Generative Adversarial Neural Networks for Muography simulation: image prediction

R. López, C. Fernández, C. Díez, P. Gómez, P. Martínez

4th Mode Workshop

Valencia, 23rd-25th September 2024

MUON systems

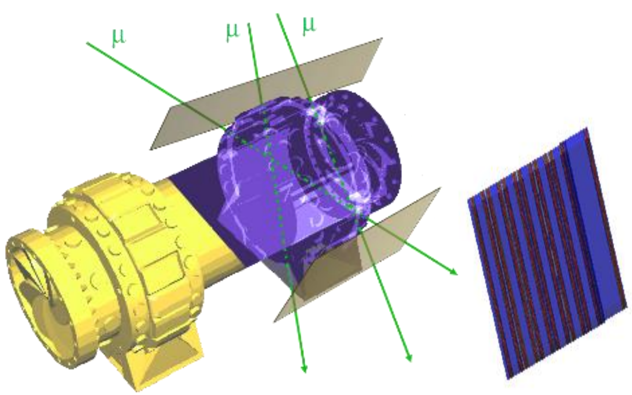
IFCA Instituto de Física de Cantabria

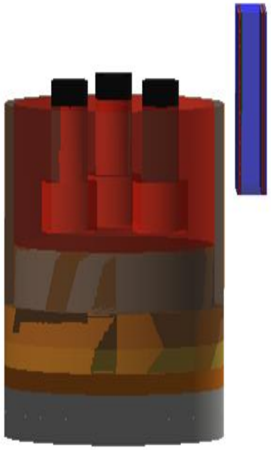
UC UNIVERSIDAD DE CANTABRIA

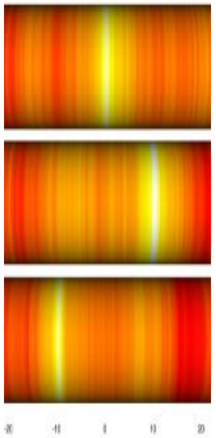
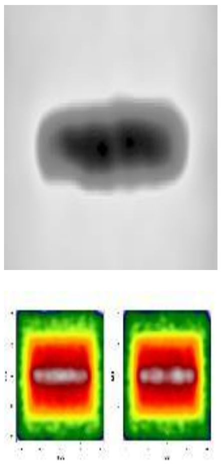
CSIC CONSEJO SUPERIOR DE INVESTIGACIONES CIENTÍFICAS

Context: muography for industrial applications

- Idea: use muon tomography as a **Non-Destructive Testing (NDT) technique in the industry**
  - Preventive maintenance of equipment (estimation of the degradation)
  - Quality control of the production process (measurement of liquid interfaces, tolerances, etc)
  - Risk assessment and evaluation (continuous monitoring of structural integrity)
- Muography has some unique properties that can be very useful for these applications
  - Large power of penetration (no problem to deal with several meters of steel)
  - No need to physically “touch” the object → can be applied to equipment in production





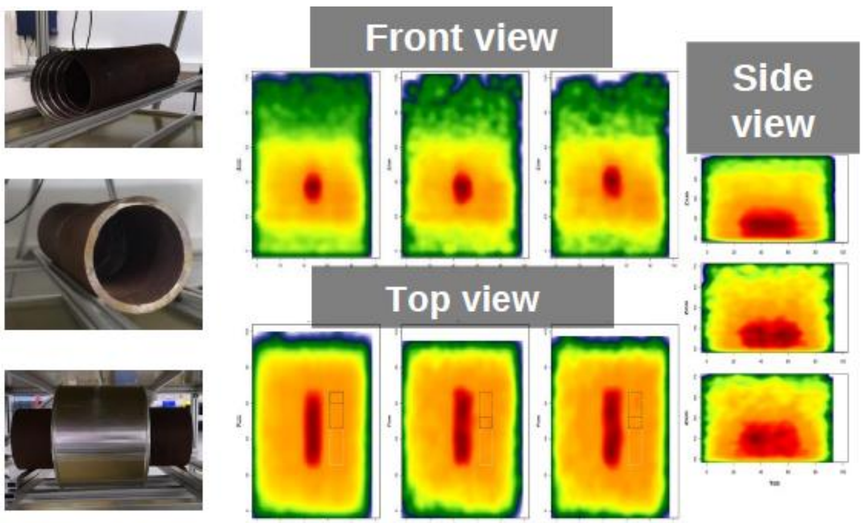



Some specifics of industrial applications

- Industrial applications usually involve to work with very well known geometries
  - In corrosión, wear, defect, etc detection the nominal geometry is known from designs
- A full image reconstruction of the object is not critical for the application
- It is more important to estimate accurately a few interesting parameters from the data

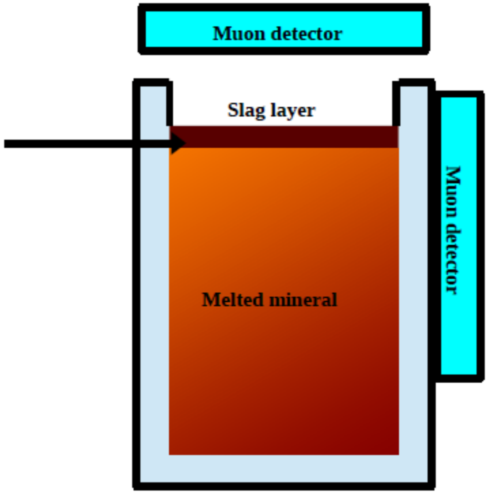
Example 1: pipes

Only the thickness of the pipe is interesting for the application (1 parameter or maybe a few to account for asymmetries in the wear of the walls)



Example 2: ladle furnace

Only the position of the slag-mixture interface is really interesting for the application (1 parameter)



Suitability for traditional ML methods

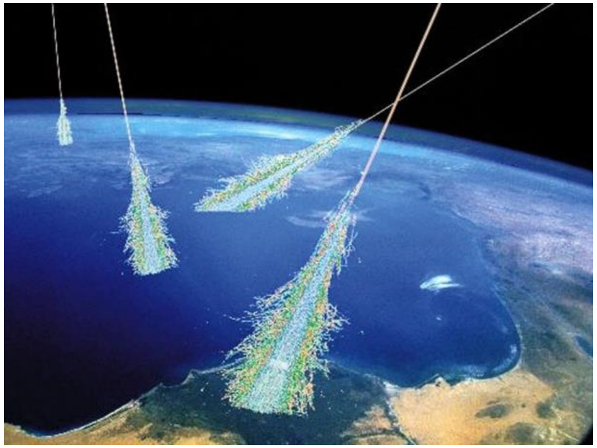
- Since the number of parameters is relatively small this can be attacked by traditional ML
  - Simple fully connected DNNs operating in regression mode to the parameters of interest
- Basic muon distributions (angular deviation, spatial deviation, etc, etc) can be used as input
  - They can be quantified for example through quantiles or any other technique
- Ideally one could use real data to train the algorithms since often this is no problem
  - Think about the pipe problem: companies have hundreds of new, fresh, perfect pipes
- To achieve good stats these algorithms require also MC simulations to complete the training
  - This is problematic since tools such as GEANT4 can be very time consuming
  - For example, in a simple setup with pipes can take 6 minutes to simulate one hour of data
- Several efforts have been performed to speed up simulations
  - See for example

https://indico.cern.ch/event/1022938/contributions/4487326/

Simulation for scattering muography

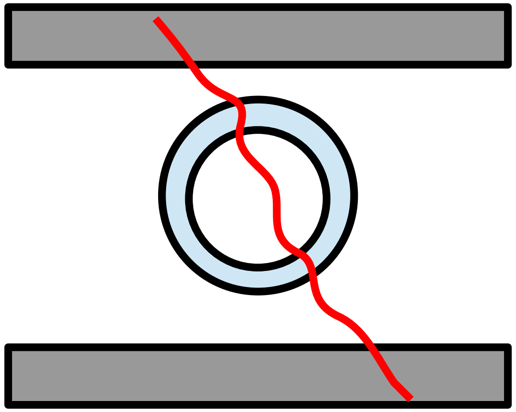
- Simulation for scattering muography has three different components

Muon flux generation



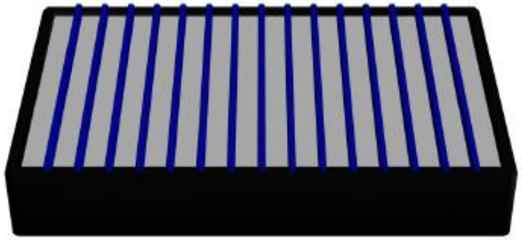
- Most generators parametrize the muon flux as a function of altitude/latitude etc
- This part is usually relatively fast
- CRY is a good example

Muon propagation through matter



- Implementation of energy loss and multiple scattering at least
- Can be very time consuming specially for complex geometres
- GEANT4 very prcise on this

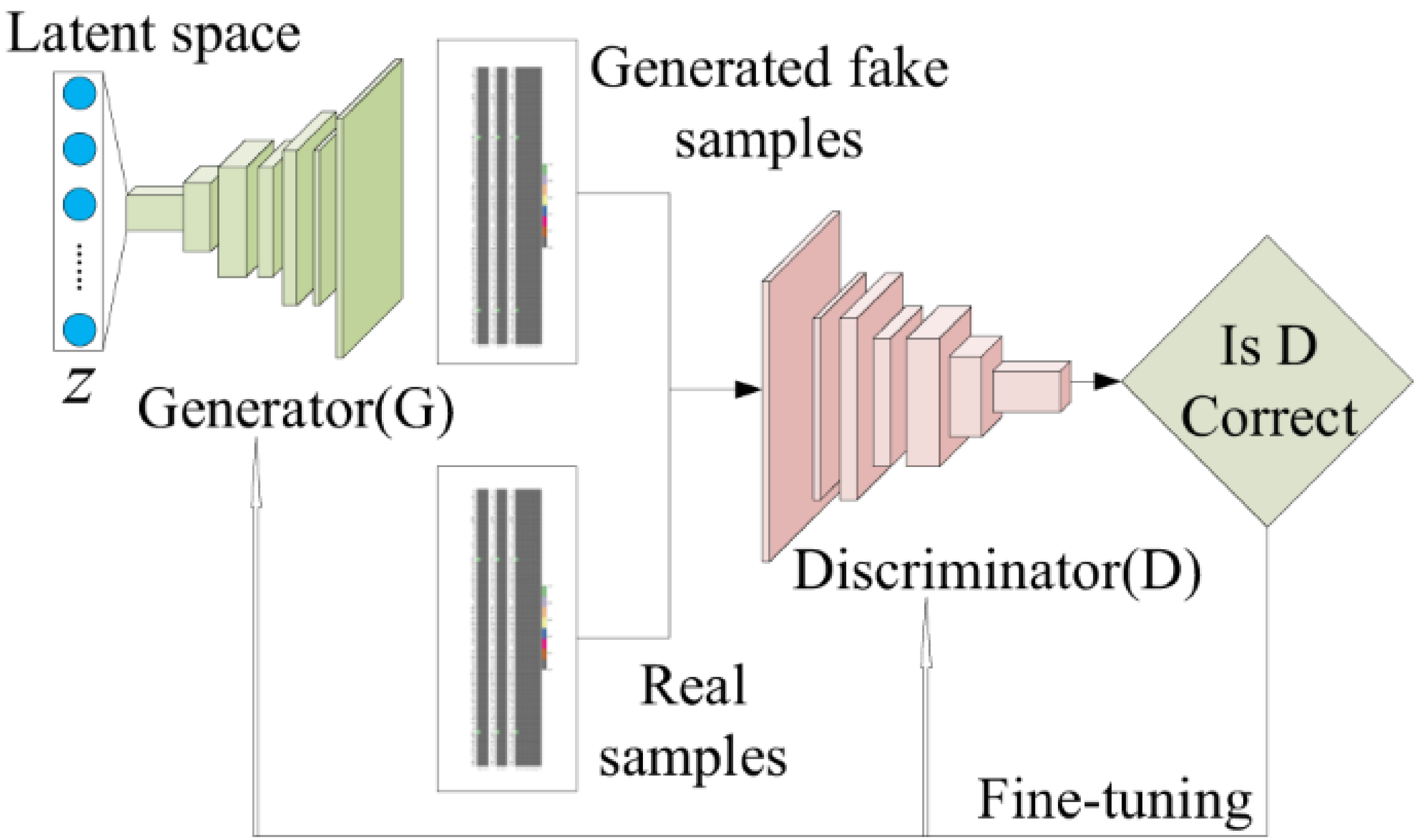
Detector simulation



- A model of the detector response has to be considered for precise MC simulation
- This part can be critical and it is typically difficult to implement
- No general récipes, every detector needs its own model

Generative Adversarial Neural Networks

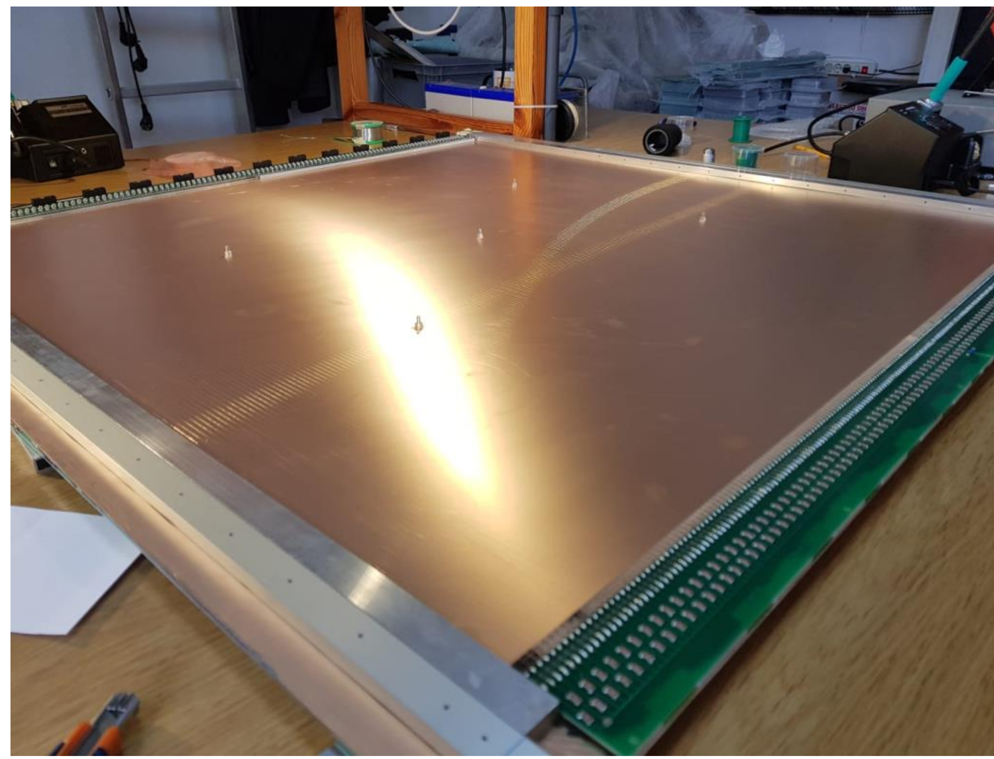
➢ We propose to use Generative Adversarial Neural Networks to produce MC simulation



Our Muography setup (I)

- Multiwire Proportional Chambers with tungsten-gold wires of 50 microns diameter every 4mm
  - Each chamber is a 89x89 cm^2 double layer with orthogonal wires to measure x and y
- Custom made electronics, ~ 95% efficiency, few microseconds deadtime, configurable trigger

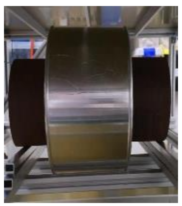
1m^2

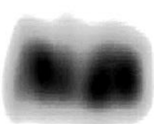


Our Muography setup (II)

Pipe corrosion

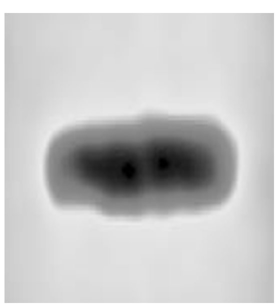
Measure of the wear: 1mm resolution
1 min exposure





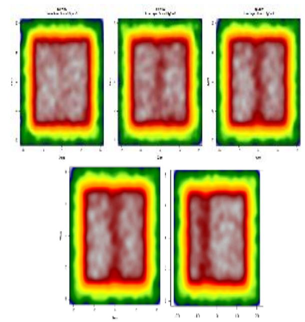
Cracks in **concrete**

Measure of the crack size:
2mm resolution
10 min exposure time



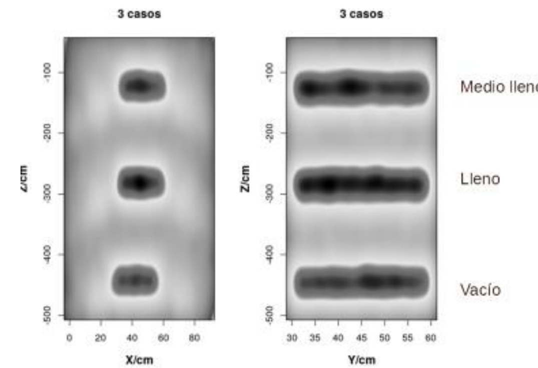
Tailings

0.2g·cm^{-3} density loss detection



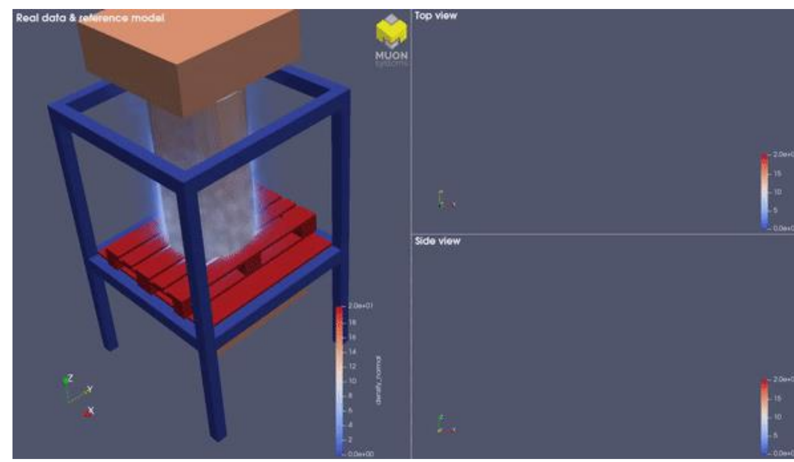
Prestressed concrete

grout level detection



Furnace hearth

Measure of the wall refractory:
1cm resolution
15 min exposure



Real data 3D reconstruction of a silicon smelting furnace

Our Muography setup for the GAN studies

- Our GAN simulation is running for a muography setup as the one used for the pipe problem
  - This setup corresponds to the one from Muon Systems (see previous slides)
- Simulator target: predict lower segment having the upper segment as input
  - This means that we rely on CRY for the simulation of the upper segment
  - All tests performed on MC samples where detectors are assumed to be perfect

Upper detector

$x_1, y_1, v_{x1} = \mathrm{atan}(\theta_{x1}), v_{1y} = \mathrm{atan}(\theta_{y1})$

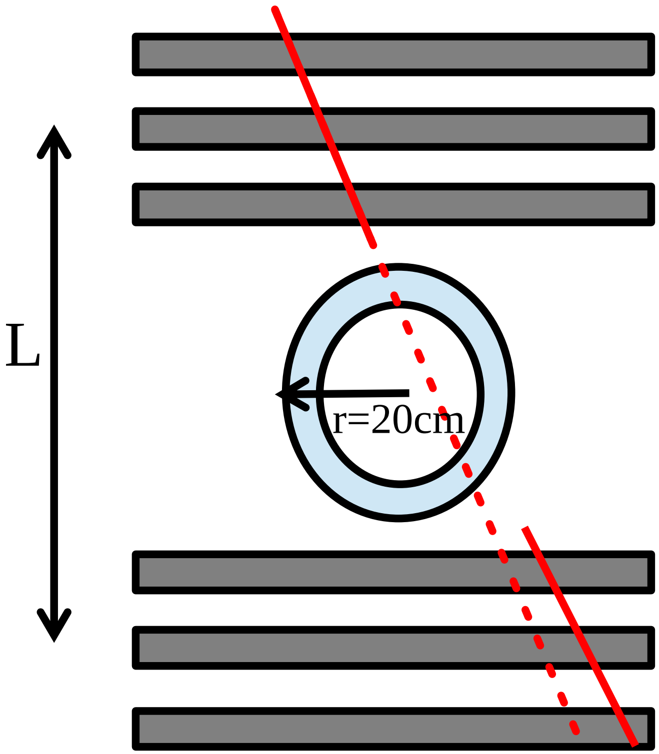
Lower detector

$x_2, y_2, v_{x2} = \mathrm{atan}(\theta_{x2}), v_{2y} = \mathrm{atan}(\theta_{y2})$

Target variables

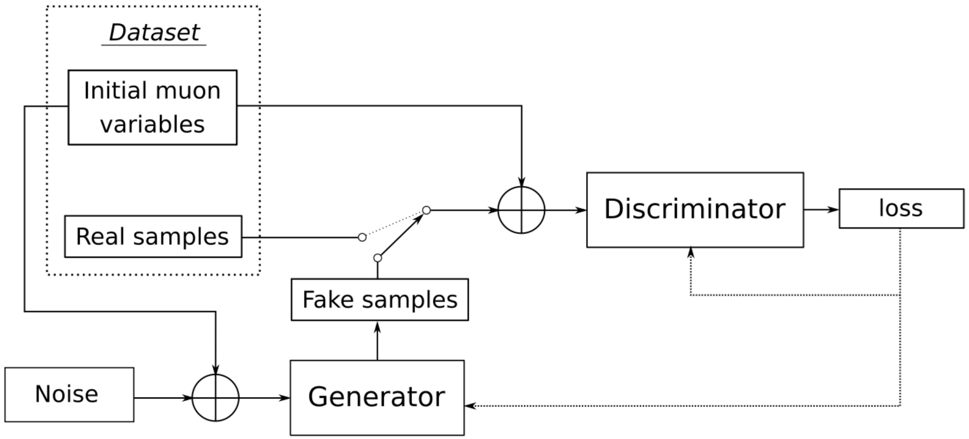
$\Delta x = x_2 - L\, v_{x1} - x_1$ $\Delta y = y_2 - L v_{y1} - y_1$

$\Delta v_x = v_{x2} - v_{x1}$ $\Delta v_y = v_{y2} - v_{y1}$



First attempt: simple GAN

- Our first attempt uses a simple GAN
  - Keras + TensorFlow
  - The variables of the segment in the first detector are given as input to the generator
  - Loss function: Mean Squared Error
  - Architecture: 512, 256, 256, 128, 64, 16 LeakyReLU
  - Latent space dimension: 64
  - Optimizer: Adam, 0.001 (halves every 50 epochs)
  - Trained for 200 epochs (Total training time ~ 2-3 hours, GeForce RTX 3090)

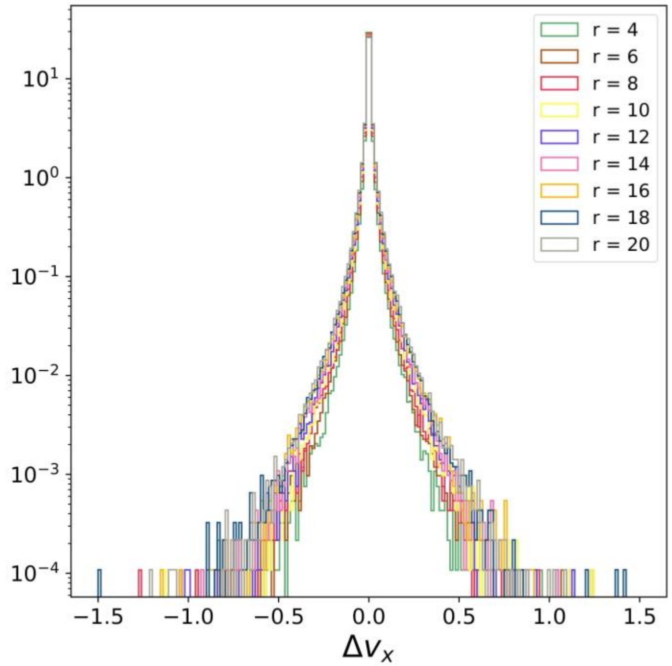


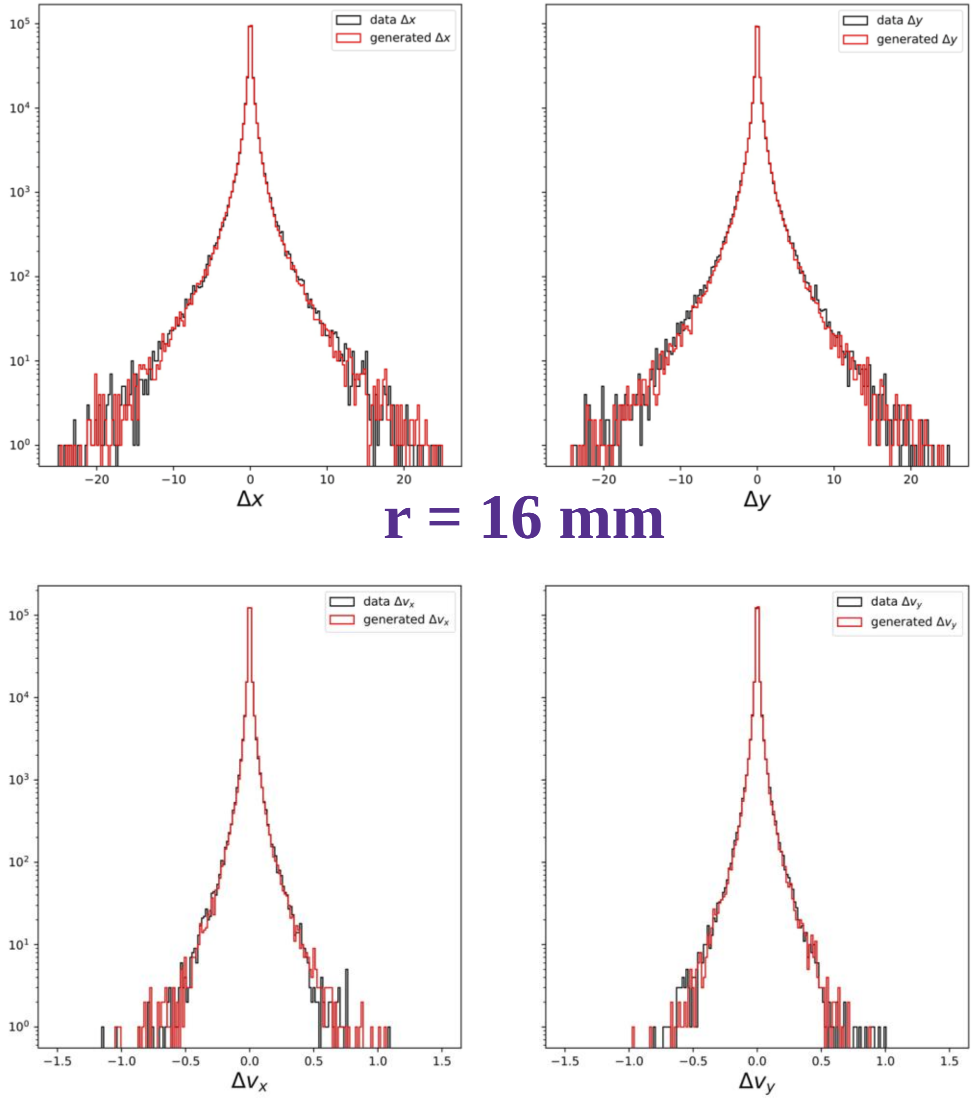
Simple GAN		
Pipe thickness (mm)	Number of training samples	Number of evaluation samples
16	306707	307352
Conditional GAN		
Pipe thickness (mm)	Number of training samples	Number of evaluation samples
4	619605	300000
6	618798	300000
8	617951	300000
10	616700	300000
14	614944	300000
16	615216	300000
18	614109	300000
20	613692	300000
12*	-	300000

Results using a conventional GAN (I)

➢ The GAN is able to produce the correct 1D distributions with a ~ 1 mm resolution

GEANT4 simulation for different pipe thickness (mm)

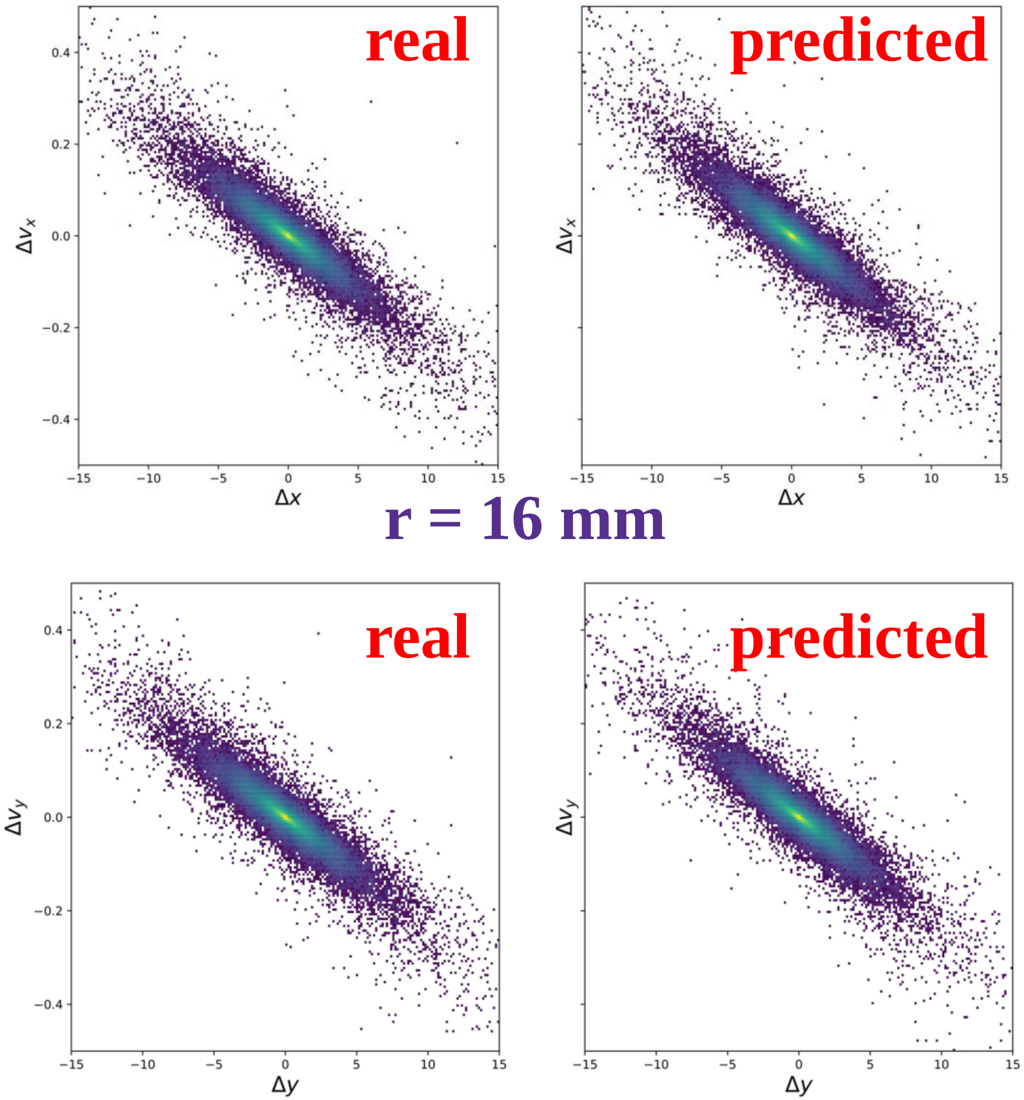




r = 16 mm

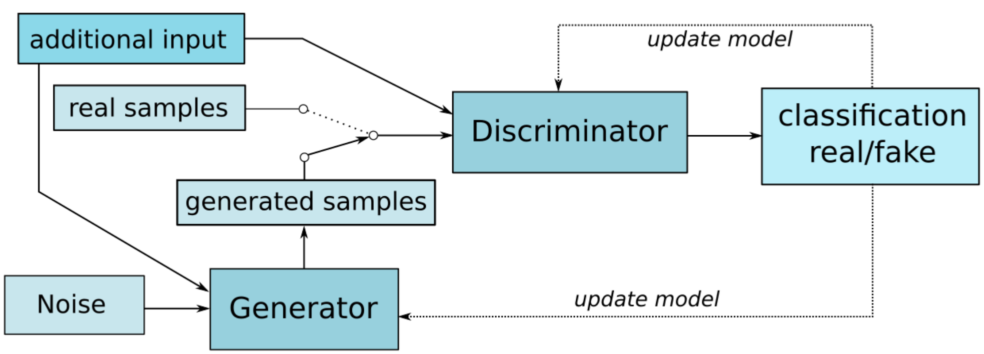
Results using a conventional GAN (II)

➢ Correlations among variables seems to be very well described as well by the GAN



Second attempt: Wasserstein conditional GAN

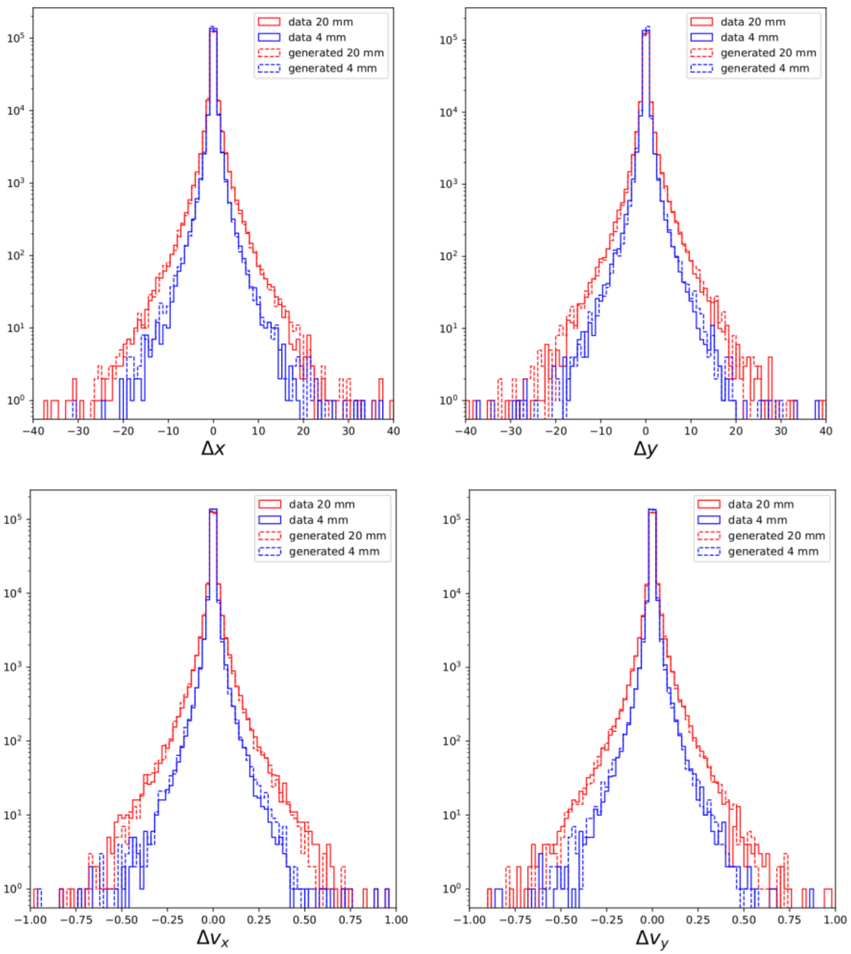
- Our second attempt uses a Wasserstein conditional GAN
  - Keras + TensorFlow
  - The variables of the segment in the first detector are given as input to the generator
  - **The thickness of the pipe to be generated is also provided as input**
  - Critic + Loss function → more stability
  - Architecture: 32, 64, 128 LeakyReLU
  - Latent space dimension: 64
  - Optimizer: Adam, 0.001 (halves every 50 epochs)
  - Trained for 1000 epochs (Total training time ~ 2-3 hours, GeForce RTX 3090)

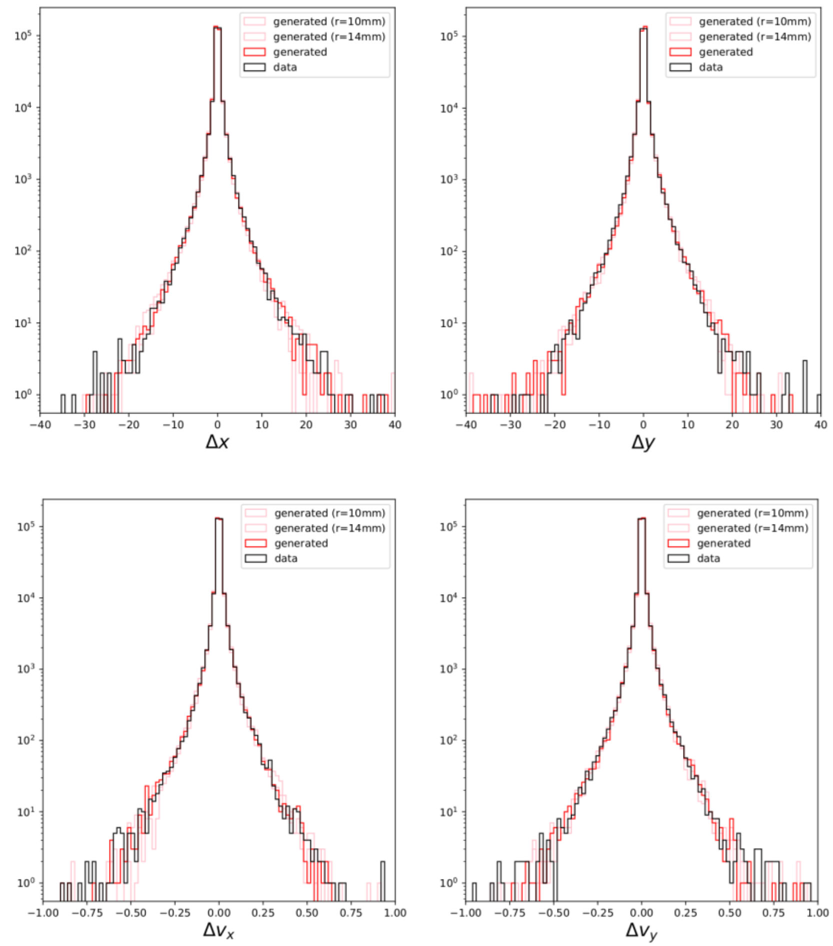


Simple GAN		
Pipe thickness (mm)	Number of training samples	Number of evaluation samples
16	306707	307352
Conditional GAN		
Pipe thickness (mm)	Number of training samples	Number of evaluation samples
4	619605	300000
6	618798	300000
8	617951	300000
10	616700	300000
14	614944	300000
16	615216	300000
18	614109	300000
20	613692	300000
12*	-	300000

Results using a conditional WGAN (I)

- The WGAN also provides very good results in describing the 1D distributions (and correlations)
- It also exhibit very good interpolation capabilities -> generating a thicknes not previously seen works fine

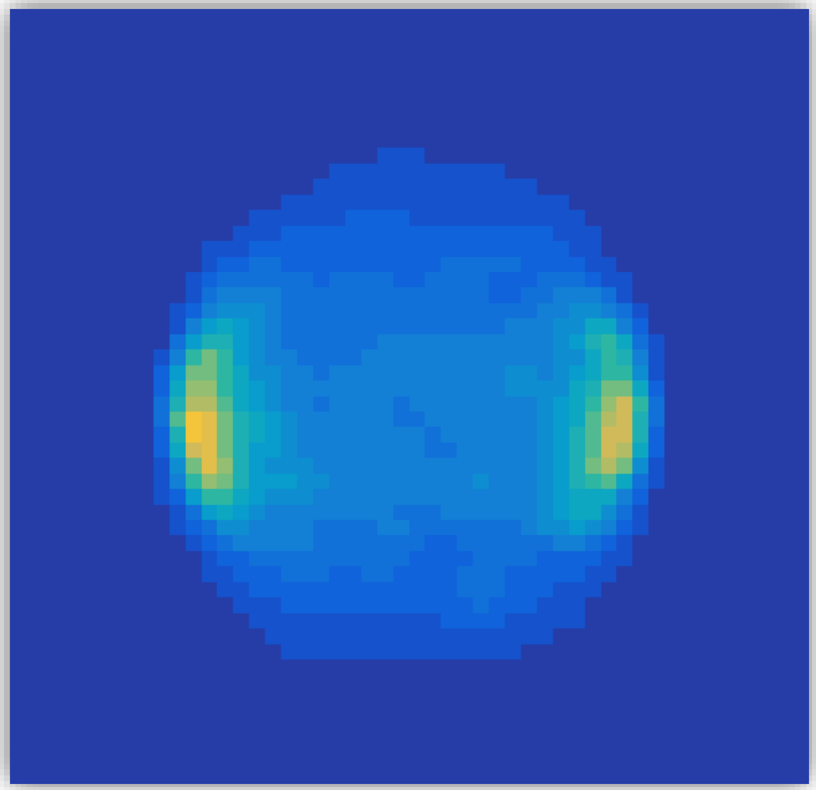




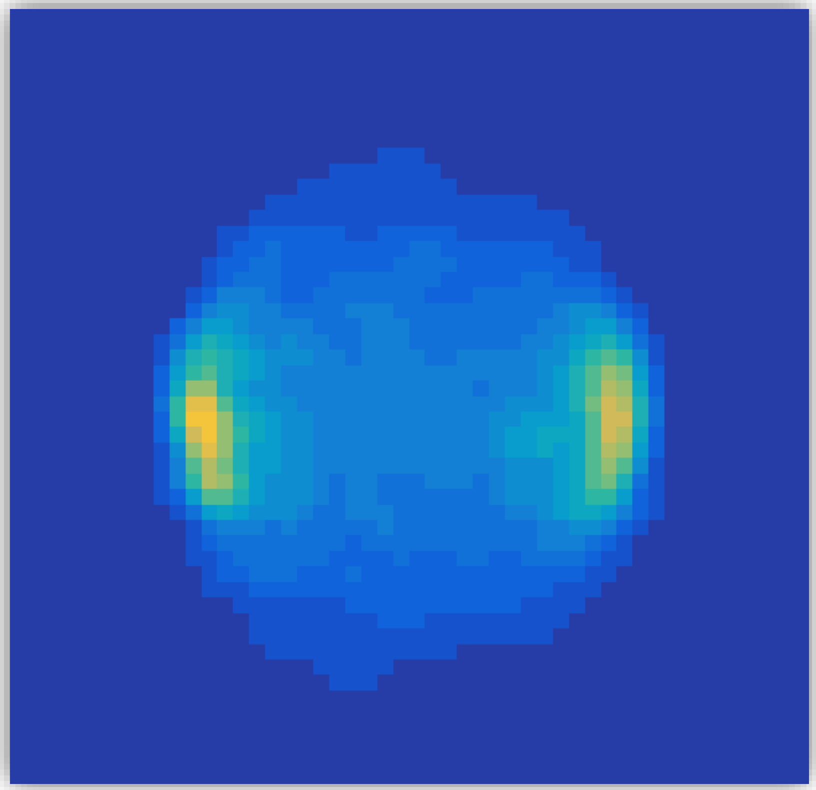
Is the GAN really absorbing the geometry?

- The purpose of this system is to produce simulations quickly to be used with a ML method
- To proof that we really need to see whether the GAN has “absorbed” the geometry
- First thing we can do is the see how the POCA estimation looks for this simulation

Geant4 generated - 16 mm thickness pipe



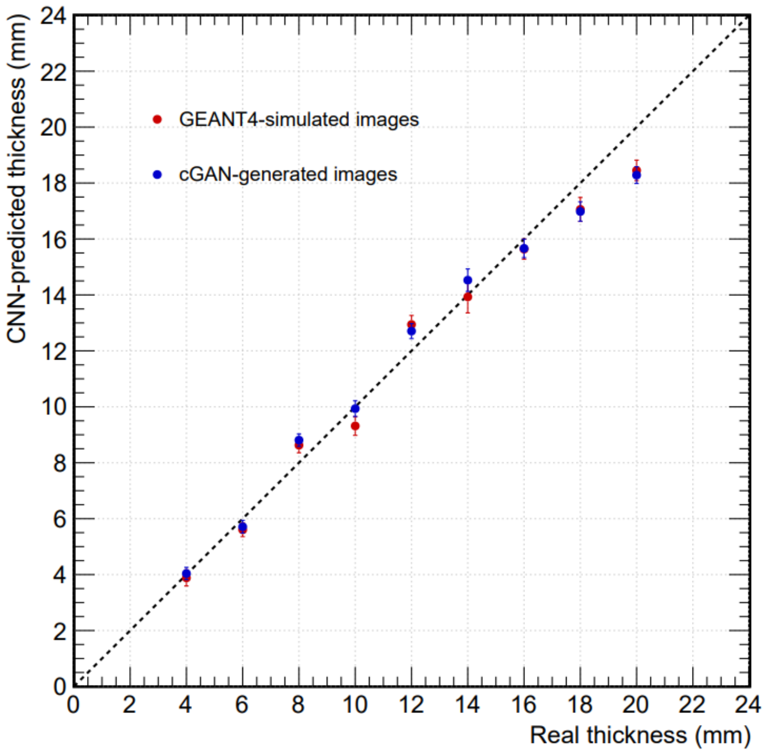
GAN generated - 16 mm thickness pipe



CNN for pipe thickness regression

- A convolutional neural network has been trained to predict the pipe thickness based on POCA
- Training dataset based on 90 POCA images per thickness (4, 6, 8, …, 18 mm thickness)
- Each POCA image is made with 300K muons based on CRY + GEANT4 simulation
- The CNN uses RestNET50 with 4 additional dense layers with 1024, 512, 512 and 256 nodes

The CNN scores similarly on both GEANT4 and GAN samples indicating the validity of the method



Conclusions

- We have explored the possibility to use GANs to generate fast MC simulation in muography
- Two different kinds of GANs tested: simple + Wasserstein, conditional GAN
  - Both are giving very good results in terms of similarity to the targeted distributions
  - The Wasserstein GAN seems to be in general more stable and easier to converge
  - The Wasserstein, conditional GAN is able to interpolate to non-trained thicknesses
- Our setup has tested only the muon propagation part of the simulation
  - If trained with real data from a real detector → capacity to learn the detector response
  - We are focusing on this right now as it would be a ML driven detector simulation
- A CNN has been devised to predict the parameter of interest in a muography problem
- The GAN samples are similar to GEANT4 for the CNN with a speed up of at least x50