University of Glasgow

LINKEOS TECHNOLOGY LTD

A WORLD TOP 100 UNIVERSITY

Deep Learning for Enhanced Muon Tomography Imaging

IOP Nuclear Physics Conference 2026, Brighton

William O'Donnell, David Mahon, Guangliang Yang, Simon Gardner, Richard Tyson

1. MOTIVATION

PROBLEM: Non-Destructive Testing of Built Infrastructure

- It has been widely established that there is a growing amount of **aged, concrete** infrastructure coming to **end of life**.

- However, current NDT techniques are **limited** in establishing high quality reconstructions of concrete interiors.

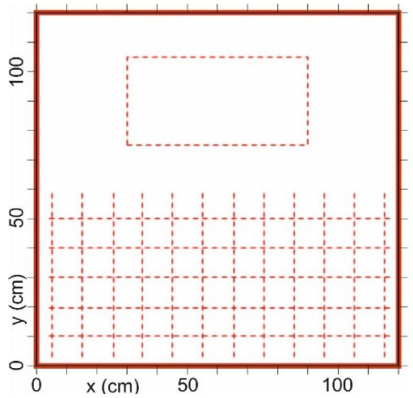
- A 2019 [1] study tested and compared NDT techniques:
  - X-Ray laminography
  - Ground penetrating radar (GPR)
  - Ultrasound
  - **Muography**

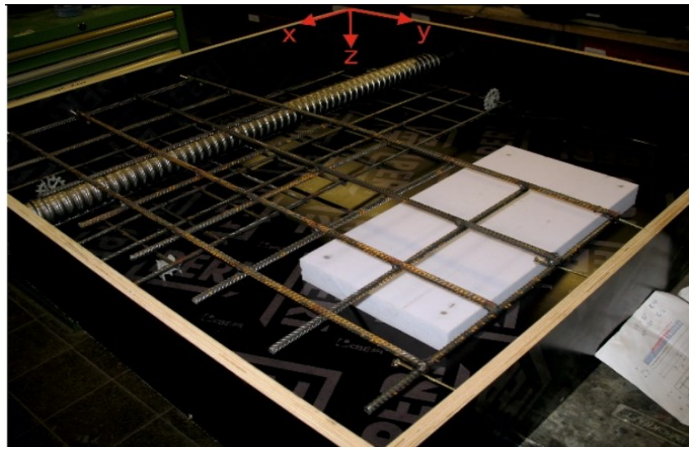
[1] Neiderleithinger et al., 2021. https://doi.org/10.1007/s10921-021-00797-3

University of Glasgow

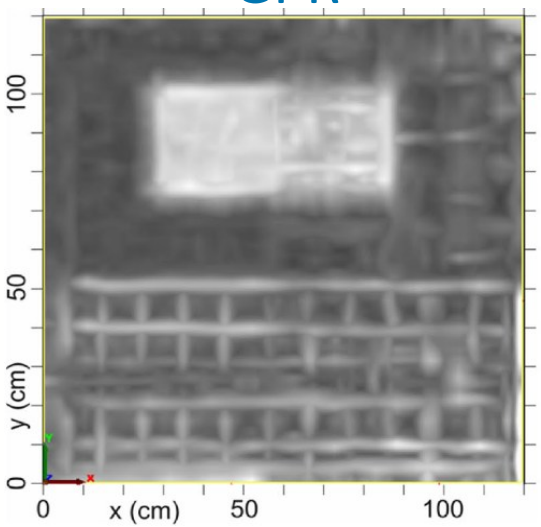
PROBLEM: Non-Destructive Testing of Built Infrastructure

Reference, z = 17cm

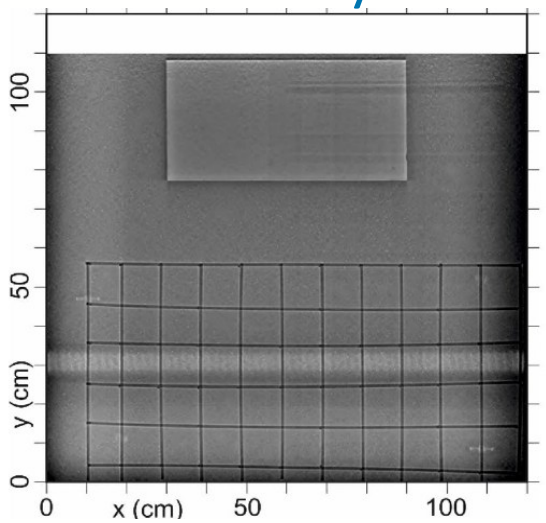




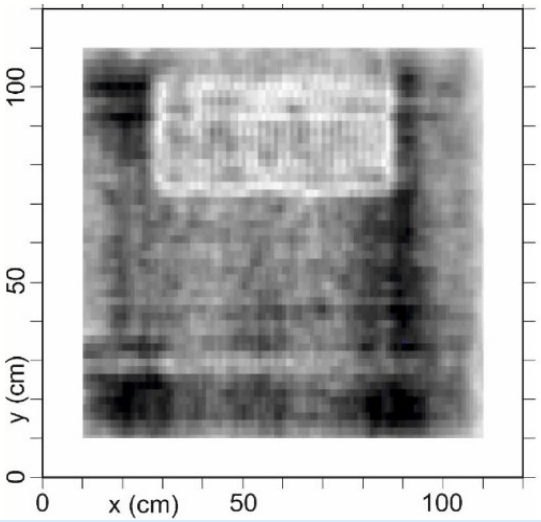
GPR



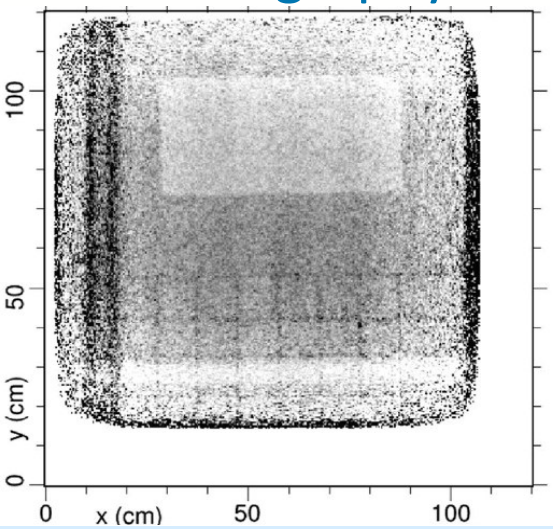
X-Ray



Ultrasound



Muography



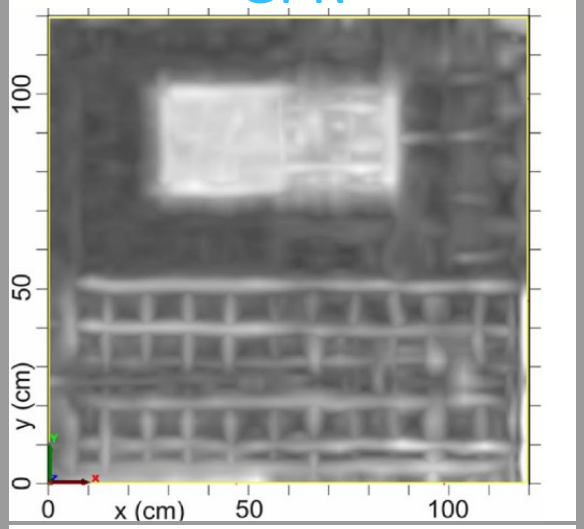
[1] Neiderleithinger et al., 2021. https://doi.org/10.1007/s10921-021-00797-3

University of Glasgow

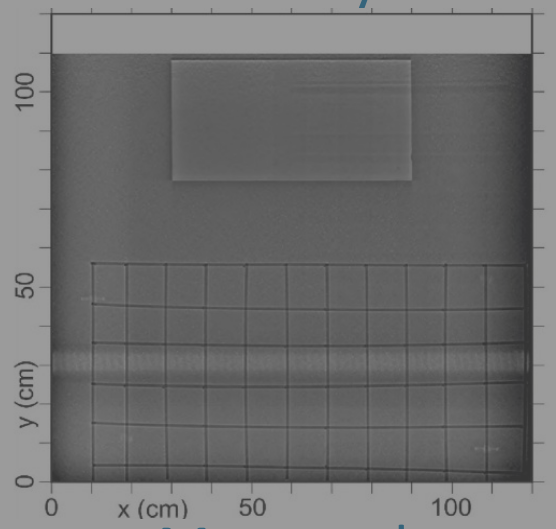
PROBLEM: Non-Destructive Testing of Built Infrastructure

- Poor Resolution
- Must be tuned to certain depths

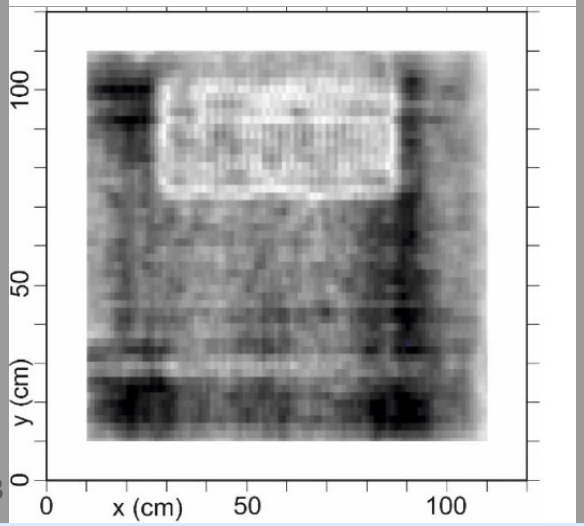
GPR



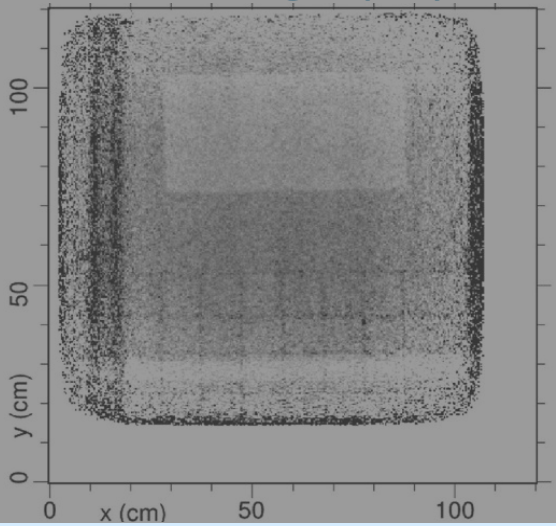
X-Ray



Ultrasound



Muography

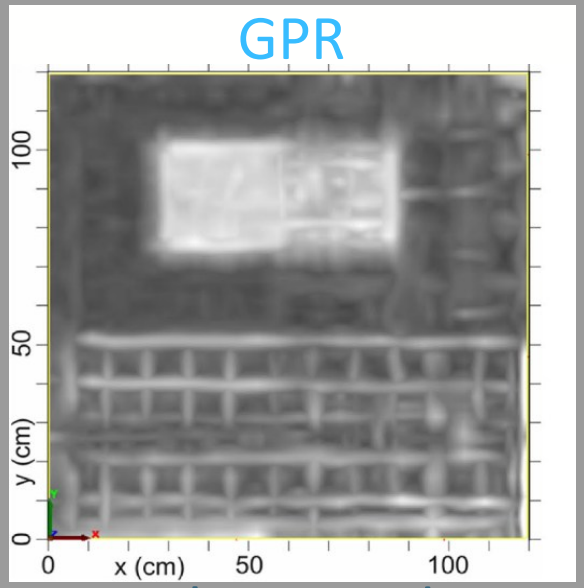


[1] Neiderleithinger et al., 2021. https://doi.org/10.1007/s10921-021-00797-3

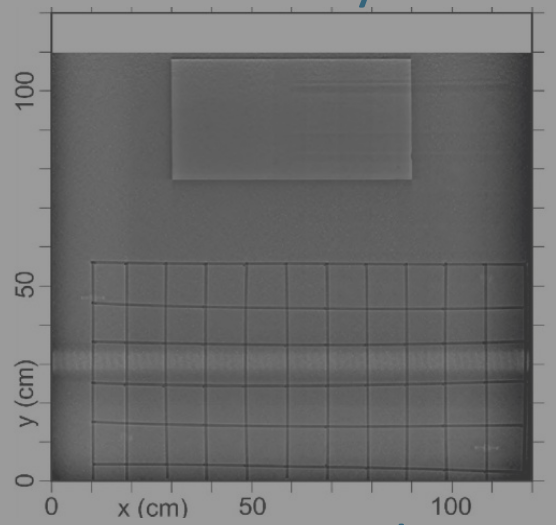
PROBLEM: Non-Destructive Testing of Built Infrastructure

- Object sizes misrepresented

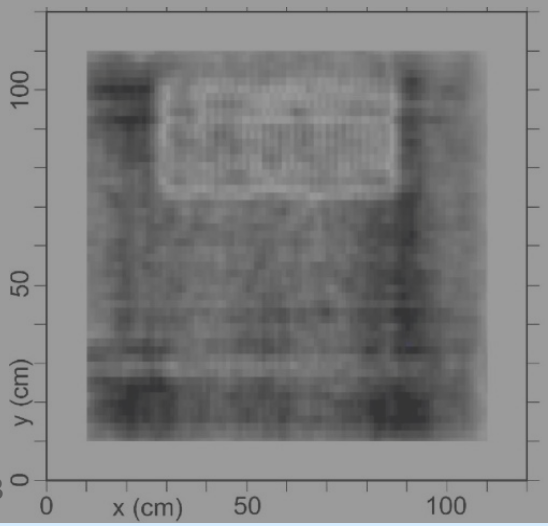
GPR



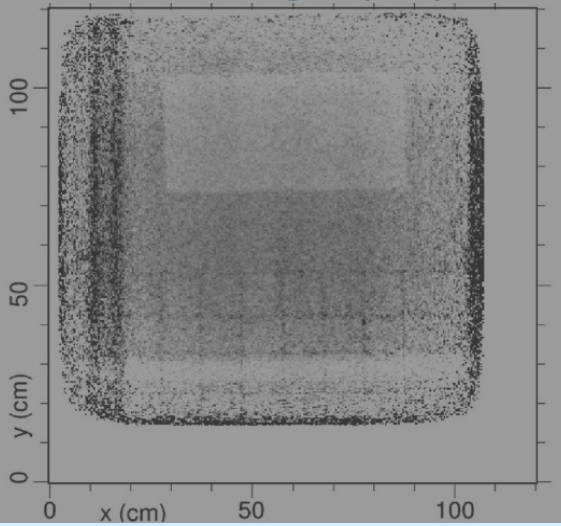
X-Ray



Ultrasound



Muography



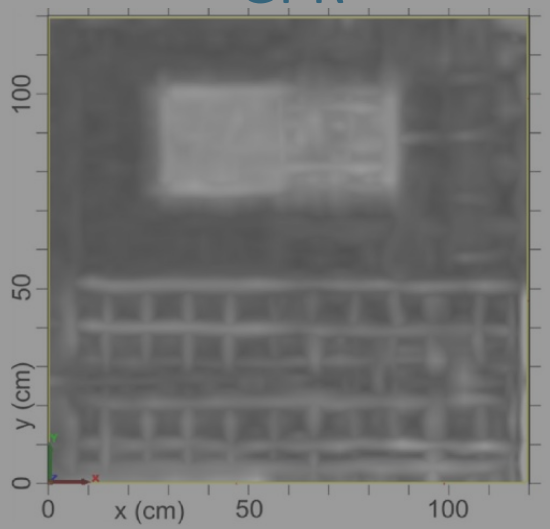
[1] Neiderleithinger et al., 2021. https://doi.org/10.1007/s10921-021-00797-3

University of Glasgow

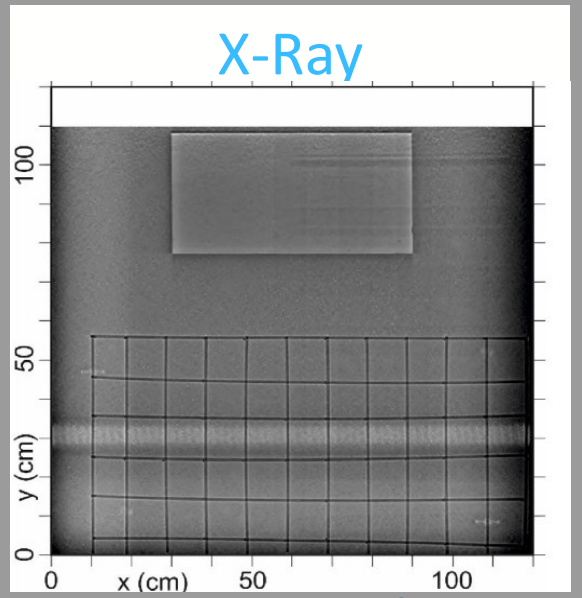
PROBLEM: Non-Destructive Testing of Built Infrastructure

- Health concerns (public use): lots of red tape

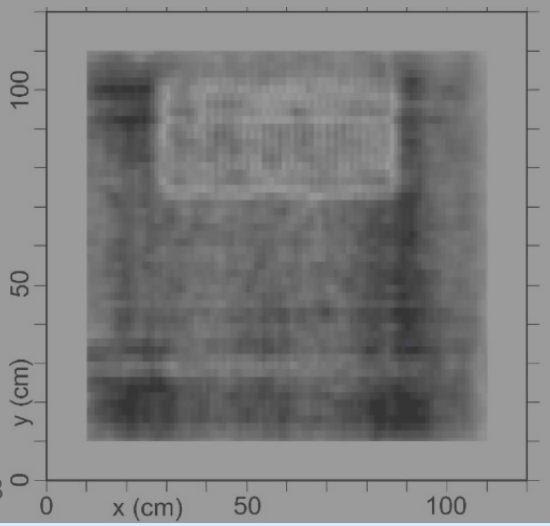
GPR



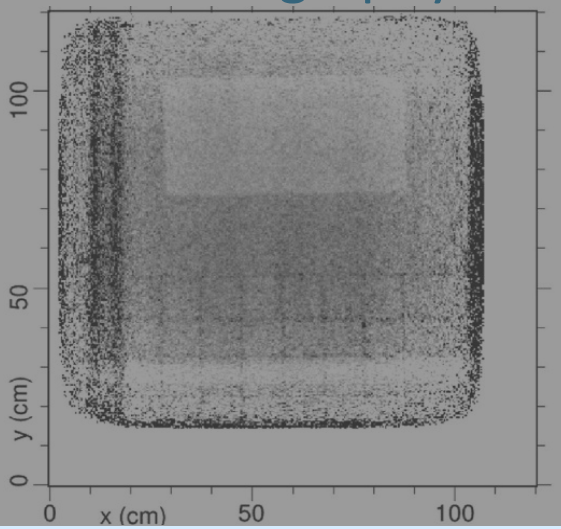
X-Ray



Ultrasound



Muography



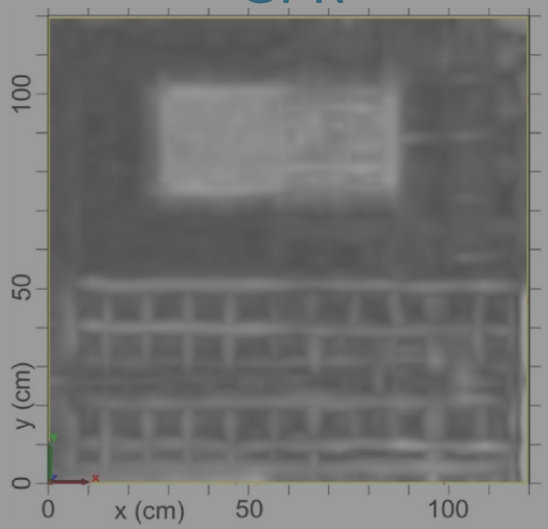
[1] Neiderleithinger et al., 2021. https://doi.org/10.1007/s10921-021-00797-3

University of Glasgow

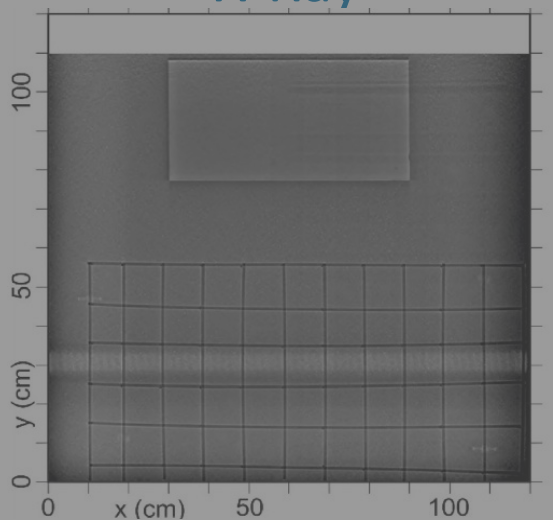
PROBLEM: Non-Destructive Testing of Built Infrastructure

- Good Resolution, at any Depth
- Uses a natural Source

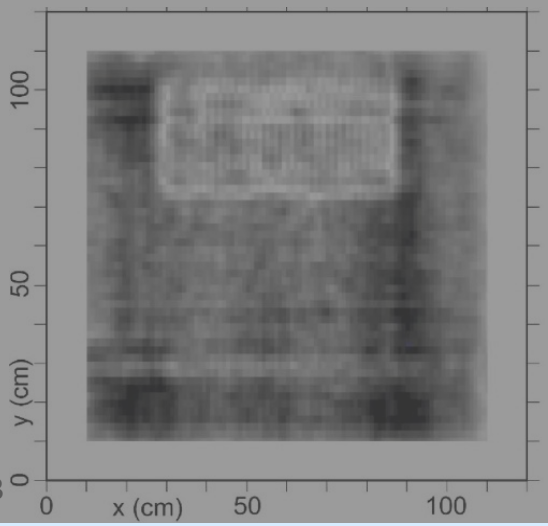
GPR



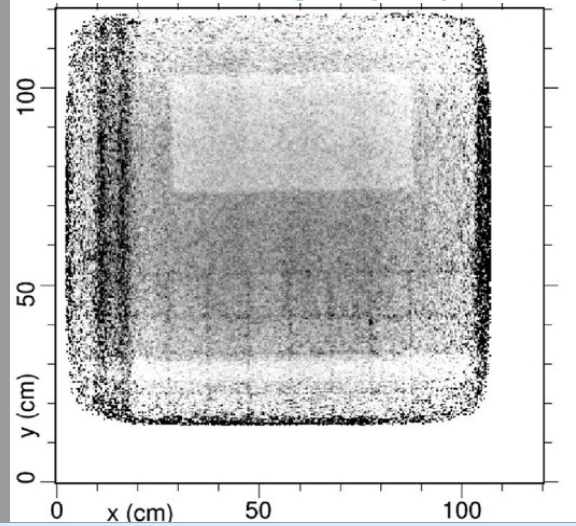
X-Ray



Ultrasound



Muography



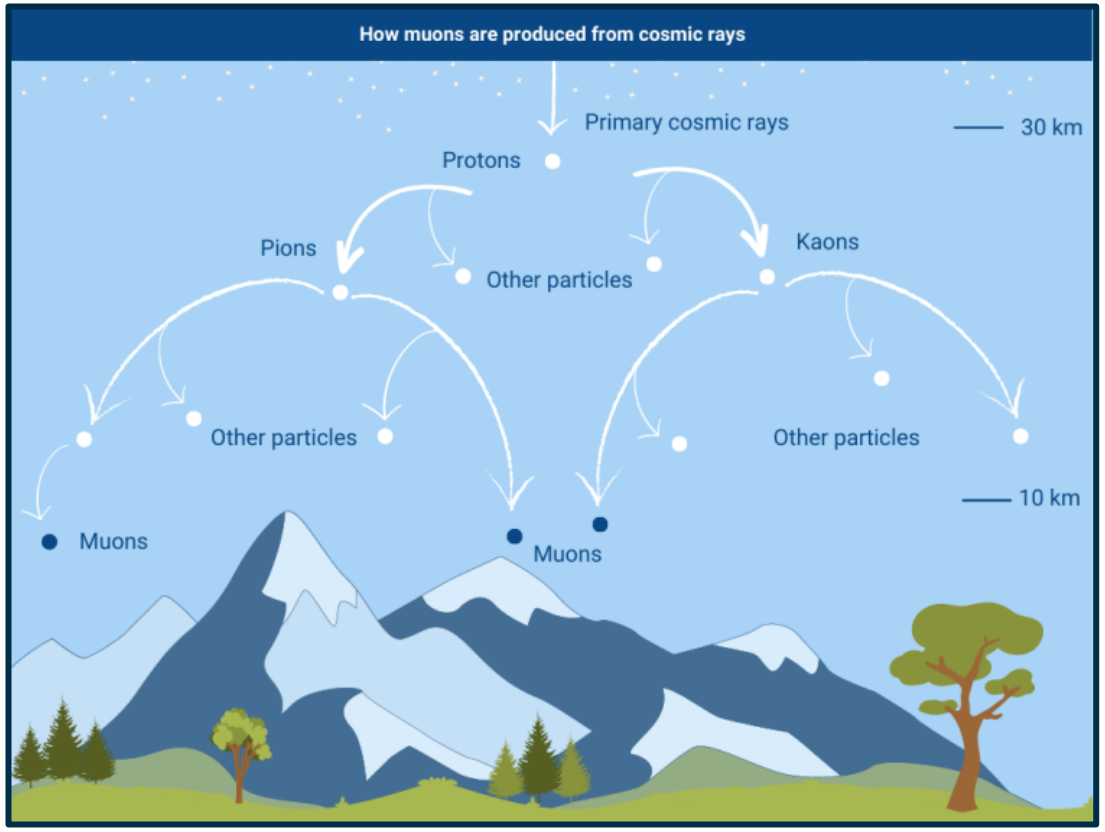
[1] Neiderleithinger et al., 2021. https://doi.org/10.1007/s10921-021-00797-3

What is Muography?

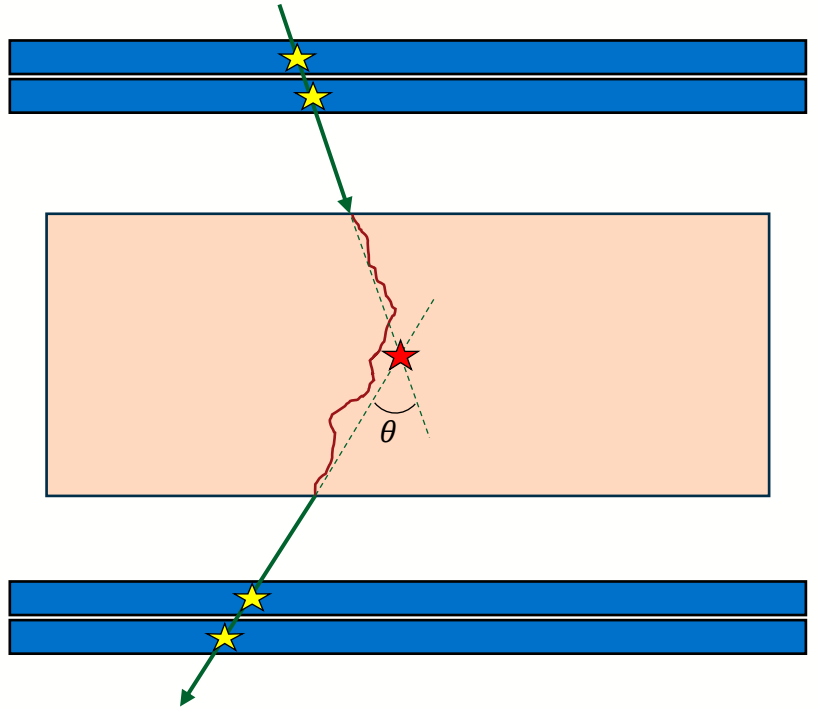
- **Muon tomography** (muography) utilises *naturally occurring*, high energy **cosmic rays.**
  → **Zero radiation risk.**

- They are **highly penetrating (~4GeV)**.
  → *~10,000 times more energy* than X-rays.
  → Can pass through hundreds of metres of rock, or heavy shielding.

- However, at a relatively **low flux ($\mathbf{1cm^{-2}min^{-1}}$)**.
  → Often **~weeks** of data gathering to form a comprehensible image (dependent on the task).



How does Muography Work?

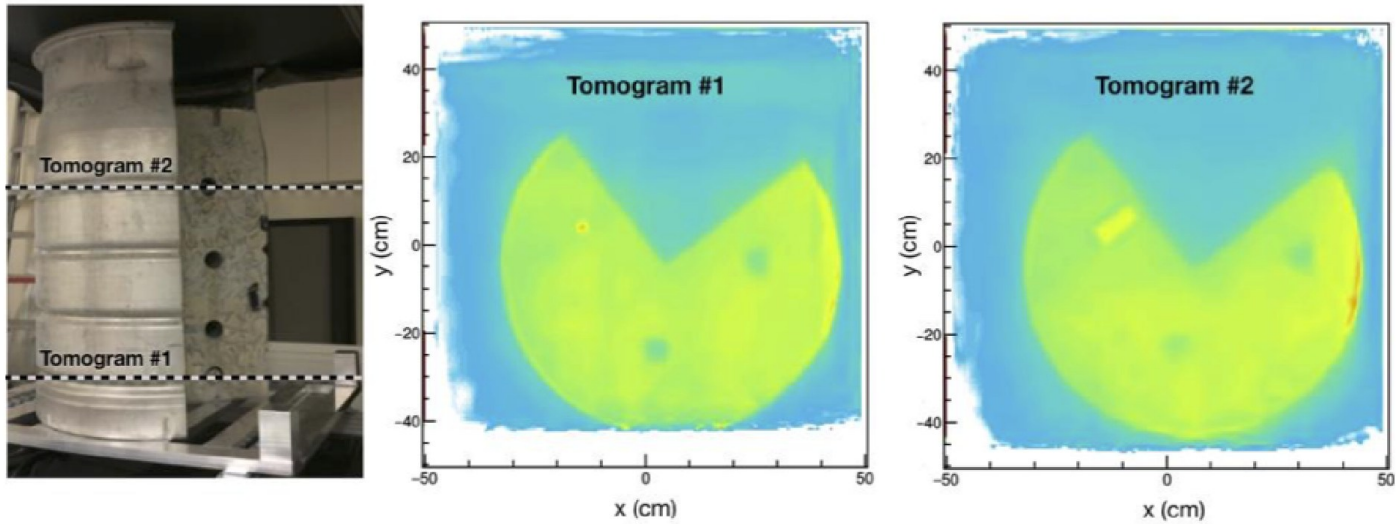


Muon Scattering Tomography:

- Muons interact with matter through **Coulomb scattering**.
- This causes a muon to deflect at an **angle θ**, which is proportional to **material density**.
- We detect an incoming and outgoing track to measure θ.
- This allows us to use reconstruction algorithms to form **3D density maps**.

Applications of muon tomography:

- Nuclear waste characterisation.
- Civil Engineering NDT.
- Cargo scanning.
- Electric arc furnace monitoring,
- (and others).



[2] Mahon et al., 2018. https://doi.org/10.1007/s10921-021-00797-3

Limitations of Muography

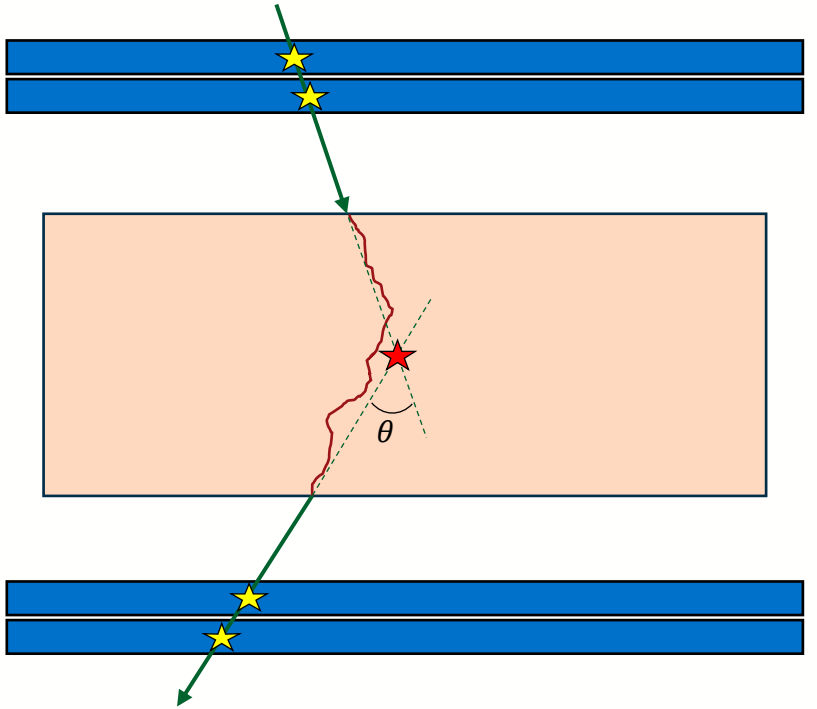
We will be utilising **muon scattering tomography**.

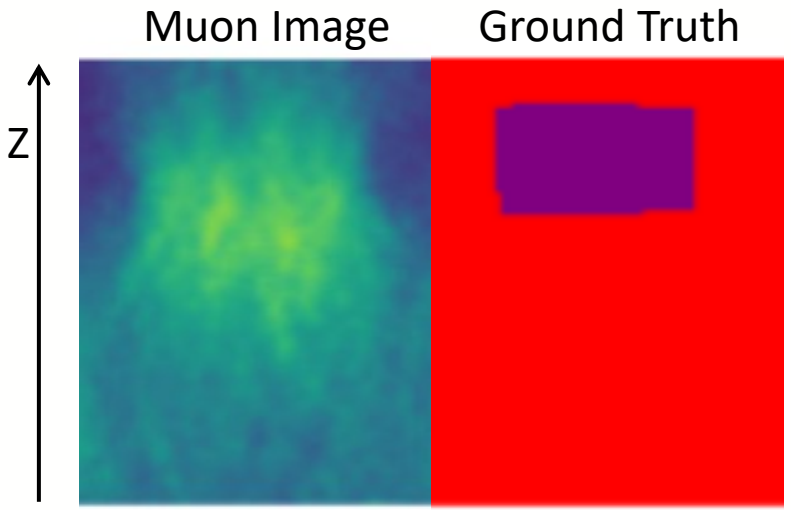
1. Muon imaging time

- Relies on a low **natural** muon flux.
- **Multiple scattering** makes it hard to model the muon path.
- Thus, requires high statistics - so images can take **days to months** to **resolve** features.

2. Z-plane smearing

- Objects **'smear'** in the direction perpendicular to the detector plane, creating **shadows or artefacts**.
- Limited **angular acceptance** ($\pm 30°$) and the **inverse imaging problem** greatly reduce z resolution.



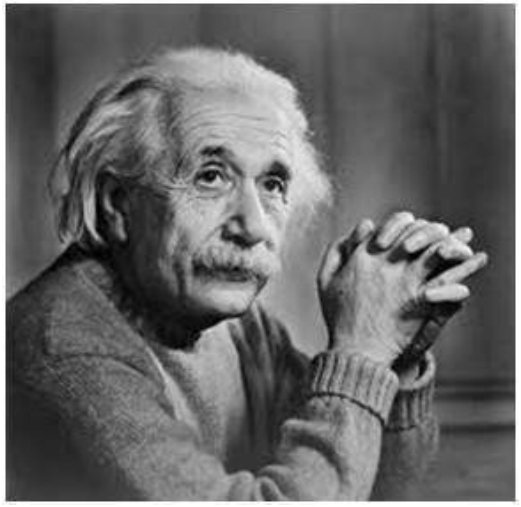


University of Glasgow

2. WHY USE MACHINE LEARNING?

Convolutional Pattern Recognition

- **Convolutional filters** are powerful tools for detecting patterns in data.
- They are **localised**, with typical receptive fields of 3x3 to 5x5 pixels.
- However, these are **user-defined** and limited to high level feature extraction, so **cannot capture complex patterns**.



$\otimes$ Vertical Sobel Filter $\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$

$\otimes$ Horizontal Sobel Filter $\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$

Why Use Machine Learning?

- Instead of a user-defined kernel, **convolutional neural networks** (CNN's) use a data-driven approach to optimise a kernel of learned parameters.
- These are then layered for **abstract feature learning** (deep learning), where features can be used to perform a given task.

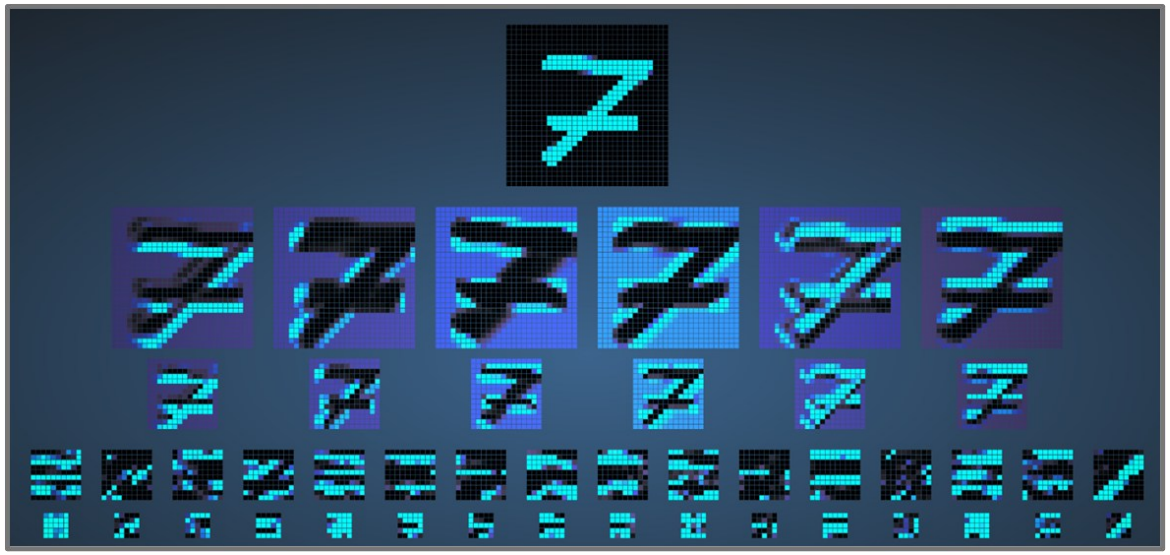


Figure: Example of abstract feature learning across many learned kernels

Back to the problem at hand…

- Muon Imaging requires ***long exposure times*** to gather enough data for object resolution.
- **Why?** We're waiting for global pixelwise differences to exceed a threshold that allows ***human perception*** to identify objects.
- **Key Question:** Can we detect these differences ***before*** they become perceivable to the human eye?

University of Glasgow

3. Machine Learning Results

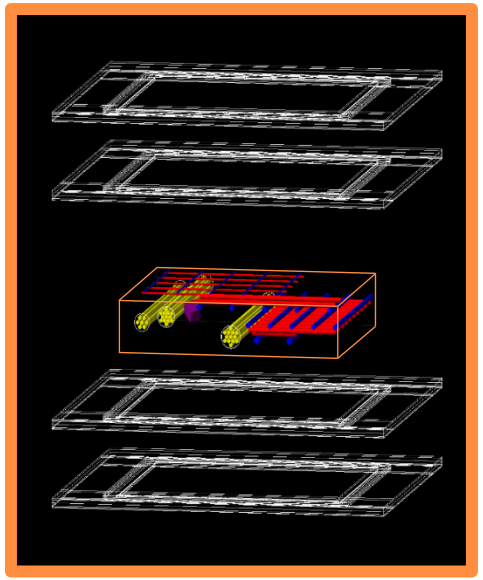
(For a Concrete-NDT Problem)

Creating a Dataset

- For a **supervised** task, we need inputs matched with ground truth labels.
- Due to the long sampling times, and volume of data required, we **cannot rely on real data**.
- We instead use muography data from **physics simulations** for ML model training.

Simulation Specs:

- **Framework:** Geant4 with Ecomug.
- **Detector:** Lynkeos Muon Imaging System (MIS).
- **Block Dimensions:** 1m x 1m x 0.2m.
- **Sampling time:** 100 days (14.4×10^6 muons/day).
- Image reconstruction using point of closest approach (PoCA).



2D Image Processing

Dataset:

- **700 Volumes** (split: 560 train/ 70 validation / 70 test).
- Each slice has **100 versions**, reflecting different sampling durations (1–100 days).
- Each slice has a **simulation-derived ground truth**.

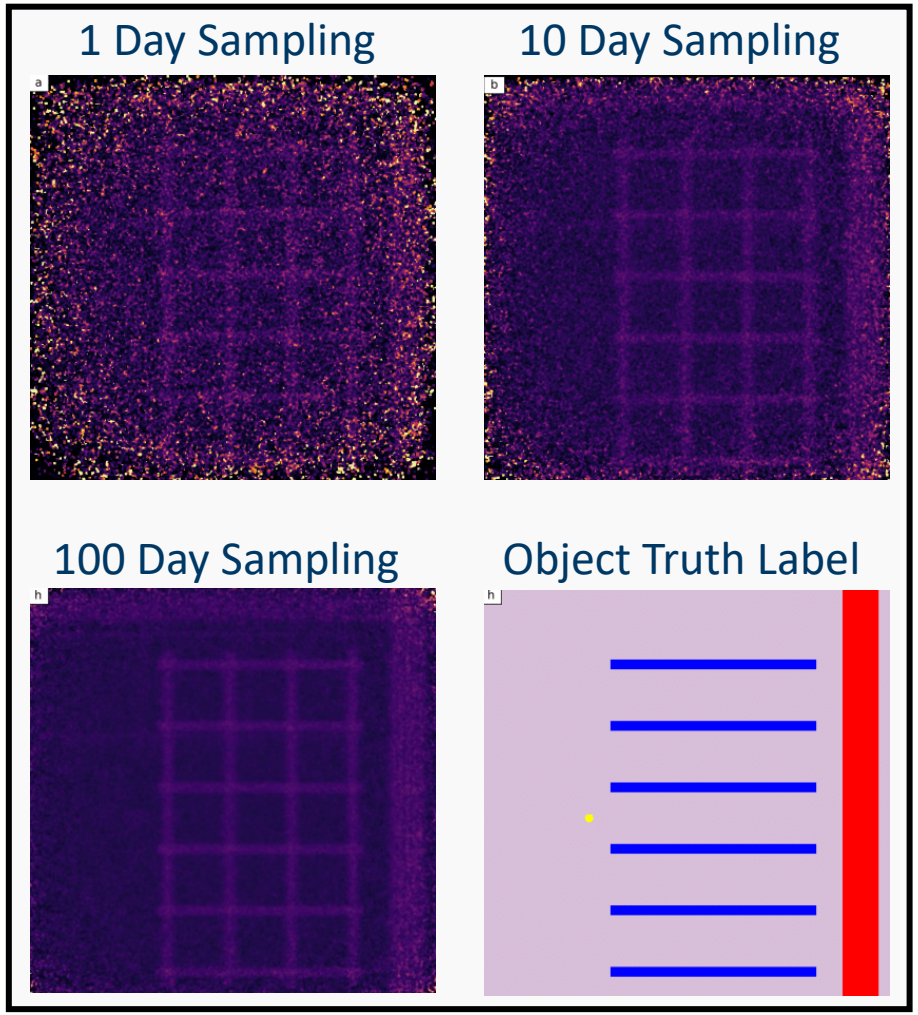
Aims:

Primary Objective: ***Upsampling model***

- Denoise and enhance relevant features.
- Trained with high-sample (100-day) reference images.
- Model: cWGAN-GP

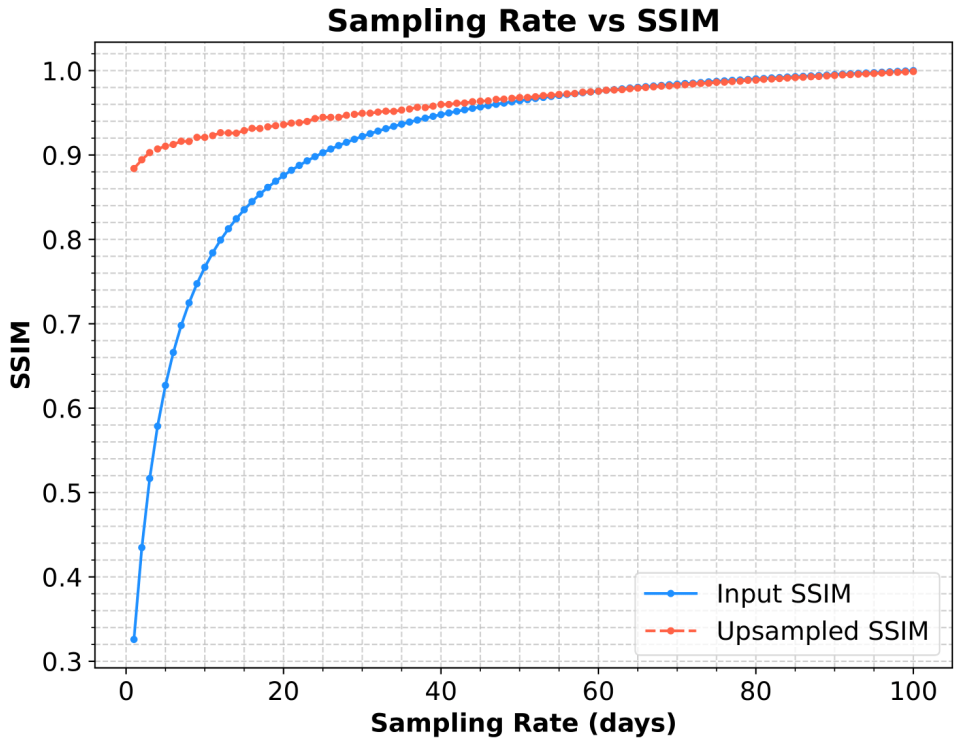
Secondary Objective: ***Segmentation model***

- **Identify object features** from upsampled images.
- Uses simulation object truths for evaluation.
- Model: U-Net



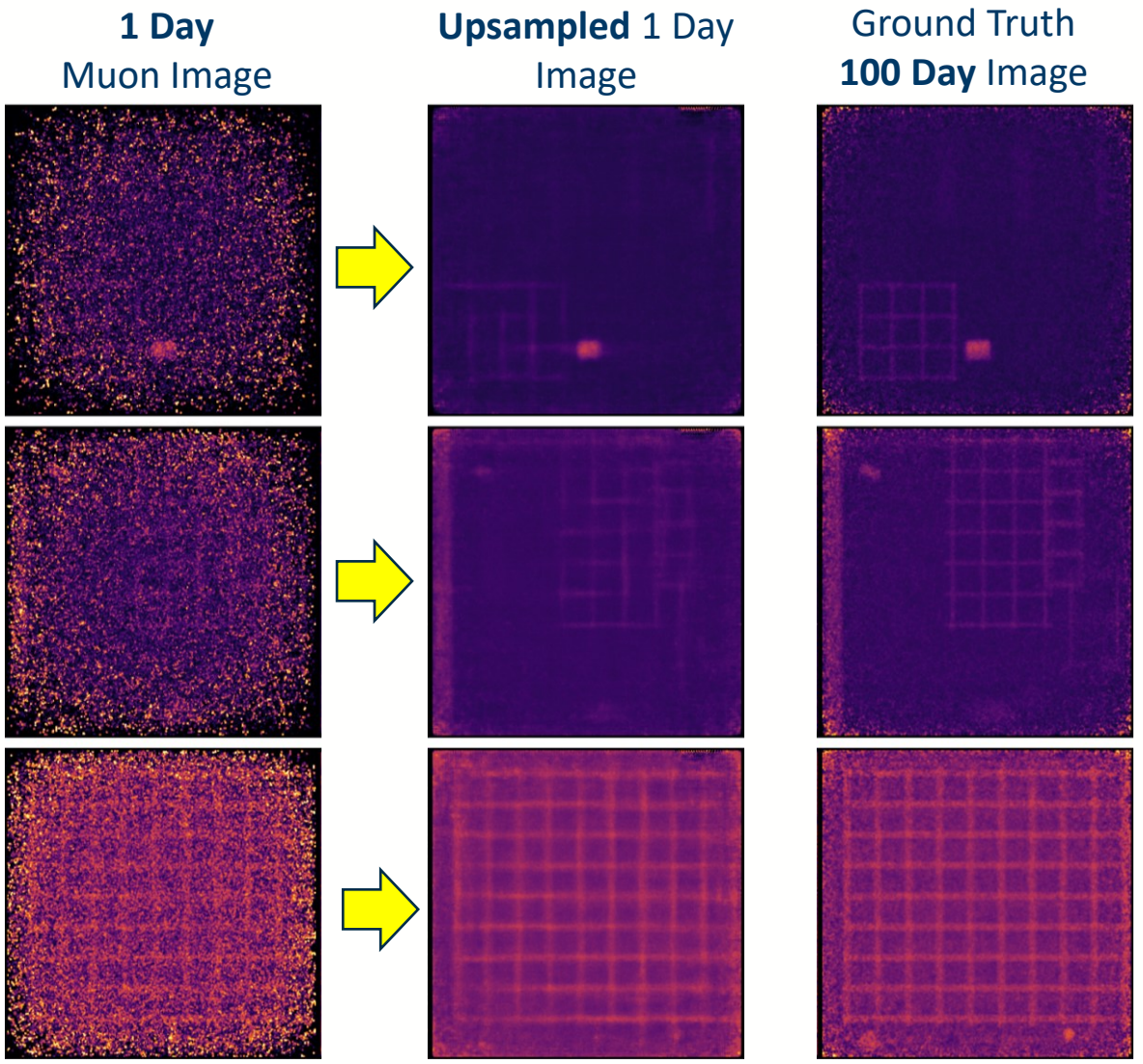
[3] O'Donnell, et al., 2025. https://doi.org/10.3390/particles8010033

2D Image Upsampling



Takeaways:

1. The model can take **1 days'** worth of data to produces an image that would otherwise take **~20 days.**
2. At around **50-60 days**, we see convergence.



[3] O'Donnell, et al., 2025. https://doi.org/10.3390/particles8010033

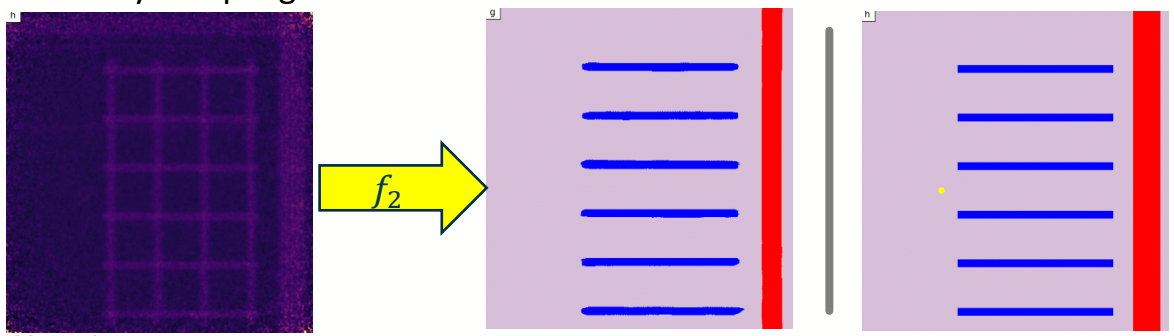
University of Glasgow

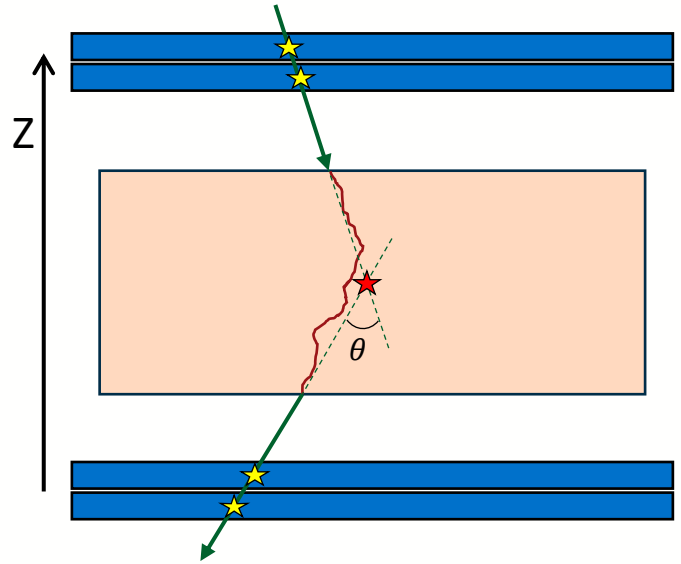
2D Segmentation: Z-Smearing

Input:
100 Day Sampling

Output:
Segmentation Map

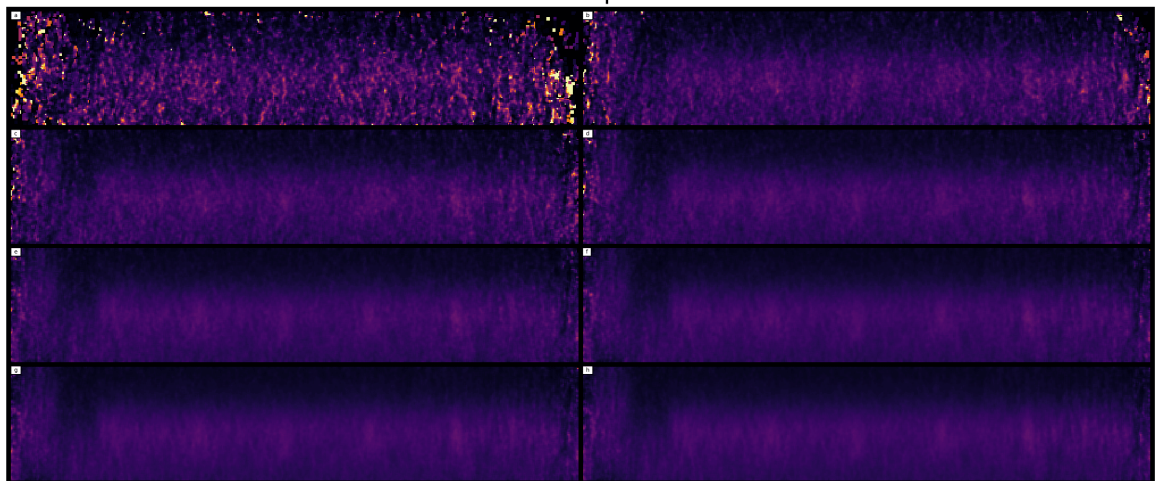
Reference Truth:
True Geometry



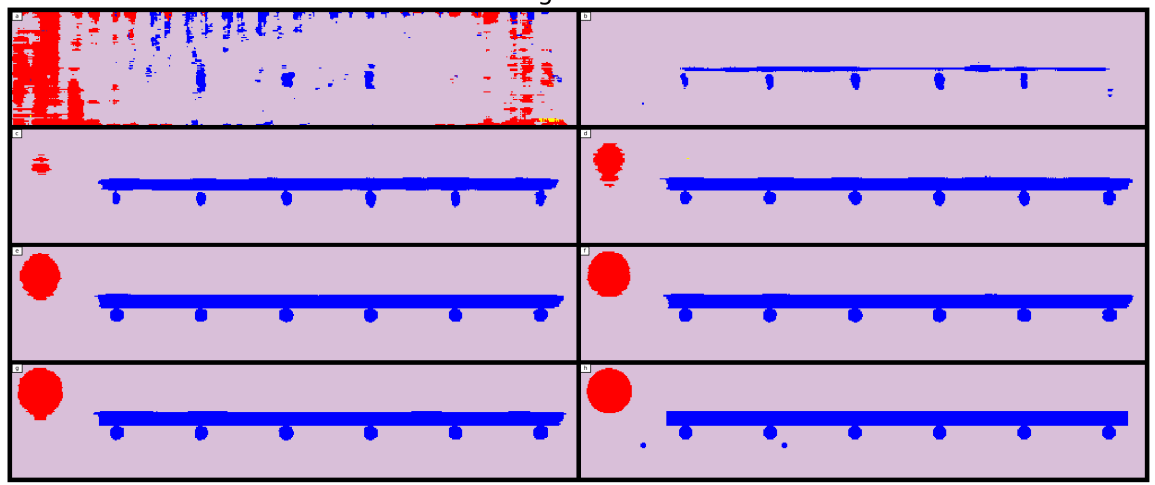


[3] O'Donnell, et al., 2025. https://doi.org/10.3390/particles8010033

Muon Inputs



Segmented



- Inverse imaging problem means images get **smeared in the z-direction.**
- Since the model is using known geometries (without smearing), it has **learned to distinguish between shadows and objects.**

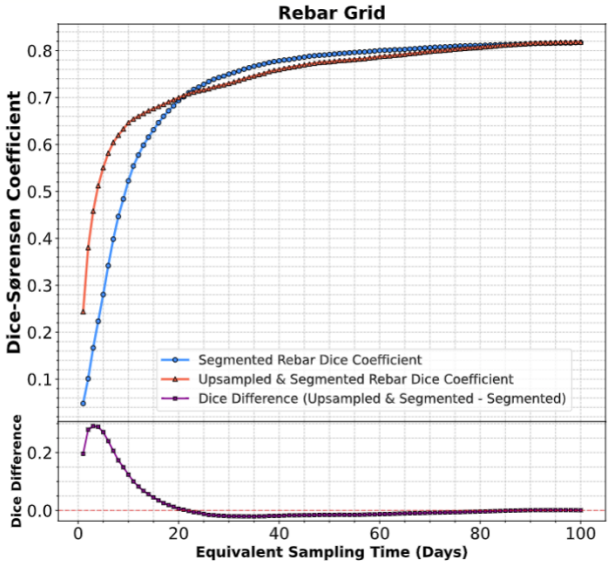
1 Day	5 Day	Lilac = Concrete
10 Day	20 Day	Blue = Rebar Grid
40 Day	60 Day	Red = Tendon Duct
80 Day	Ground Truth	Yellow = Air Void

University of Glasgow

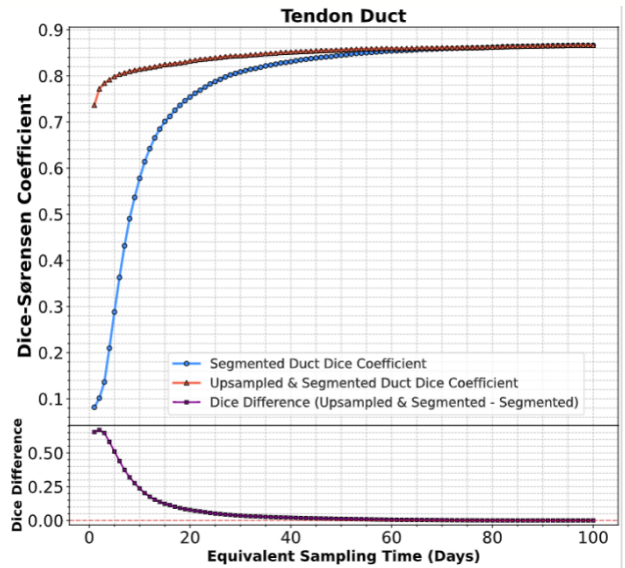
Upsampling and Segmentation

- Dice coefficient ranges from 0 (bad) to 1 (perfect).
- Above 0.7 is considered good.

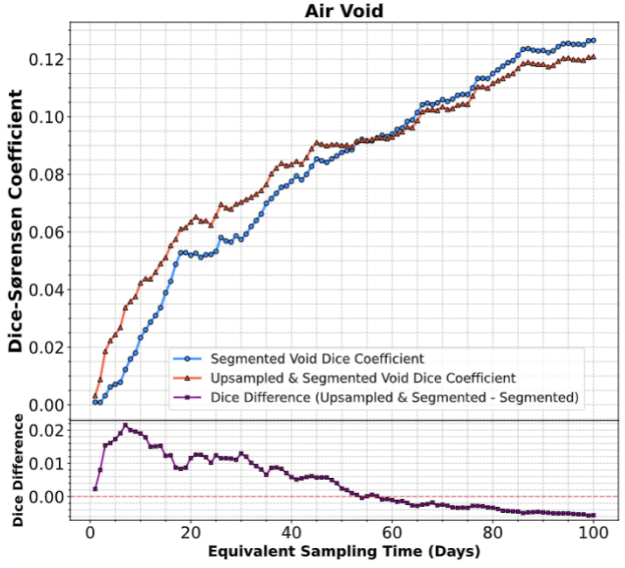
$$\mathrm{Dice}_i = \frac{2 \times \mathrm{TP}_i}{2 \times \mathrm{TP}_i + \mathrm{FP}_i + \mathrm{FN}_i}$$



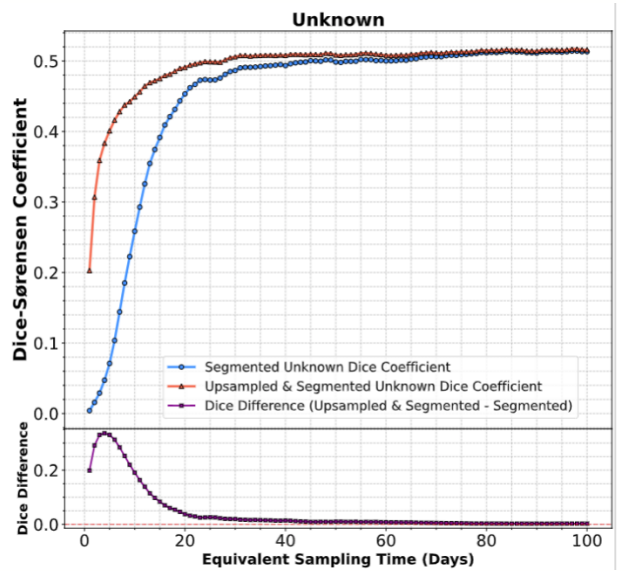
- Good detection scores
- Performance increase to ~20 days



- Very good detection score
- Significant performance increase up to ~ 50 days



- Poor overall detection
- Little performance improvement.



- Ok detection scores.
- Performance increase to ~20 days

[3] O'Donnell, et al., 2025. https://doi.org/10.3390/particles8010033

University of Glasgow

3. Improving our Results

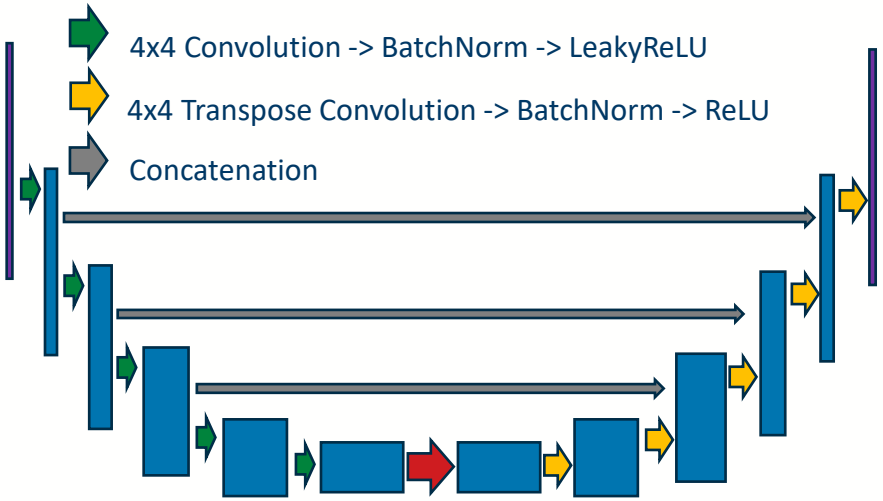
(Looking at segmentation)

Altering Model Architecture: Baseline Model

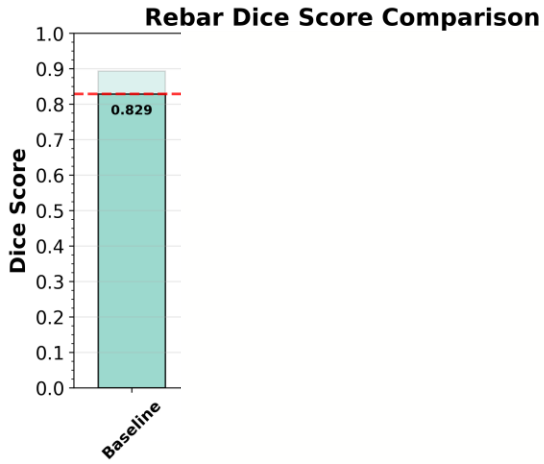
- High sampled image → Segmentation

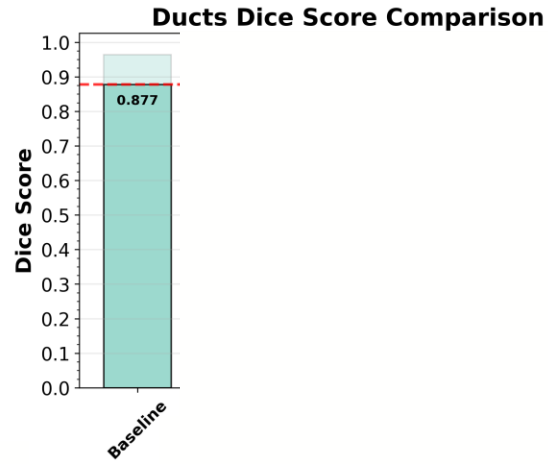
$$f: x_{60\,days} \longrightarrow y$$

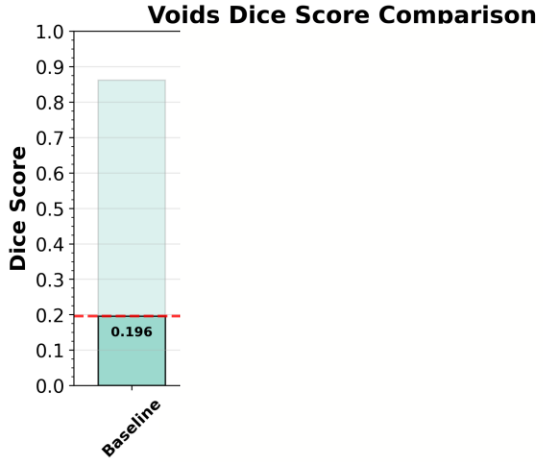
- Using just a U-Net model:

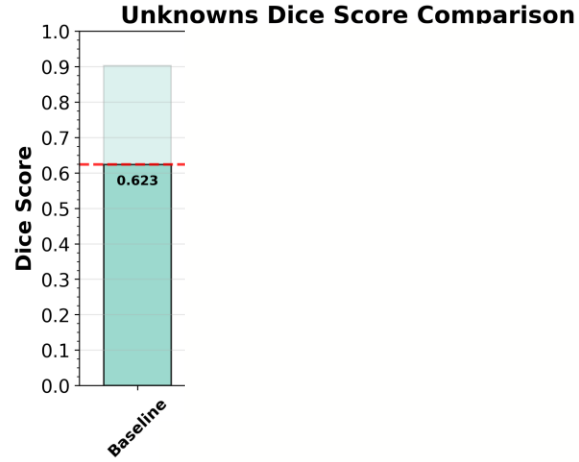


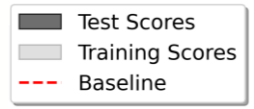
- Background class omitted from loss calc.
- Hyperparameters are all kept constant:
  - Batch size: 64
  - Cosine LR scheduler (5e-4 to 1e-6)
  - Trained for 150 epochs











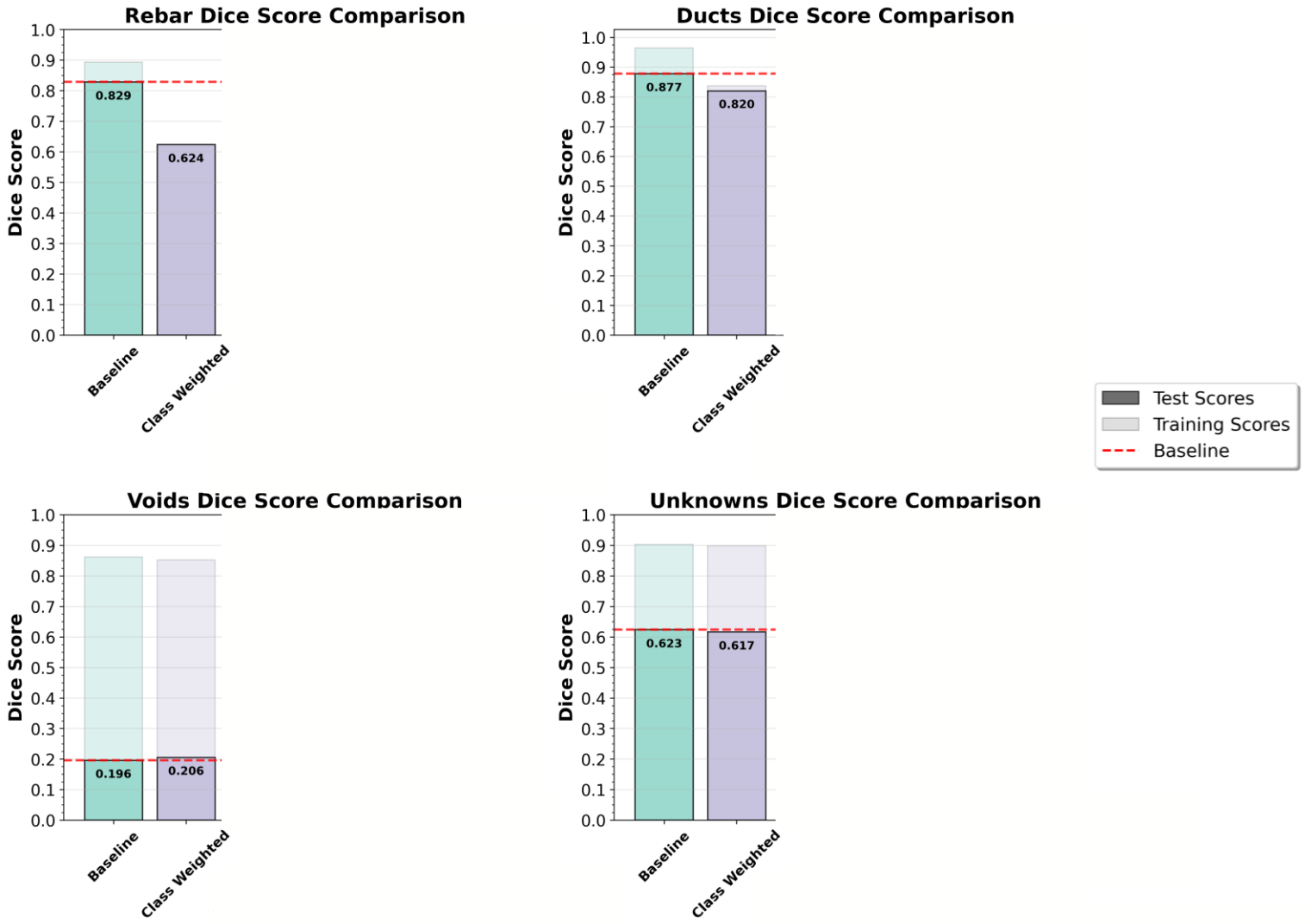
Inverse-Frequency Class Weighting

Alternative training/architecture approaches:

- Class weighting

- **Imbalanced classes** in dataset:
  - Concrete: 95.5%
  - Rebar: 1.17%
  - Ducts: 3.11%
  - Voids: 0.11%
  - Unknowns: 0.07%

- **Solution:** Inverse-frequency class weighting in loss function.



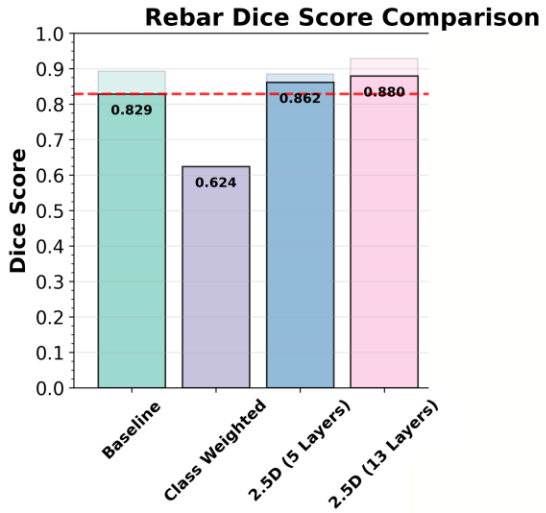
2.5D Context

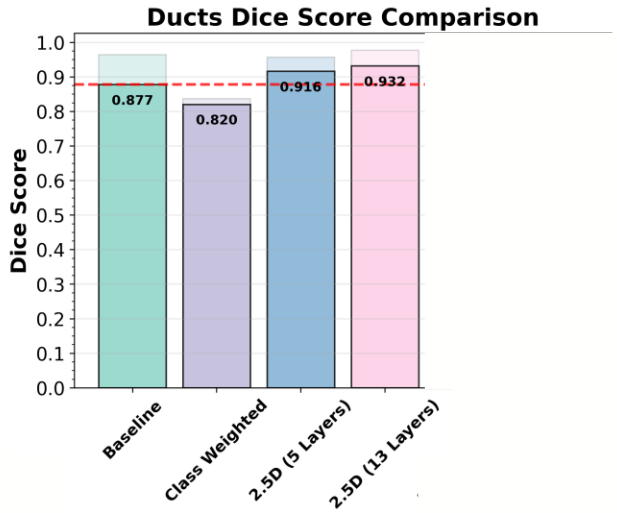
Alternative training/architecture approaches:

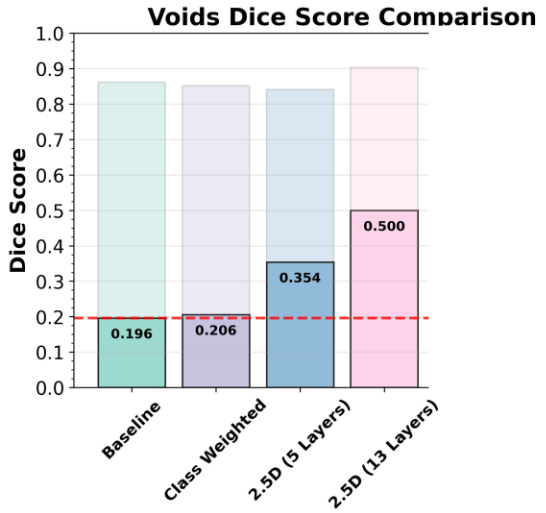
- Class weighting
- 2.5D Context

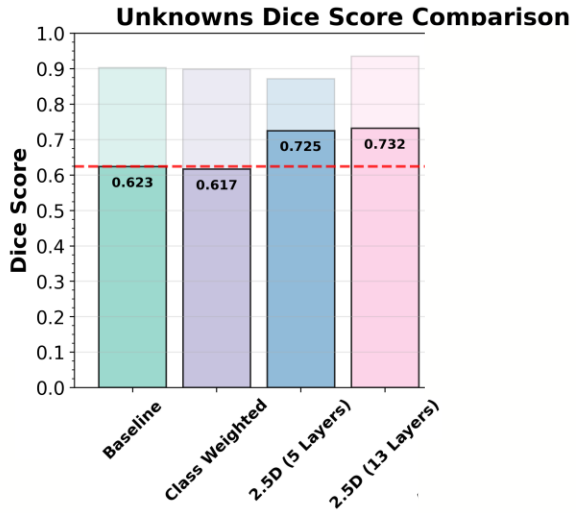
- 2D model sees **no context** in the slices above and below.

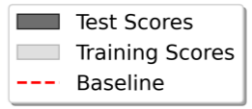
- **Solution:** 2.5D Model
  - Stack adjacent slices in the input channel dimension.
  - Lightweight compared to a 3D conv. model.









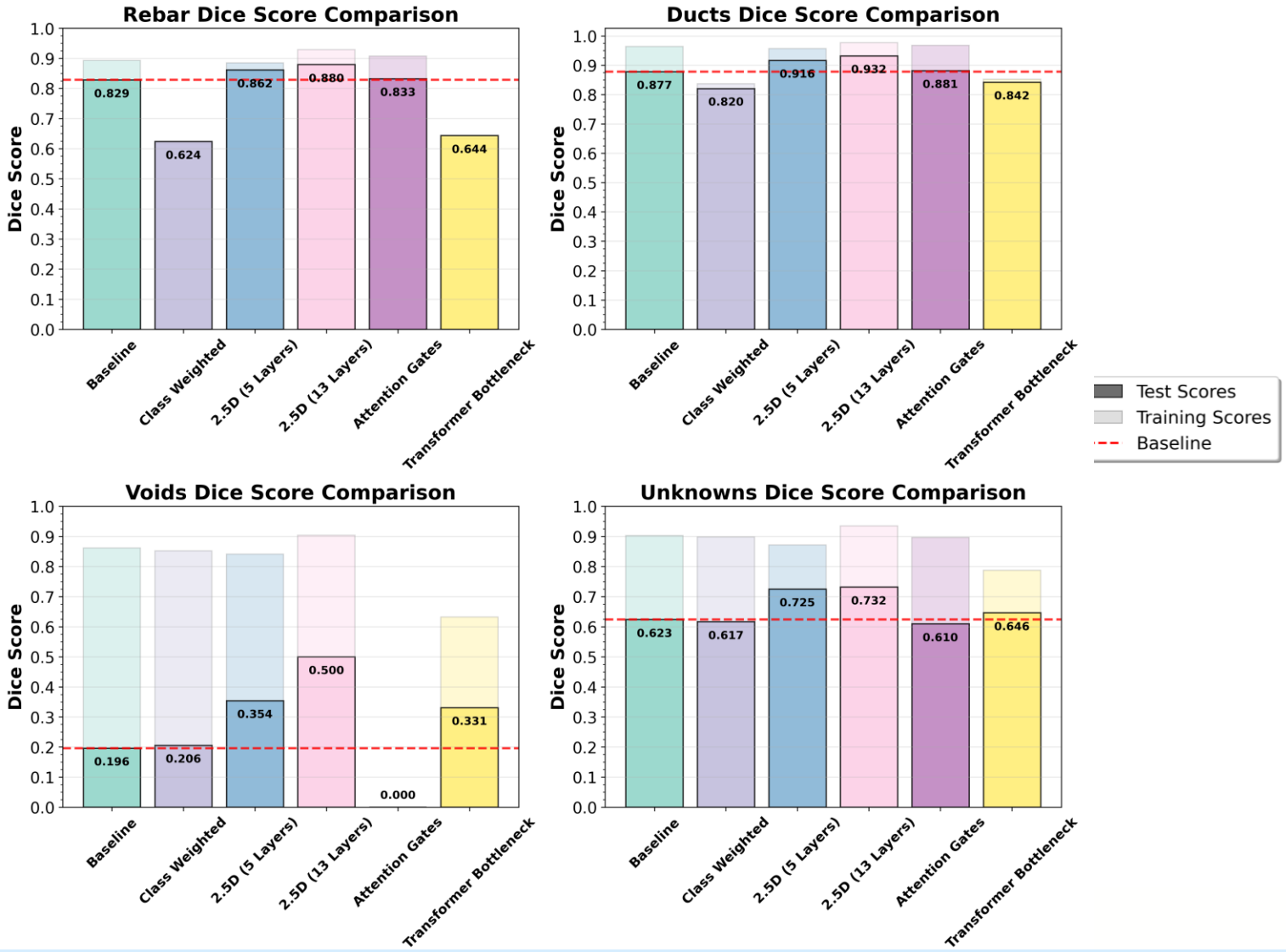


Attention Blocks

Alternative training/architecture approaches:

- Class weighting
- 2.5D Context
- Attention Gates
- Self-attention block (at bottleneck)

- **Convolutions are limited** by the receptive field of the convolutional kernel.
- Attention mechanisms provide **global context awareness**.
- Attention gates and Transformer blocks were tested.



University of Glasgow

Summary

Muon Imaging Problems:

1. Imaging time.
2. Z-plane smearing.
3. Interpretability.

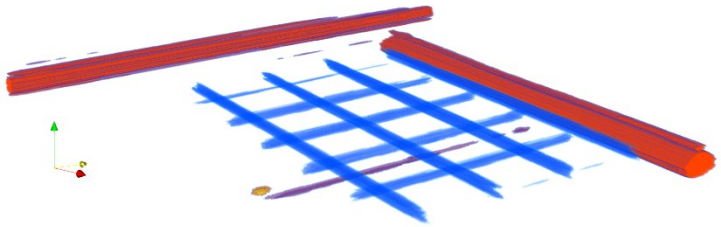
Machine Learning Solution:

- **Upsampling** significantly reduces imaging time and reduces noisy effects.
- **Segmentation** significantly reduces smearing effects AND provides automatic interpretation of results.
- **2.5D Models** significantly improve results.

Moving Forward:

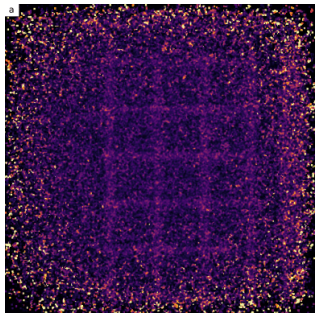
- Get models working on **real datasets** and **generalise well** to different scenes.
  - ➢ New datasets with better generalisation.
- Modelling **defect detection** scenarios:
  - ➢ RAC concrete (rebar corrosion)
  - ➢ Tendon duct defect analysis (voiding/strand snapping/strand corrostion)

1 Day Upsampled and segmented:

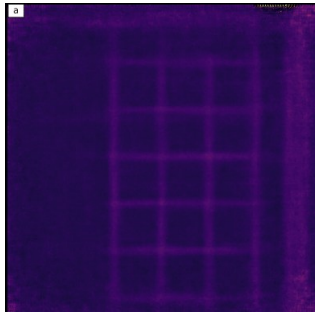


Data Pipeline:

Input:
1 Day Sampling image

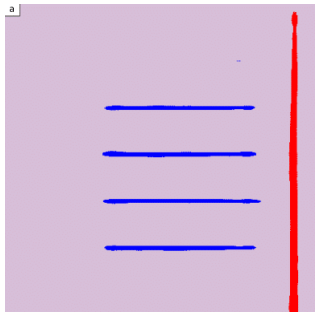


Step 1: Upsampling



Step 2: Segmentation

Output:
Upsampled & Segmented image



Thanks for Listening

Additional thanks to my supervisors D. Mahon, G. Yang and S. Gardner, as well as E. Niederleithinger from BAM, for their guidance and support.

For more info, see https://doi.org/10.3390/particles8010033, where this work has been recently published.

University of Glasgow

Backups

A Brief Overview of Machine Learning

Machine learning uses data-driven approaches to create complicated non-linear functions.

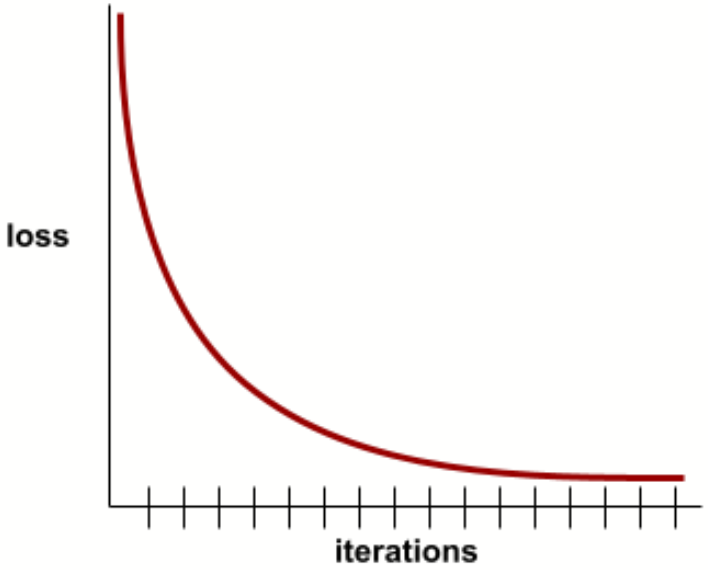
Where a conventional linear model (e.g. $y = m\,x + c$) transforms x inputs to y outputs through the parameters m and c, a trained machine learning model $y = M(x)$ also transforms x inputs to y outputs using a complex network of **millions of parameters**.

Supervised Learning Process

To determine the values of this large number of parameters, we use **supervised learning**. This is where we have a dataset of inputs and expected outputs (truths).

To train a model:

1. **Forward-pass** the inputs through the model to generate outputs.
2. **Compare** the generated outputs with the expected outputs using a **loss function** (e.g. mean squared error).
3. **Back-propagate** gradients through the model, using an **optimiser** and the calculated loss.
4. **Update** model parameters.
5. **Iterate** through steps 1-4 until the loss minimises/converges.



---------- BACKUP SLIDES ----------

Convolution in CNNs

- CNNs however learn the kernels they use – allowing for complex task-specific learning.
- The learnable parameters in a CNN are the components of these kernels – each containing a set of weights ($w_{i,j}$) and a single bias term (b):

$$O_{i,j} = w_{i,j} \times I_{i,j} + b$$

- $I_{i,j}$ is the input (3x3 window of input)
- $w_{i,j}$ are the weights of the 3x3 kernel
- b is the bias term
- $O_{i,j}$ is the 3x3 output of the element-wise product with bias.

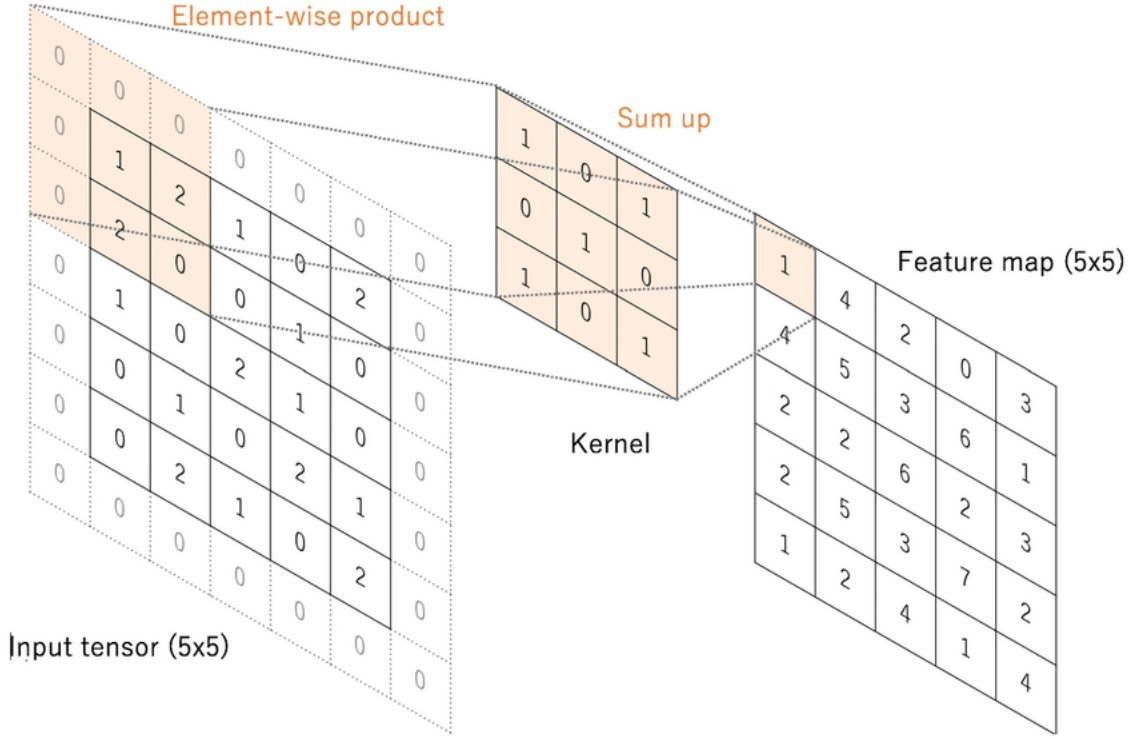
9 params for one 3x3 kernel

$w_{0,0}$	$w_{1,0}$	$w_{2,0}$
$w_{0,1}$	$w_{1,1}$	$w_{2,1}$
$w_{0,2}$	$w_{1,2}$	$w_{2,2}$

$+ b$

Convolutional Feature Extraction

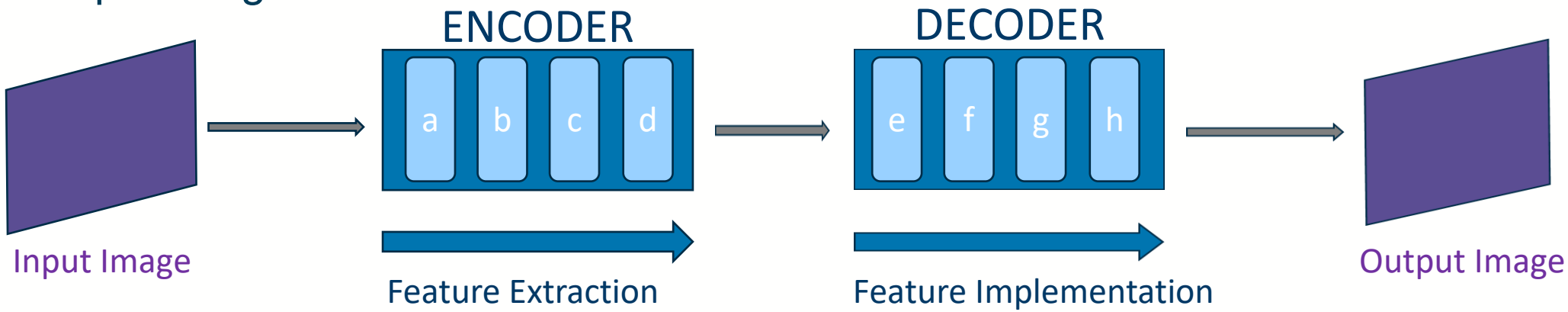
- Convolution operations have been used for image processing for a long time.
- The feature extracted from an input image depends on the kernel.
- Convolution of the input with a kernel produces a feature map.
- Many different kernels can be performed, each looking for different features and each producing a feature map.



$$= \sum_{i=1}^{3} \sum_{j=1}^{3} \begin{bmatrix} 0*1 & 0*0 & 0*1 \\ 0*0 & 1*1 & 2*0 \\ 0*1 & 2*0 & 0*1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} = 1$$

Encoder-Decoder Architecture

- For image translation, we need to extract features from the input image and build them back into an output image.



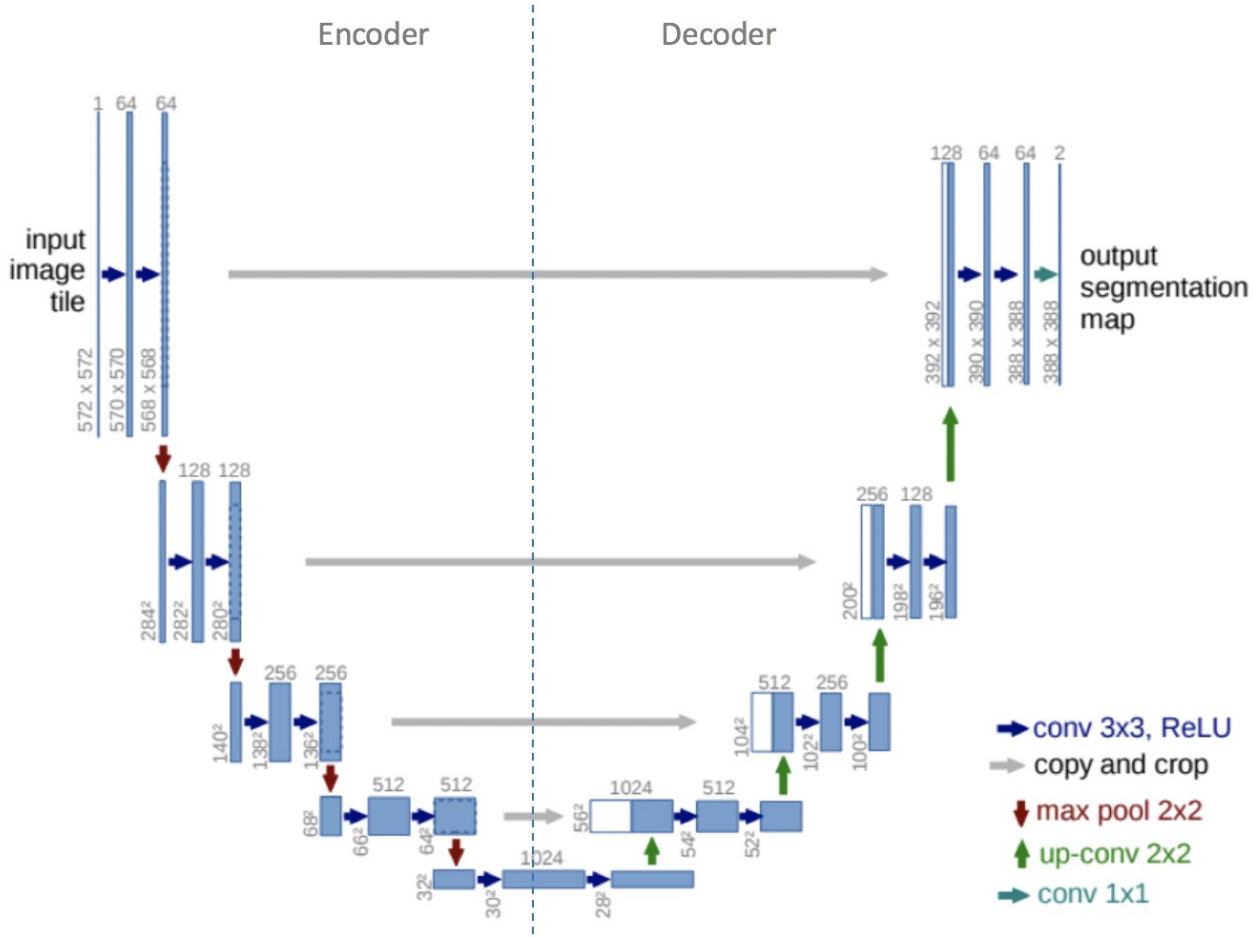
- There are two main techniques for feature extraction in machine learning:
  - Convolutional layers – *Convolutional Neural Networks (CNNs)*
  - Attention blocks – *Vision Transformers (ViTs)*

- We will explore using the older, but well established CNNs.

---------- BACKUP SLIDES ----------

U-Nets

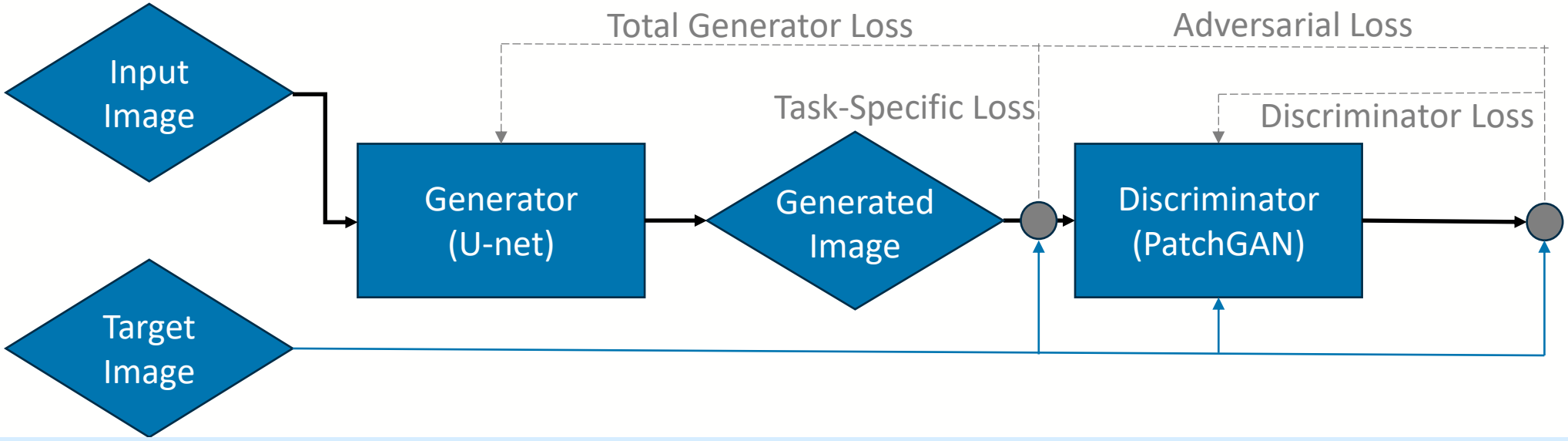
- Standard encoder-decoder CNNs are lossy – lose information.
- Introduce ‘skip connections’ between layers in the encoder and decoder.
- Allows for uncaptured, minor details to be preserved while keeping model complexity low.
- U-Nets are widely used for I2I translation tasks, especially in medical imaging.



The Conditional GAN (cGAN)

- cGANs are the supervised version of the GAN (conditioned on an input).
- Contain two parts: **generator** and **discriminator**.
- **Adversarial process**: compete until Nash equilibrium is reached.
- The model used is heavily based on the pix2pix architecture [2].

[2] https://phillipi.github.io/pix2pix/



2D Image Processing

Image to Image model (Upsampling)

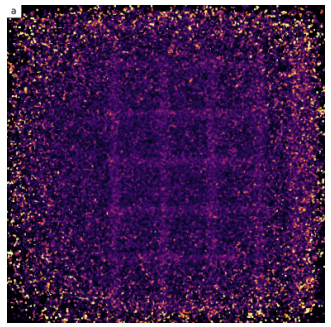
- **Inputs:** Image slices whose versions were randomised at each epoch.
- **Truths:** longest sampling time (100 day) image.

- **Loss:** Mean Absolute Error (MAE) and Wasserstein loss.

- For an input image x:

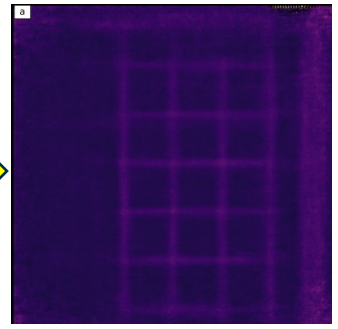
$$f_1 : x_i \longrightarrow x'_{100} \quad \text{where } i \in \{1, 2, \dots, 100\}, i \in \mathbb{Z}$$

Input:
1-100 Day Sampling

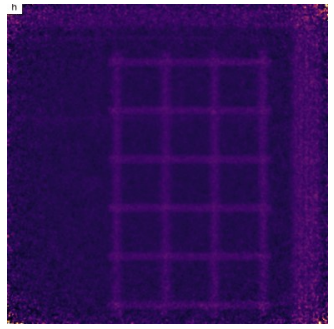


f_1

Output:
Upsampled Image



Reference Truth:
100 Day Sampling



Semantic Segmentation model

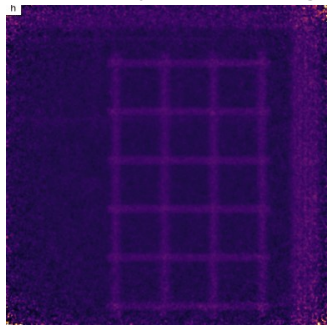
- **Inputs:** longest sampling time (100 day) image.
- **Truths:** Geometry from Geant4 Simulation.

- **Labels:** concrete, rebar, ducts, voids, unknowns.
- **Loss:** Dice and cross-entropy combined loss, Wasserstein.

- To produce a true geometry, y, and assuming: $x'_{100} \approx x_{100}$,

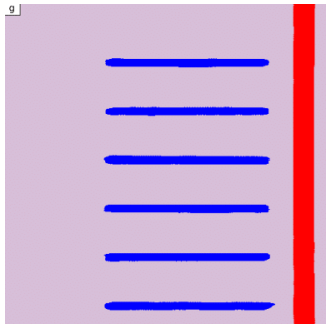
$$f_2 : x_{100} \longrightarrow y \qquad \text{where } i \in \{1, 2, \dots, 100\}, i \in \mathbb{Z}$$

Input:
100 Day Sampling

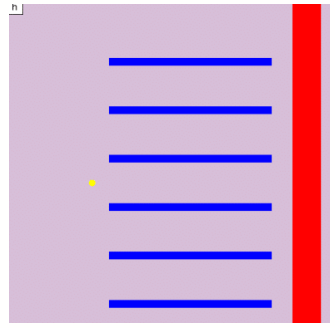


f_2

Output:
Segmentation Map

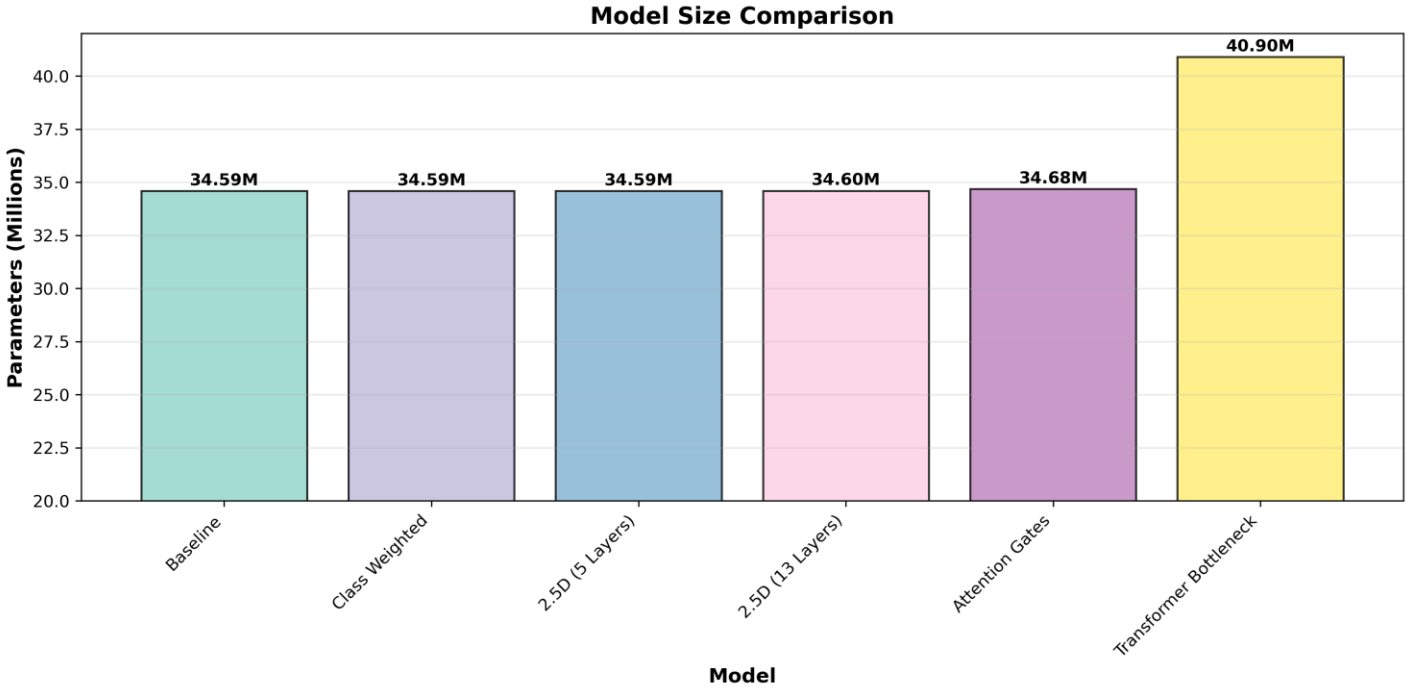


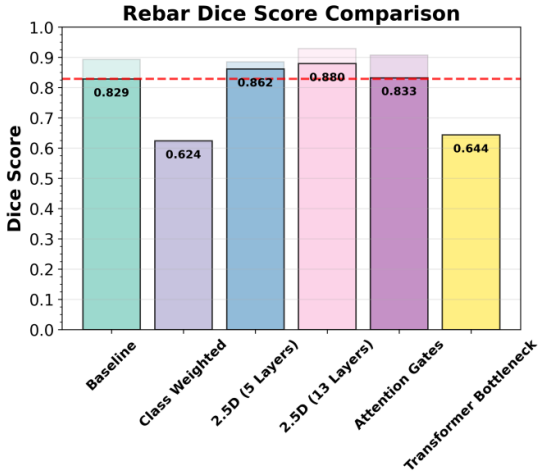
Reference Truth:
True Geometry

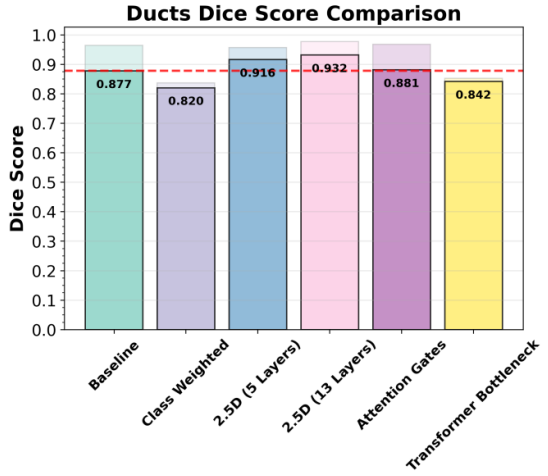


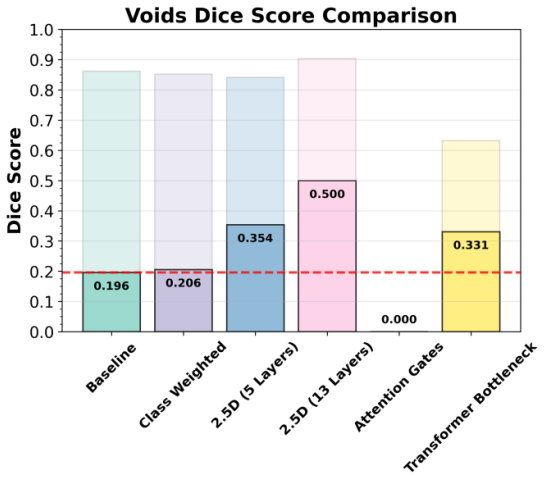
---------- BACKUP SLIDES ----------

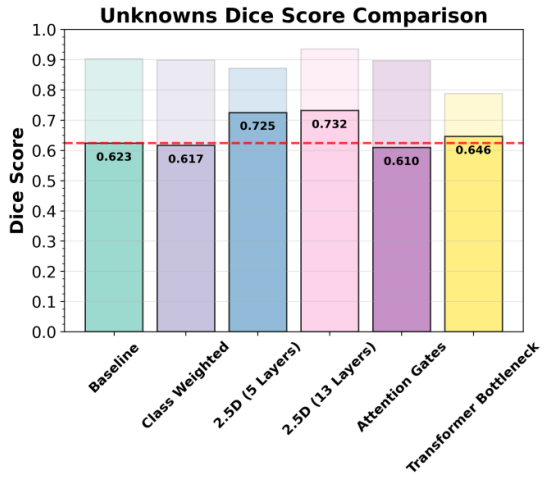
Model Parameters







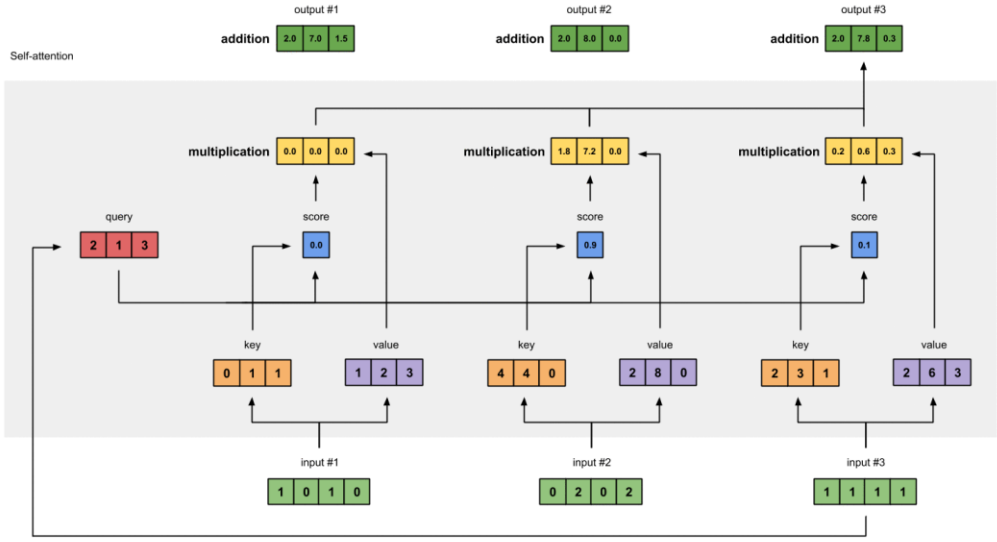




---------- BACKUP SLIDES ----------

Why do transformers help with global context?

- Input image is first tokenised into patches.
- Then flattened into 1-D vectors.
- Attention mechanism learns parameters to generate three vectors:
  - **Key** (What information can I share)
  - **Query** (What information do I need)
  - **Value** (The feature information)
- Ultimately results in **each pixel receiving a weighted combination from all pixels**, based on similarity and relevance as determined by K,Q,V.
- In CNN's we use different filters to learn features
- In attention we use multiple heads to learn different combinations of key, query, value.

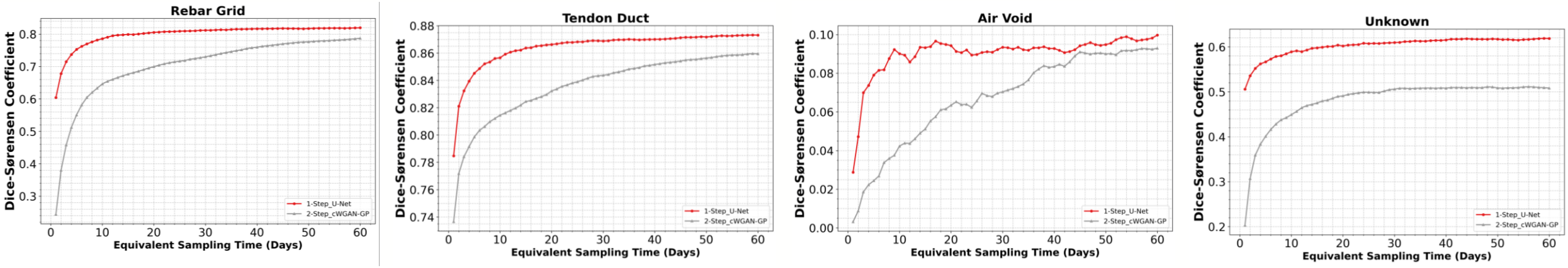


Training Strategy (1- vs 2-model approach)

- **Previously** we used two models to produce our sampling time-dependent segmentation results:

$$f_1: x_i \longrightarrow x'_{60} \text{ where } i \in \{1, 2, \dots, 60\}, i \in \mathbb{Z} \qquad f_2: x_{60} \longrightarrow y$$

- We can simplify this method and instead perform: $x_i \longrightarrow y$ where $i \in \{1, 2, \dots, 100\}, i \in \mathbb{Z}$



- Significant performance increase with one model approach – though unsurprising due to less approximations.
- However poor performance on void detection.

---------- BACKUP SLIDES ----------