: Reconstruction error in the original data units
MAE: Average absolute reconstruction error

STATISTICS

RECONSTRUCTION RESULTS

Metrics per filter

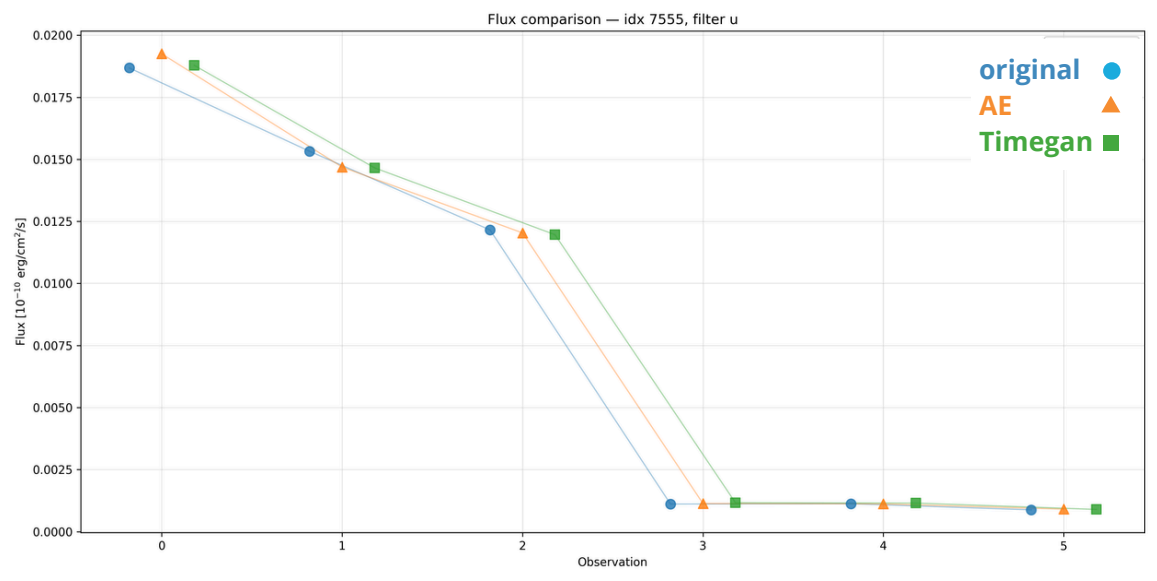
	(AE) MSE ↓	(TIMEGAN) MSE ↓	(AE) RMSE ↓	(TIMEGAN) RMSE ↓	(AE) MAE ↓	(TIMEGAN) MAE ↓
u	0.593	0.028	0.770	0.168	38.118	4.106
g	0.415	0.0194	0.644	0.139	40.677	4.060
r	1.088	0.004	1.043	0.065	62.117	2.135
i	0.562	0.006	0.749	0.080	58.170	3.226
z	0.389	0.003	0.623	0.060	54.940	2.217
y	0.005	0.005	0.076	0.076	2.416	2.407

VISUALIZATION

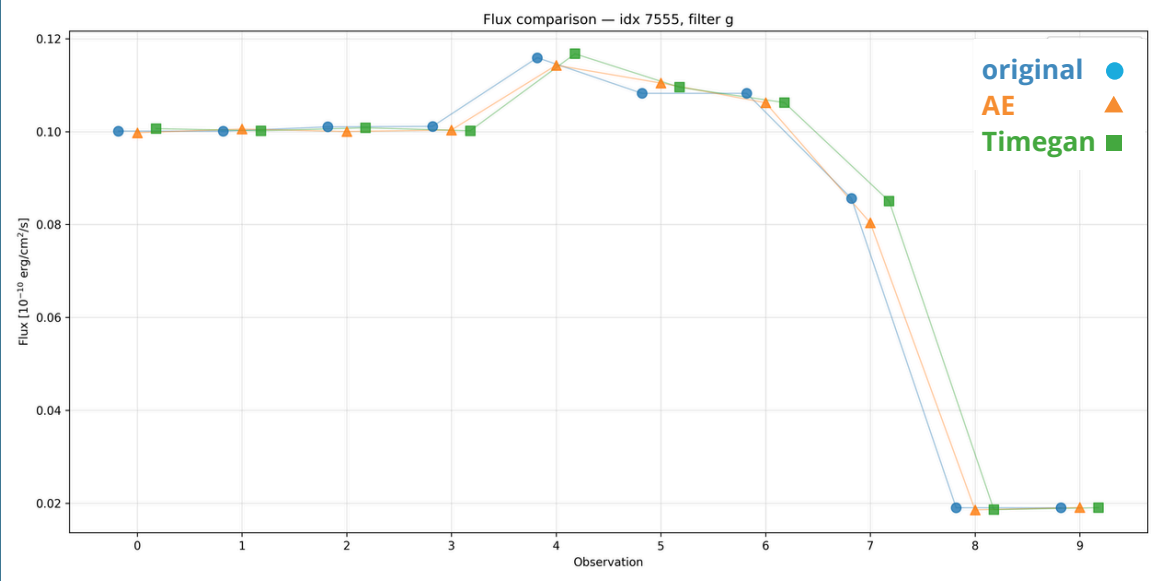
RECONSTRUCTION RESULTS

The reconstruction reproduces the evolution of the PSF, capturing not only spatial variations but also **variations in amplitude**.

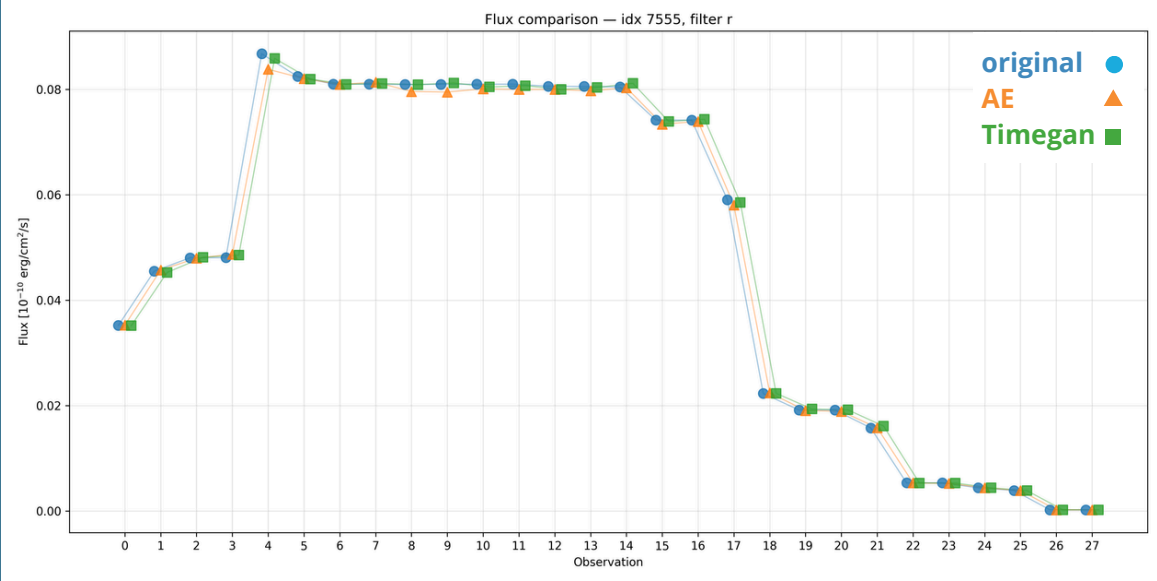
u



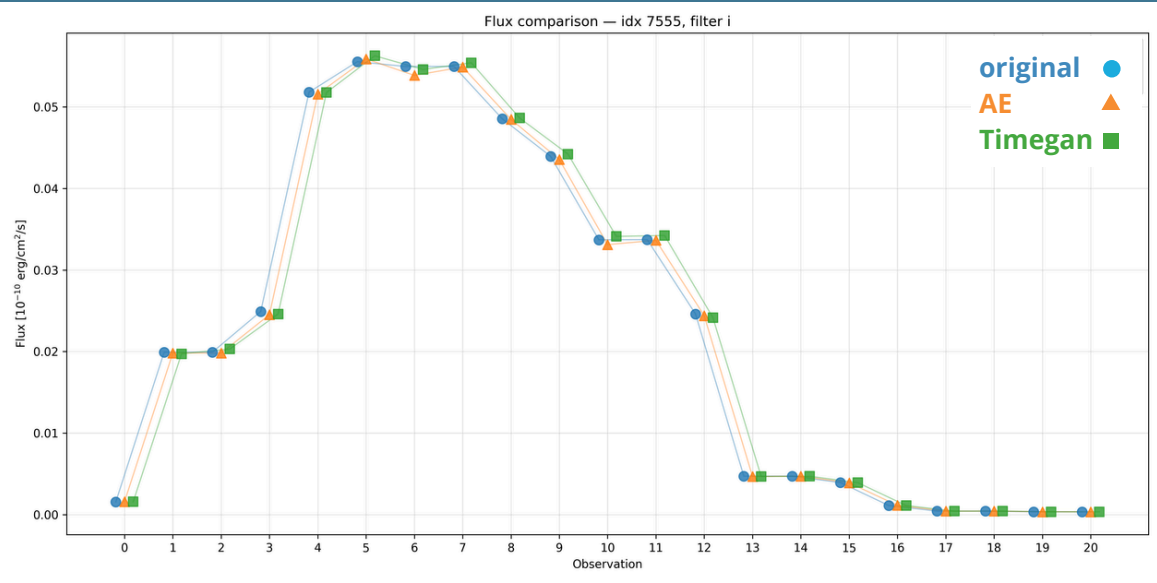
g



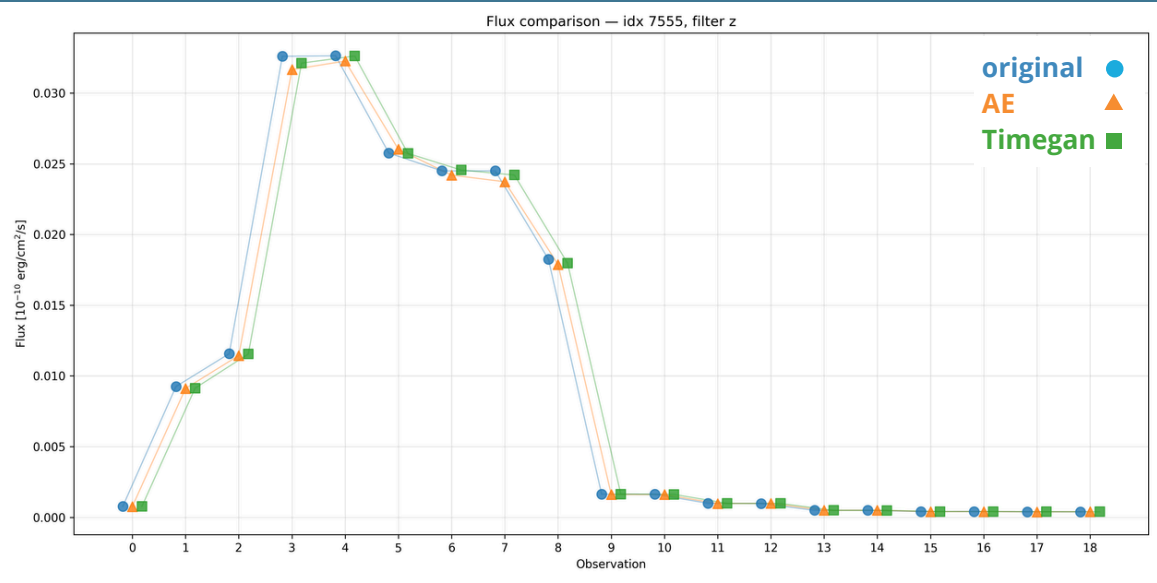
r



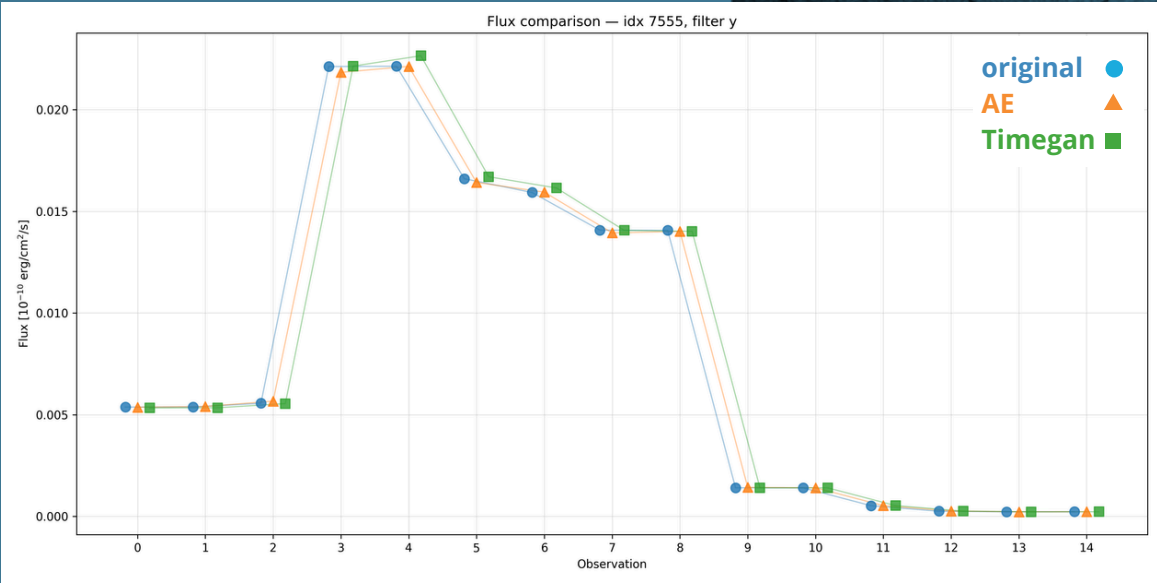
i



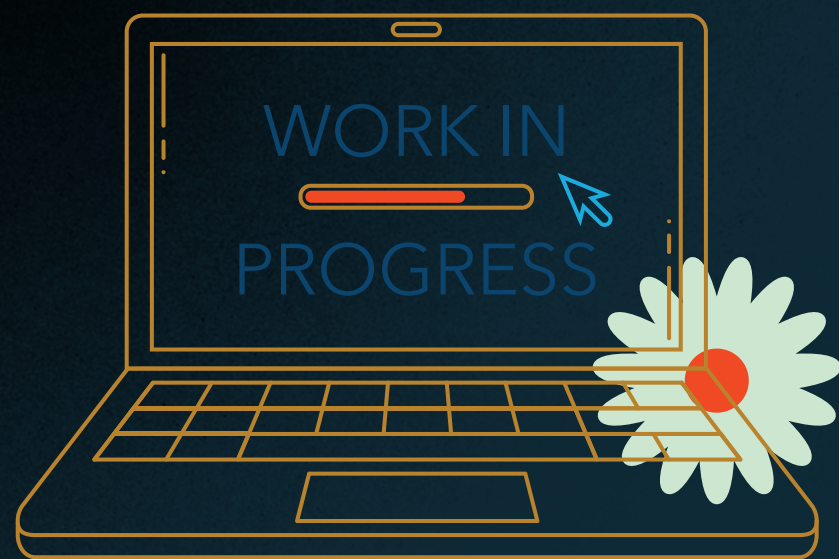
z



y

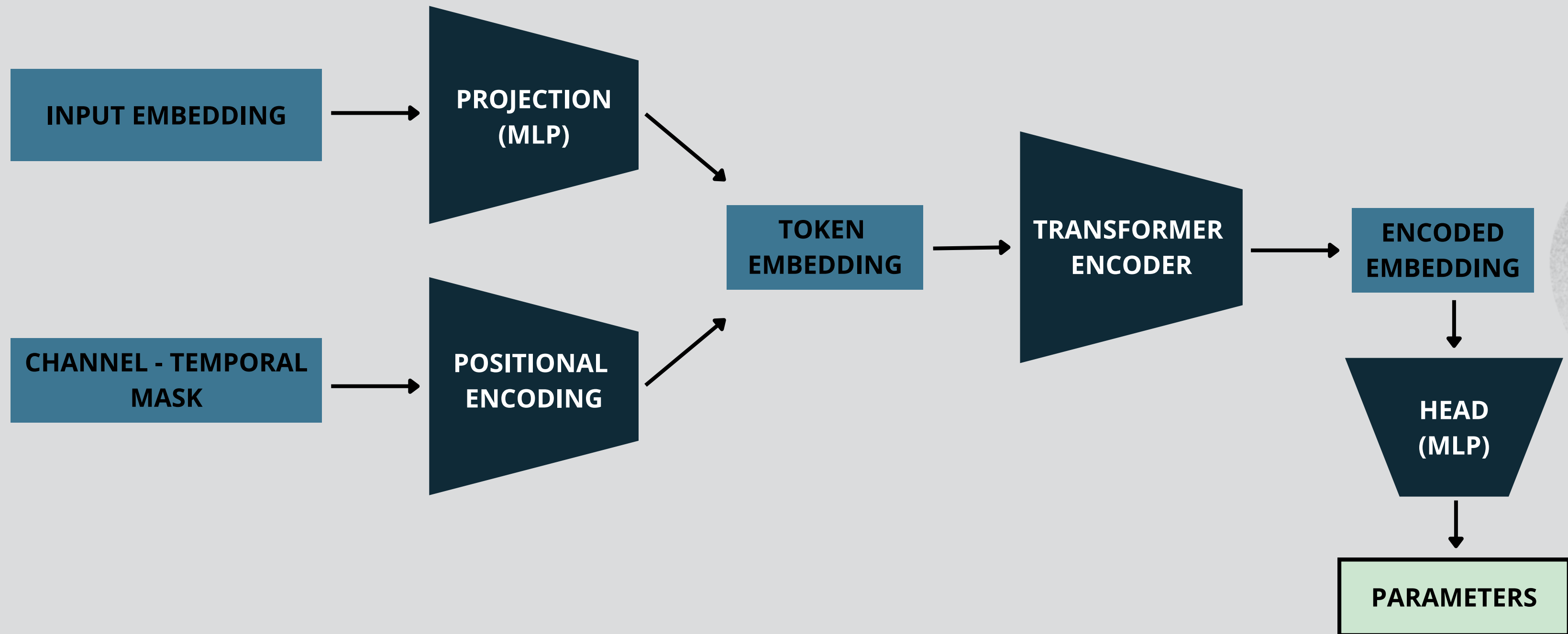


PART II: REGRESSION



REGRESSION MODEL

The model uses a masked **Transformer encoder** to capture temporal and channel-wise dependencies, combining the learned representation with the fraction of valid observations and optional conditional inputs for regression.



STATISTICS

REGRESSION RESULTS

REDSHIFT REGRESSION

METRICS / MODEL	MSE ↓	STD ↓	MAD ↓	NMAD ↓	R ² ↑
AE	0.056	0.235	0.109	0.162	0.938
TIMEGAN	0.055	0.233	0.097	0.144	0.939

- MSE:** Average squared difference between predictions and true values
- STD:** Variability/spread of the prediction errors
- MAD:** Average absolute deviation between predictions and true values
- NMAD:** normalized measure of the deviation
- R²:** how much of the variance in the target is explained by the model

$$MSE = \frac{1}{n} \sum_{i=1}^n \left(y_i - y_i^{pred}\right)^2$$

$$STD = \sqrt{\frac{1}{n} \sum_{i=1}^n \left(e_i - e_i^{average}\right)^2}, e_i = y_i - y_i^{pred}$$

$$MAD = \frac{1}{n} \sum_{i=1}^n \left(|y_i - y_i^{pred}|\right)$$

$$NMAD = 1.4826 \times median(e_i - median(e))$$

$$R^2 = 1 - \frac{\sum_i \left(y_i - y_i^{pred}\right)^2}{\sum_i \left(y_i - mean(y)\right)^2}$$

CONCLUSION

01 TimeGAN Superiority in
Temporal Reconstruction

02 High-Fidelity Redshift
Estimation

03 Physical Progenitor Parameter
Degeneracy

directly constraining other deep
physical progenitor parameters
remains unfeasible

04 A Scalable Paradigm for the LSST Era

the general embedding representation
offers a highly scalable, automated
framework capable of processing millions
of transient events in future wide-field
surveys like LSST

FUTURE STEP

01

Temporal Extension of Phase Coverage

Remove the initial 30-epoch constraint on the light curves to allow the model to capture late-time physics and post-maximum photometric evolution.

02

Cross-Validation with Observational Data

Transition from fully synthetic LSST-mock datasets to real observed multi-band supernova events to evaluate the model's performance under authentic noise and cadence conditions.

03

Expansion of the Progenitor Parameter Space

Broaden the grid of simulated stellar evolution models to map a more comprehensive parameter space, aiming to resolve the degeneracies in mass-loss history

04

Embedding Generalization for Downstream Tasks

Test the pre-trained TimeGAN embeddings on alternative astrophysical subtasks, such as transient classification, anomaly detection, and early-warning forecasting

THANK YOU!



Email me at maria.zampella@unina.it
if you have more questions.

BACKUP

MODELS INFO

Information about the models used in this study.

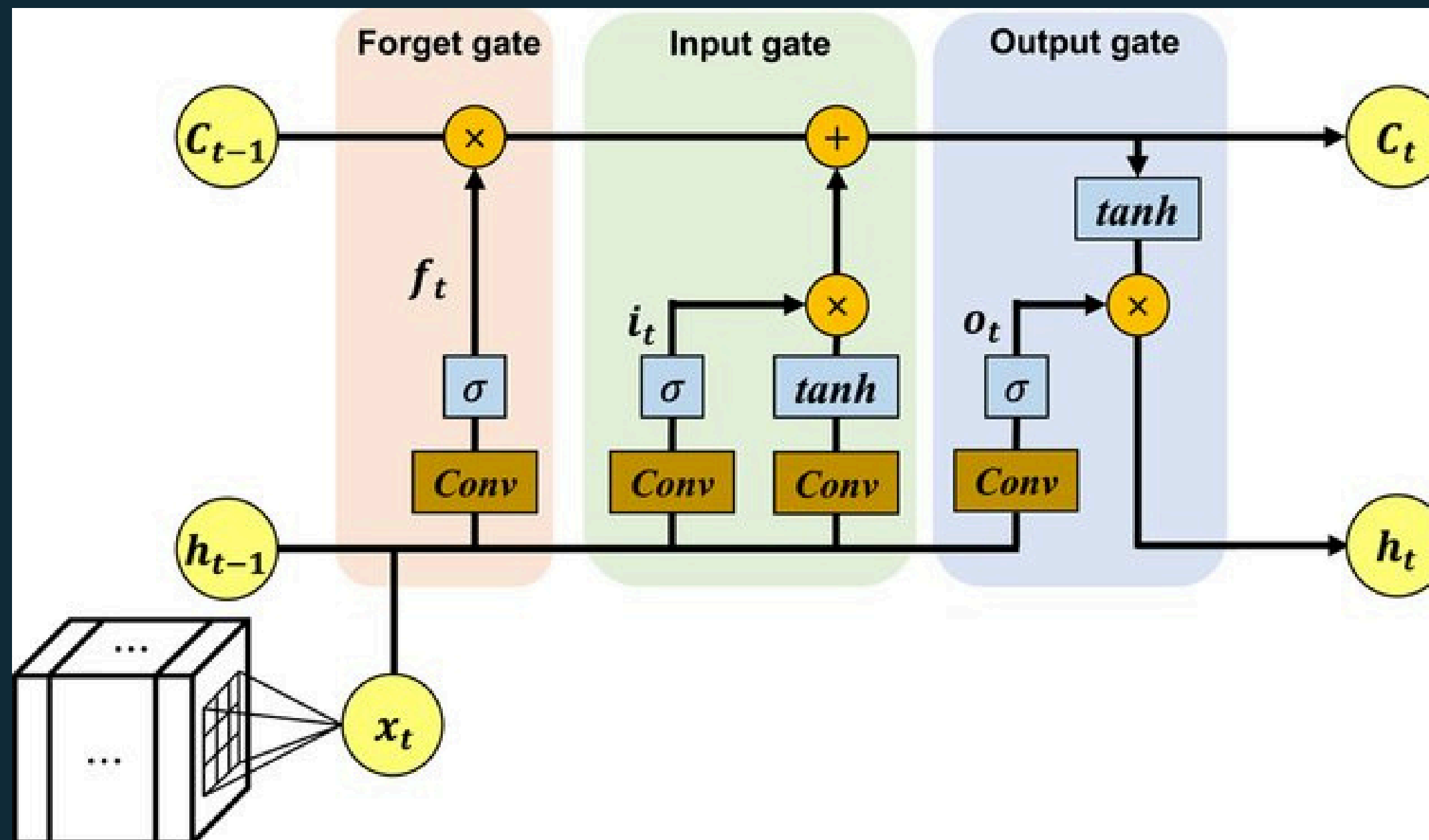
	Total training time	Inference / Val Dataset	Inference / Sample	Parameters
AE	1h30m	6.09 s	6.8 ms	2.258.625
TimeGAN	8h30m	6.18 s	6.9 ms	2.646.082
Regression	1h40m	0.78 s	0.52 ms	23.658

- Number of training epoch = 200
- Batch size = 64
- Val dataset: 1500 samples
- GPU: NVIDIA H100
- Number of GPUs: 1
- Framework: pytorch

BACKUP

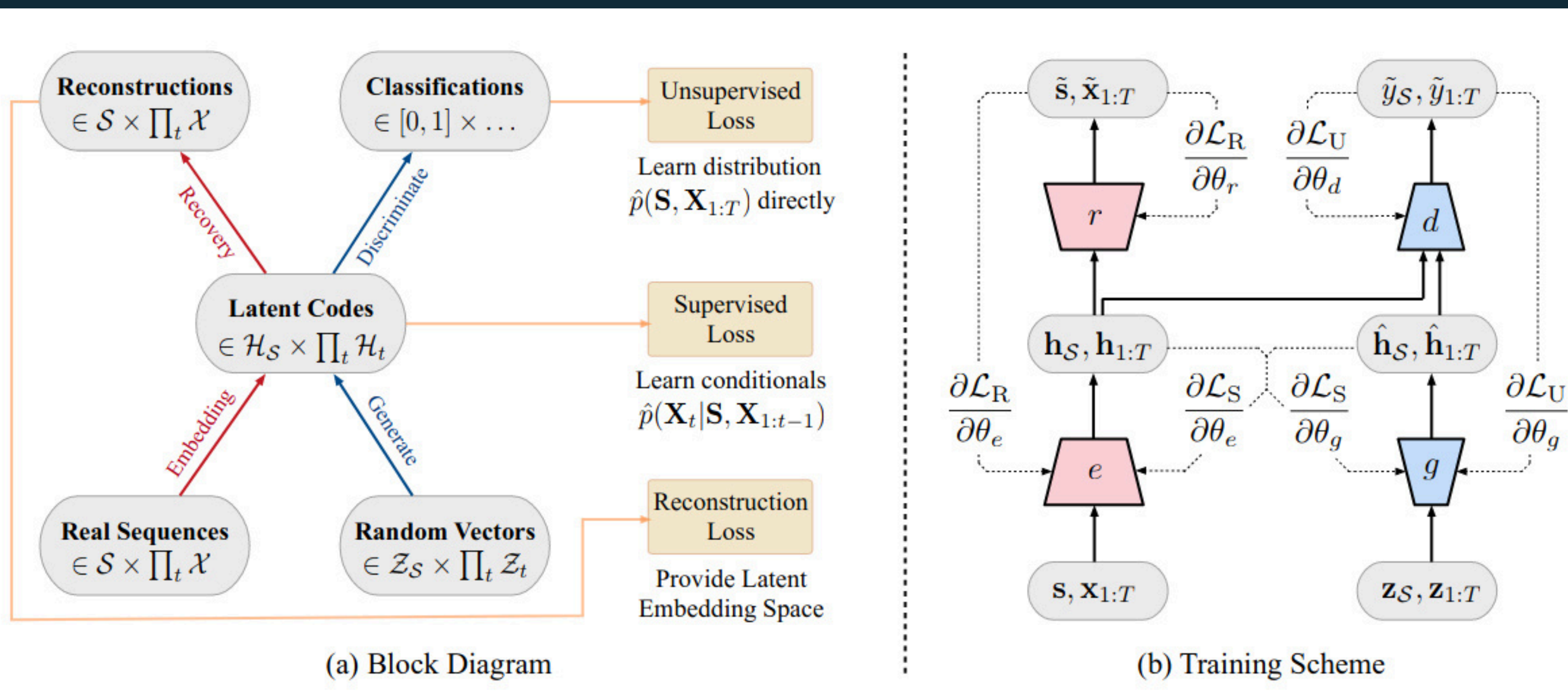
CONV-LSTM SCHEME

We use Recurrent Neural Network layers in our model for taking into account the temporal dimension



BACKUP

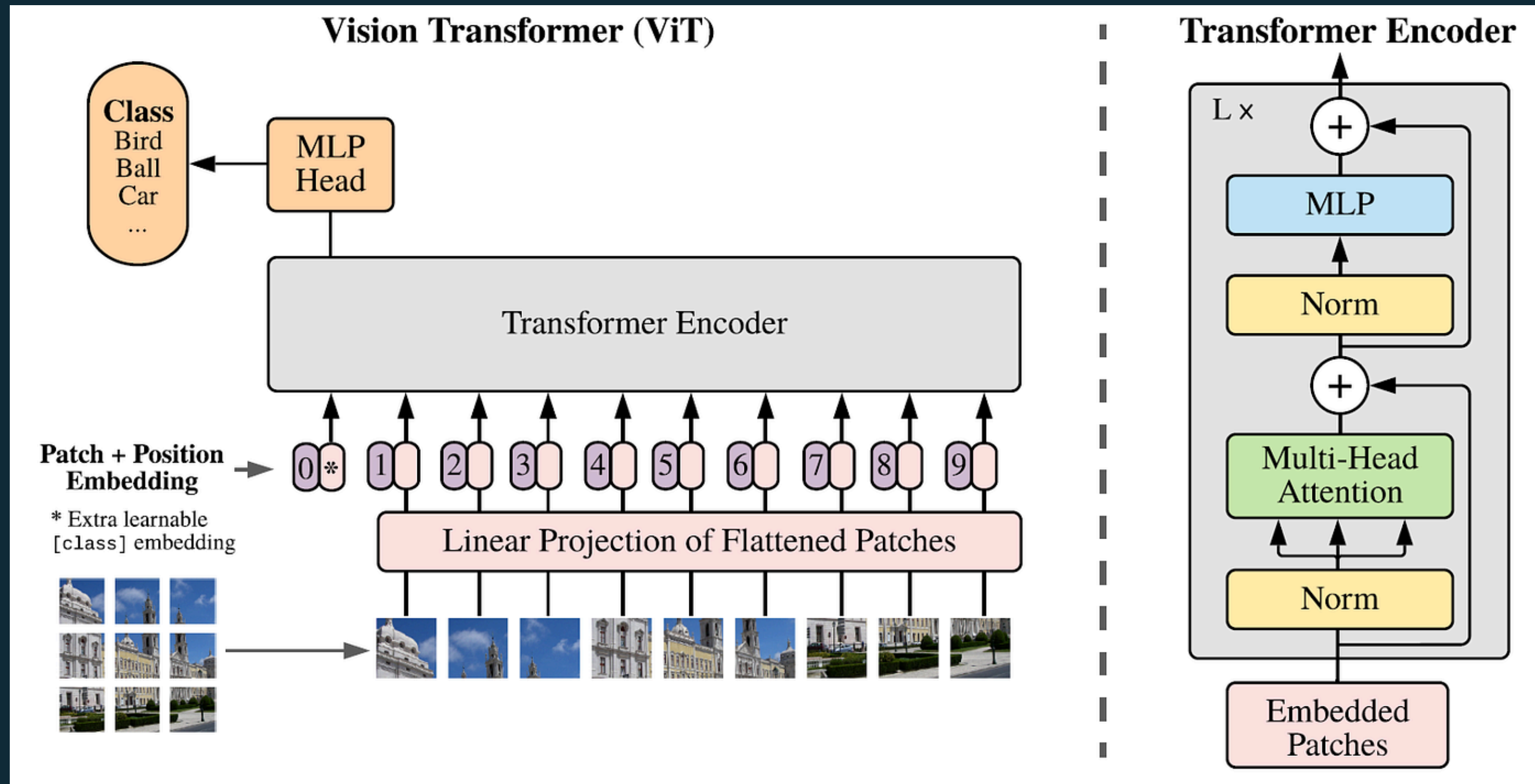
TIMEGAN TRAINING SCHEME



$$\mathcal{L}_S = \mathbb{E}_{\mathbf{s}, \mathbf{x}_{1:T} \sim p} \left[\sum_t \|\mathbf{h}_t - g_{\mathcal{X}}(\mathbf{h}_S, \mathbf{h}_{t-1}, \mathbf{z}_t)\|_2 \right]$$

BACKUP

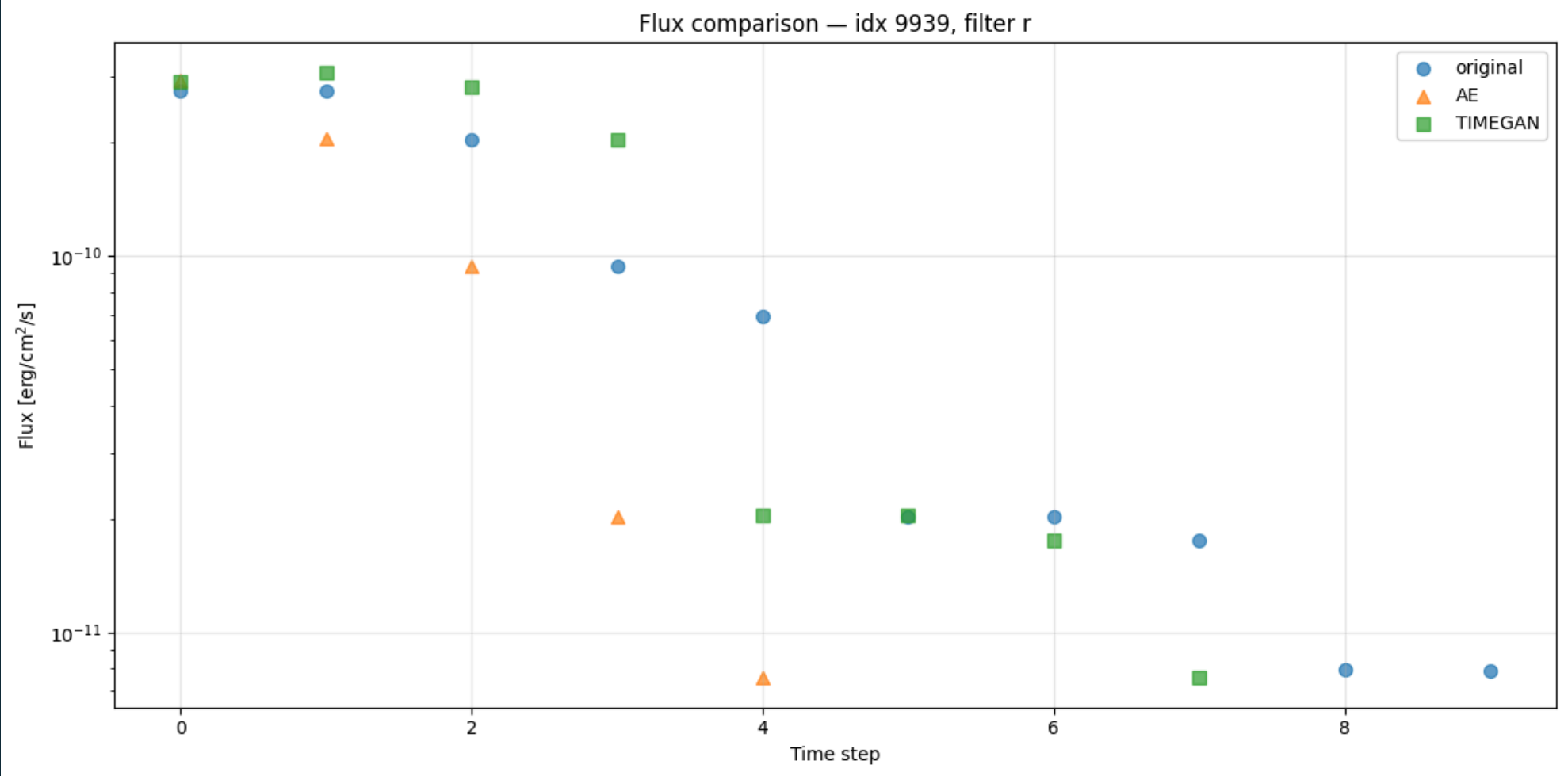
VISION TRANSFORMER ARCHITECTURE



BACKUP

RECONSTRUCTION RESULTS – LIGHT CURVES

Filter	MAE ↓ (AE)	MAE ↓ (TIMEGAN)	NMAE ↓ (AE)	NMAE ↓ (TIMEGAN)
u	1.44e-13	1.25e-13	6.49	5.15
g	2.09e-13	1.37e-13	3.61	2.60
r	2.06e-12	1.61e-12	162.50	156.89
i	1.77e-12	9.87e-13	60.81	37.34
z	1.38e-12	2.17e-12	51.77	48.69
y	1.98e-12	2.23e-12	92.94	89.68



Example of reconstruction in which TimeGAN reconstruction is better than AE

$$MAE = \frac{1}{n} \sum_{i=1}^n (|x_i - x_i^{recon}|)$$

$$NMAE = \frac{MAE}{mean(x_i)} \times 100$$

AE

Variable	MSE ↓	STD ↓	MAD ↓	NMAD ↓	R2 ↑
EBV	0.99475	0.997348	0.864344	1.281476	0.00525
M_PROG	0.95588	0.977495	0.842792	1.249523	0.04412
M_NI	0.820855	0.905992	0.755074	1.119473	0.179145
MASSLOSS	0.964343	0.981739	0.846816	1.25549	0.035657
BETA	0.991985	0.995977	0.858041	1.272131	0.008015
RCSM	0.954771	0.977039	0.8417	1.247904	0.045229
E_EXP	0.684402	0.827274	0.678863	1.006482	0.315598

BACKUP

REGRESSION RESULTS –
ALL PARAMETERS

TIMEGAN

Variable	MSE ↓	STD ↓	MAD ↓	NMAD ↓	R2 ↑
EBV	0.994659	0.997325	0.864114	1.281135	0.005341
M_PROG	0.962044	0.980196	0.845889	1.254116	0.037956
M_NI	0.840387	0.915928	0.770493	1.142333	0.159613
MASSLOSS	0.971362	0.985563	0.850415	1.260826	0.028638
BETA	0.993578	0.99678	0.859539	1.274352	0.006422
RCSM	0.981127	0.990518	0.856689	1.270127	0.018873
E_EXP	0.709501	0.842216	0.697644	1.034326	0.290499