LMU LUDWIG-MAXIMILIANS UNIVERSITY MUNICH

IEA-GAN: Intra-Event Aware GAN for the Fast Simulation of PXD Background at Belle II

Hosein Hashemi, Nikolai Hartmann, Thomas Kuhr, Sahand Sharifzadeh, Matej Srebre

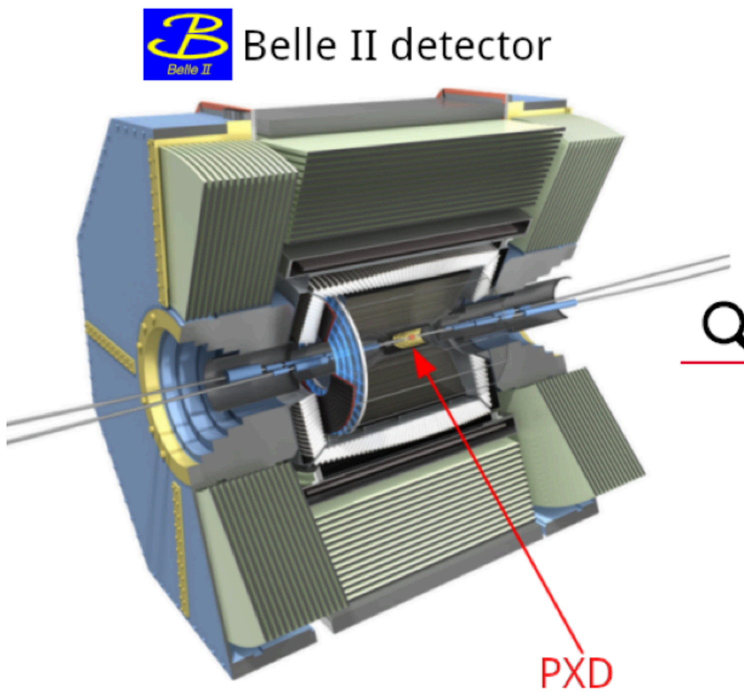
Ludwig-Maximilians-Universität München

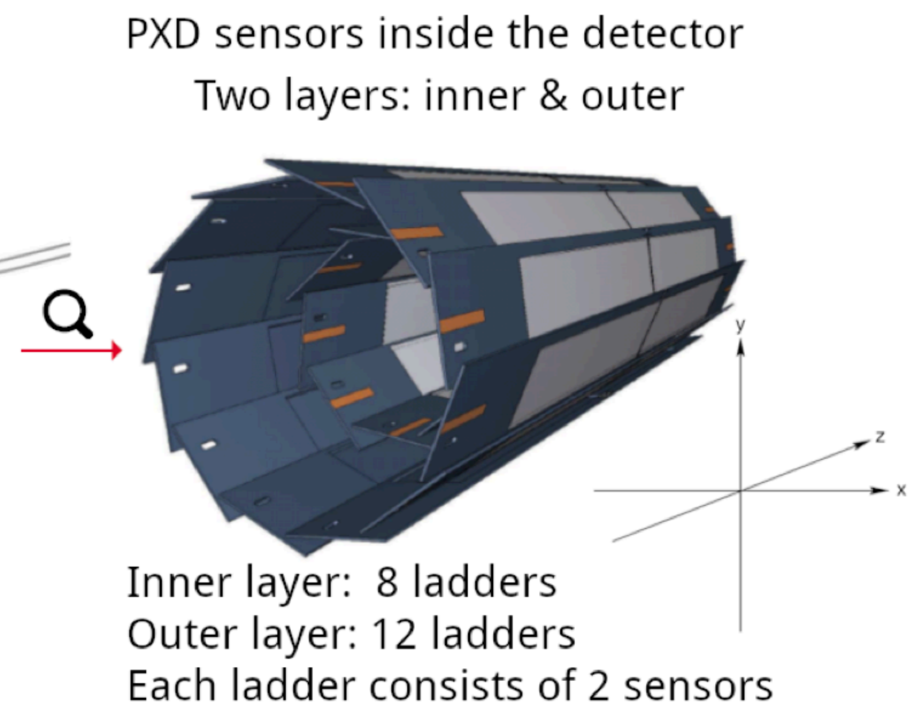
The ORIGINS Excellence Cluster

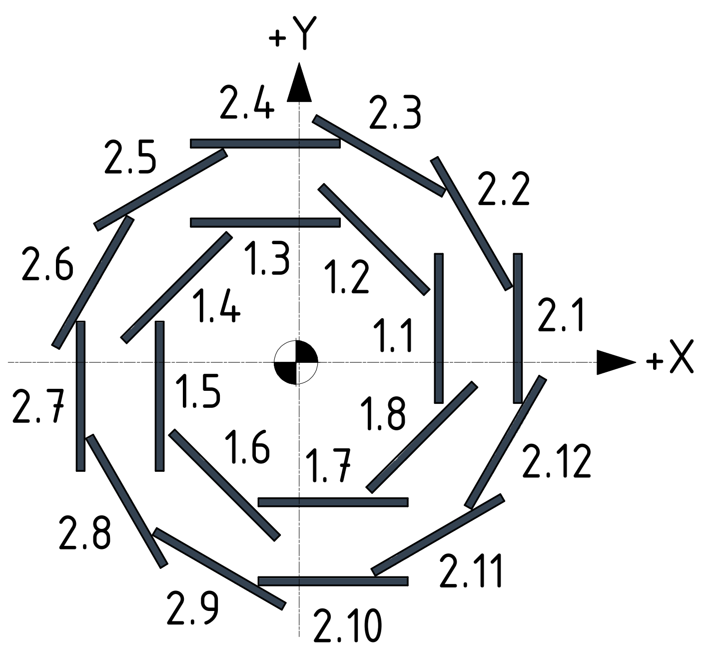
Bundesministerium für Bildung und Forschung

Introduction

- **The Pixel Vertex Detector (PXD)** is the innermost semi-conductor sub-detector at Belle II.
- The sensitive area of the PXD is assembled from **40 modules**, where each module consists of a **250** $\times$ **768** pixel matrix of the pixel sensors.
- **The inner layer:** 16 modules implemented into 8 ladders
- **The outer layer:** 24 modules implemented into 12 ladders



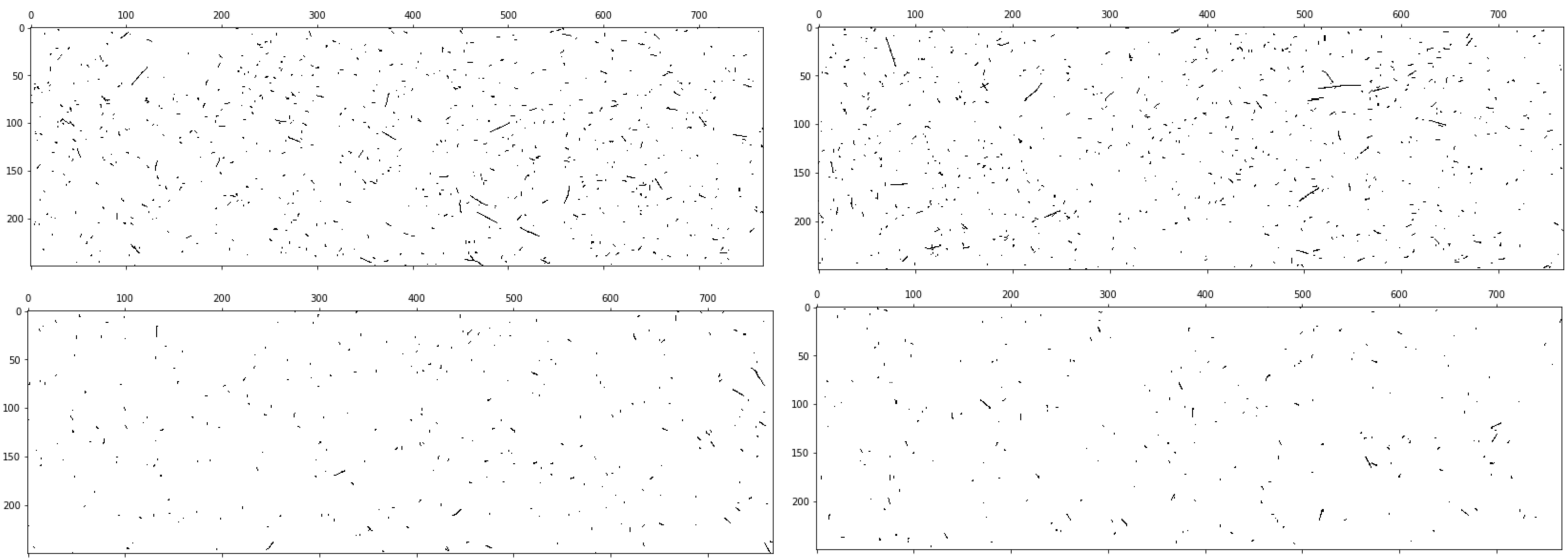




Backgrounds

- **The PXD hits are coming from two sources:**
  - **Signal Decays:**
    - A. particles which originate from the physics processes of interest.
  - **Backgrounds:**
    - A. **Beam-induced**: intra-beam scattering, Beam-Gas scattering, synchrotron radiation
    - B. **Luminosity dependent:** Radiative Bhabha scattering, two-photon process

Colour-reversed real (simulated) image



Beam direction

Problem

- **Realistic detector simulation has to take into account effects from background processes**
  - Simulation requires many PXD hitmaps with statistically independent background.
  - Overlay hits from simulated background or random trigger data to hits from signal MC.
  - PXD hits have the highest storage consumption, almost 200 kB per event cost.
  - Requires distributing over all sites where MC is produced.

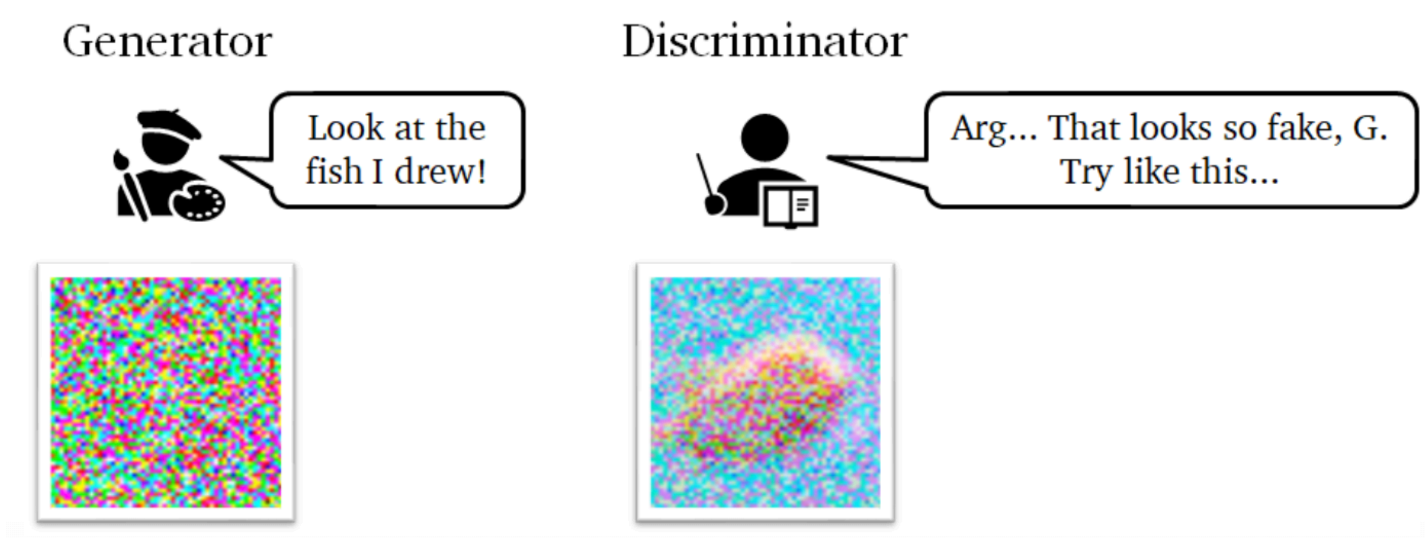
Problem

- **Realistic detector simulation has to take into account effects from background processes**
  - Simulation requires many PXD hitmaps with statistically independent background.
  - Overlay hits from simulated background or random trigger data to hits from signal MC.
  - PXD hits have the highest storage consumption, almost 200 kB per event cost.
  - Requires distributing over all sites where MC is produced.

- **Solution:** Generating and compressing the background data on the way of analysis with GANs instead of storing them.

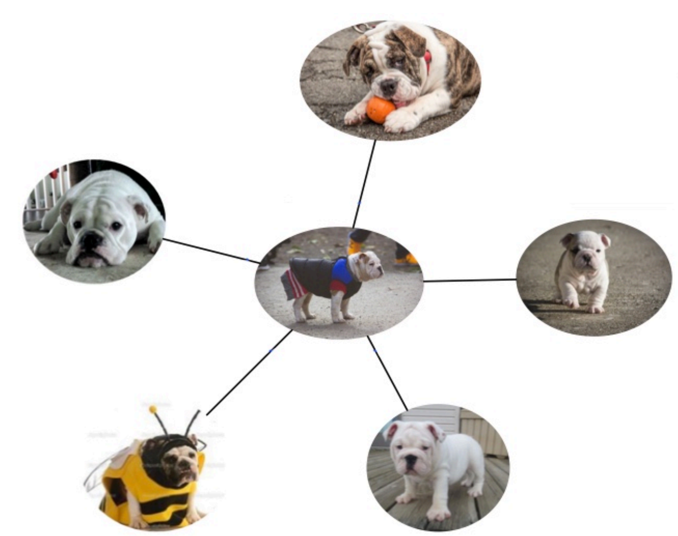
Generative Adversarial Network

- Generate PXD background events with Generative Adversarial Network **(GAN)**
- **Whats is GAN?**



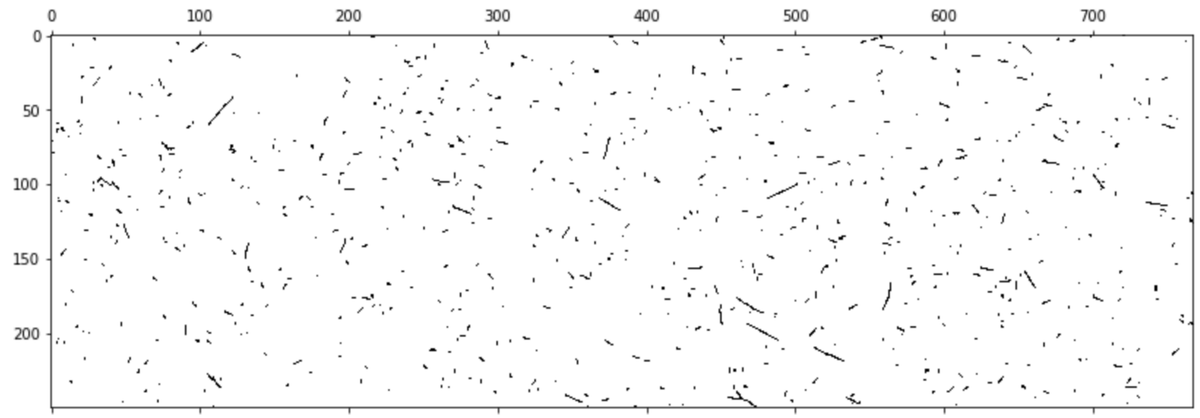
The GAN game

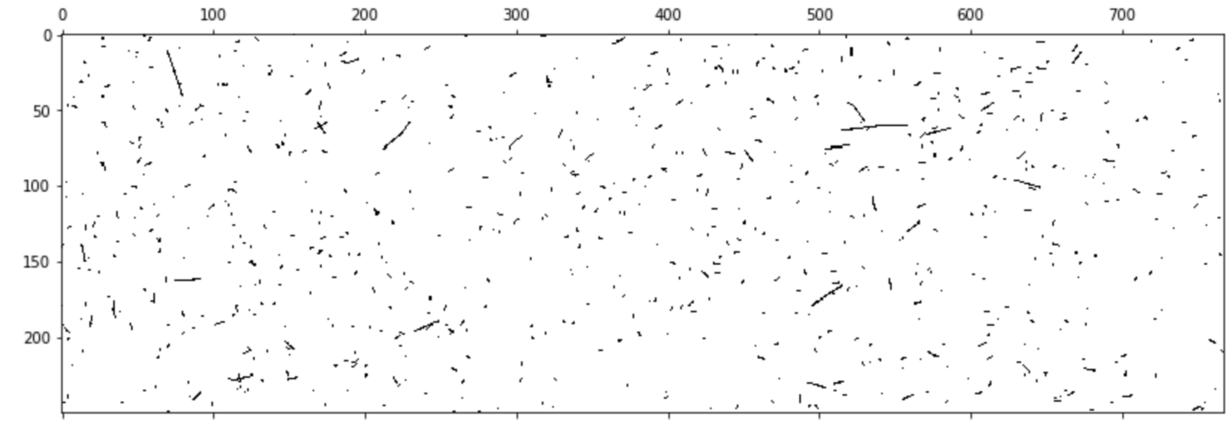
- **class-conditional GAN :** The type of animal is the condition (class)
- **Fine-grained class-conditional image generation:**
  - **A.** The classes show both statistical and semantic similarity
  - **B.** Similar datasets: The Stanford Cars, iNaturalist
  - **C.** The objective is to create objects from subordinate categories such as breeds of dogs or models of cars.
  - **D.** The small inter-class and large intra-class variation inherent to fine-grained image analysis makes it a challenging problem.

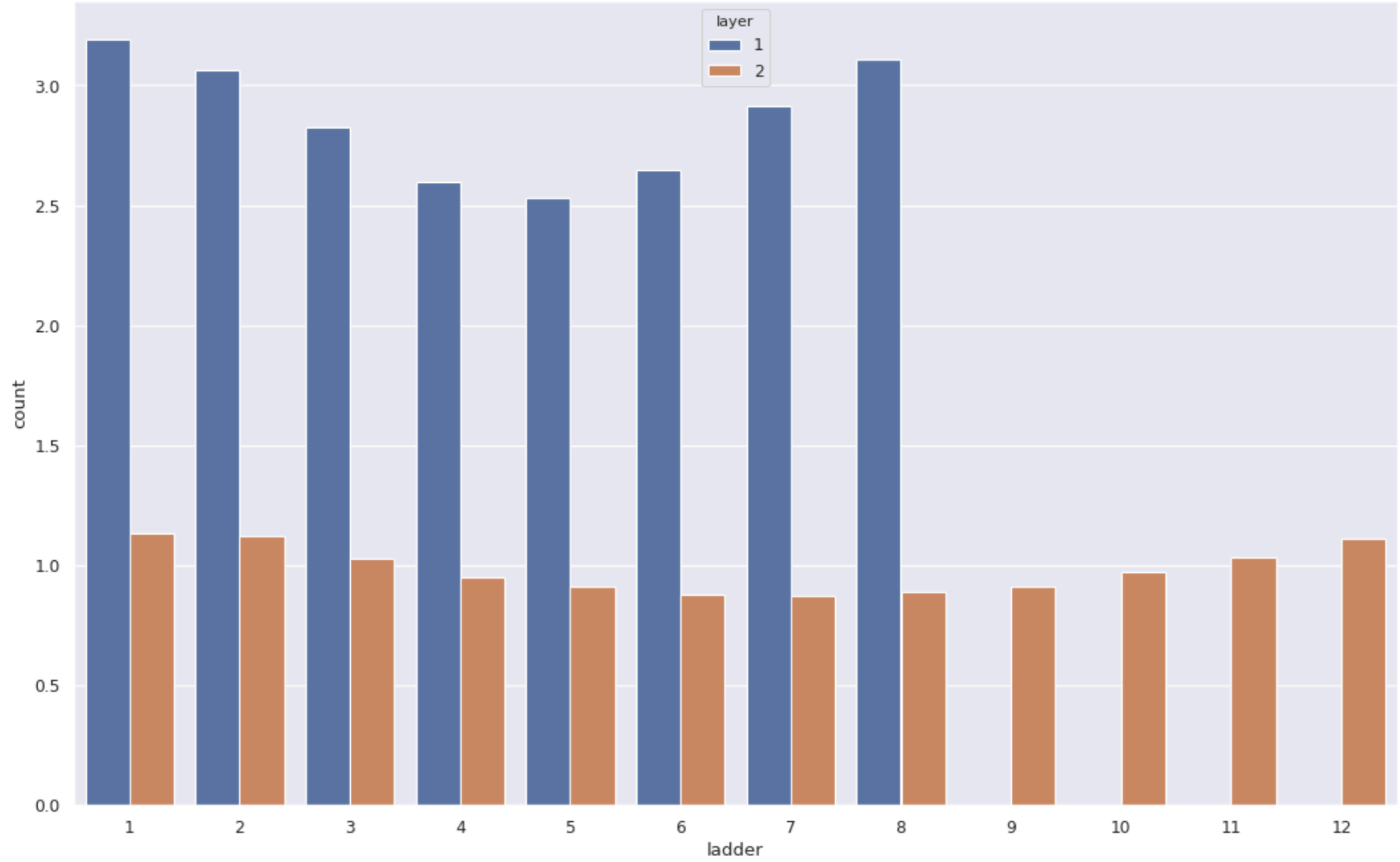


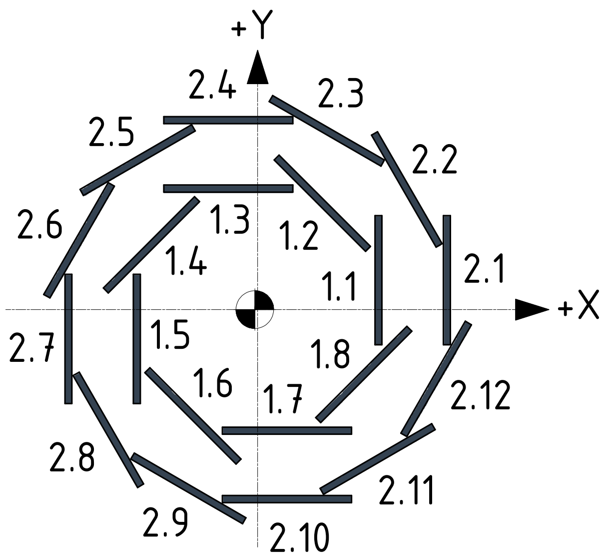
Conditional GAN

- **Using spatial class-conditions based on the sensor number 1-40:**
  - ☑ Training Data: Simulated beam background events, nominal phase 3, Bkgx2.
  - ☑ Objective: Generation of sensor-dependent images to capture all intra-event correspondence among images



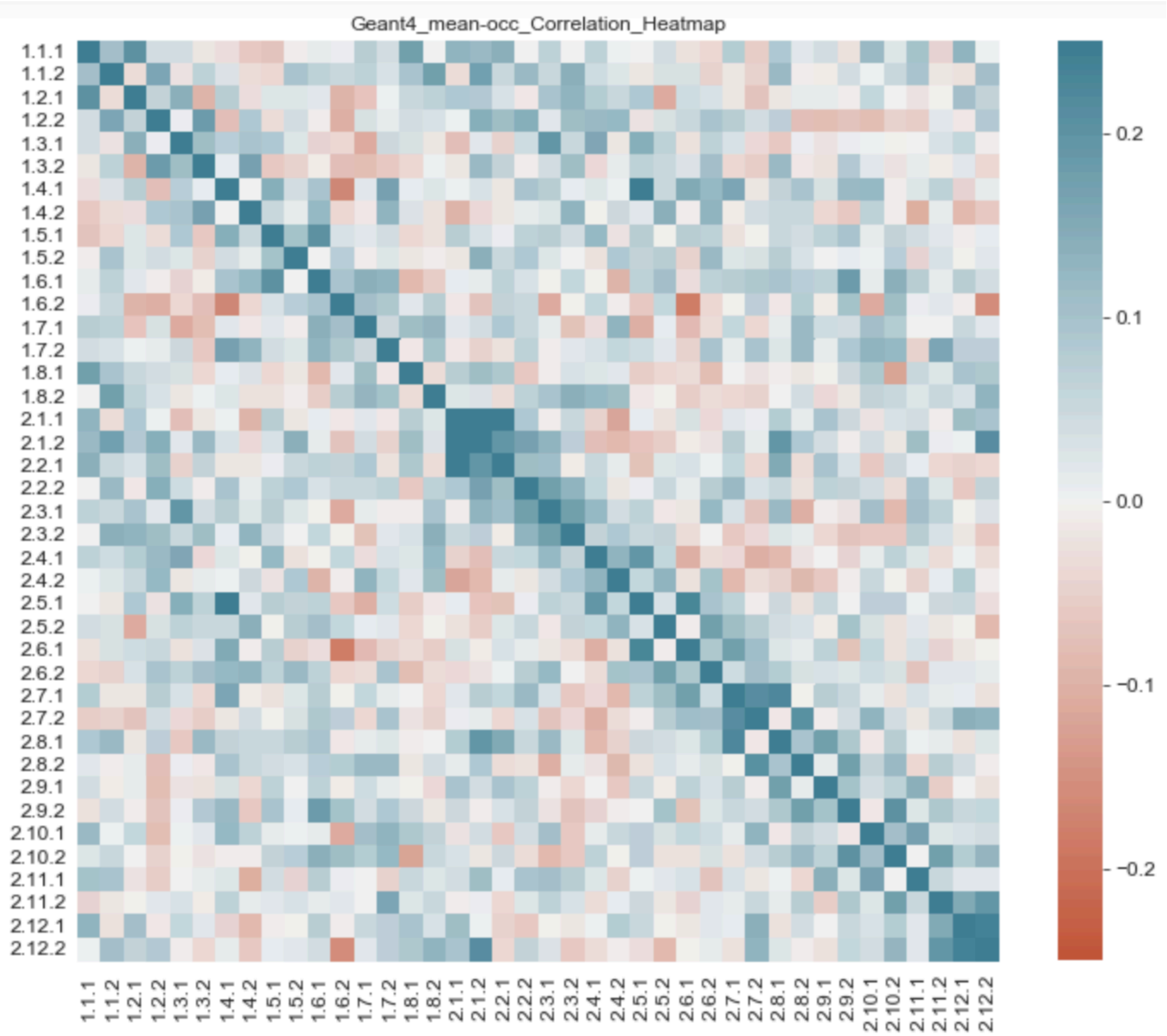


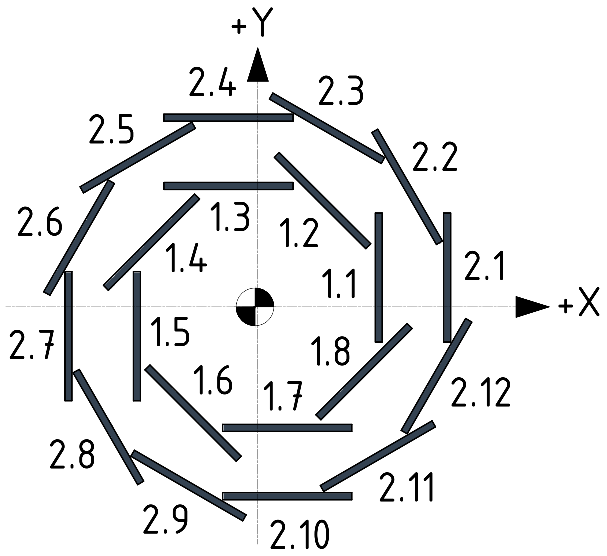




Conditional GAN

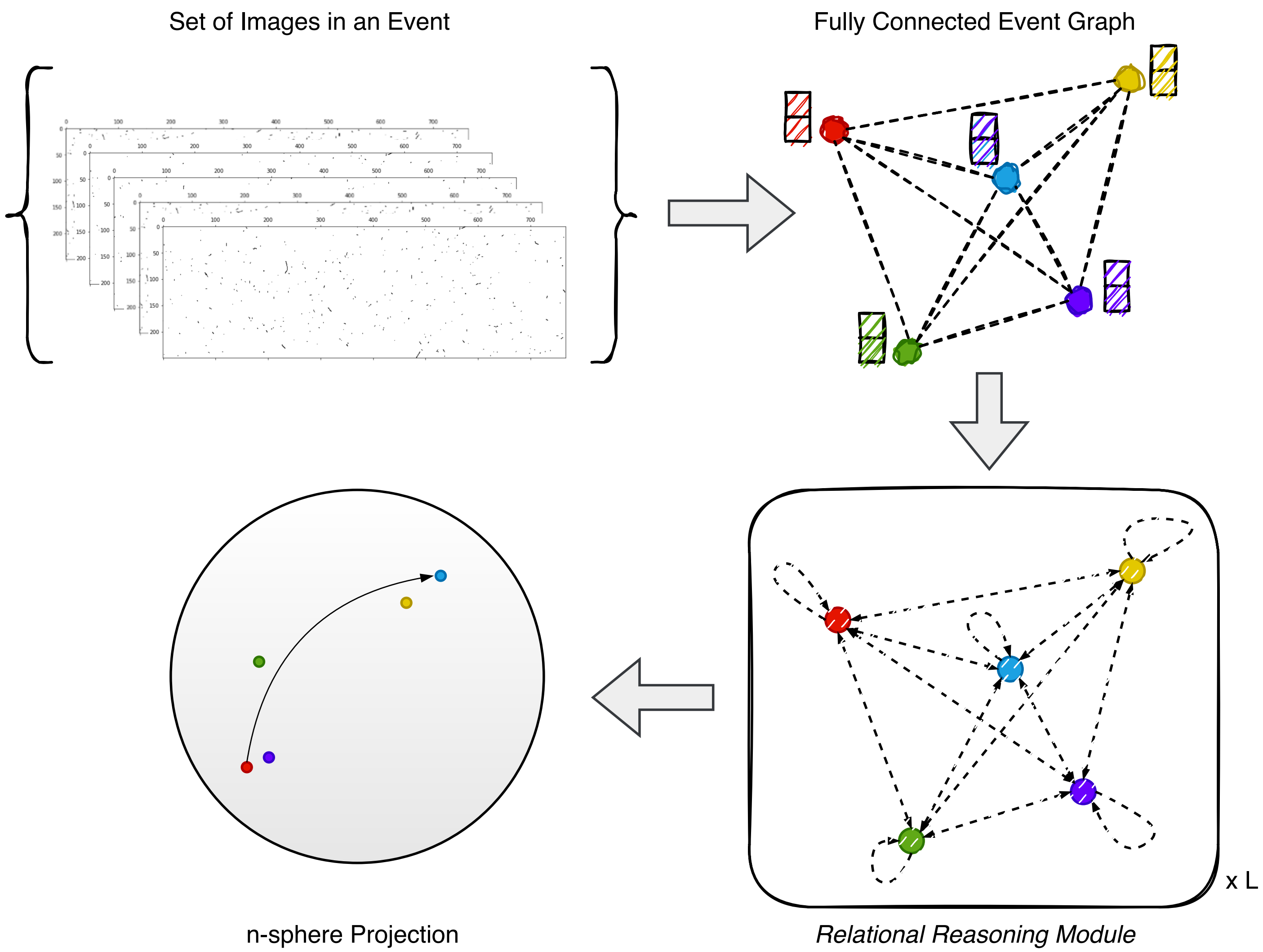
- **Using spatial class-conditions based on the sensor number 1-40:**
  - ☑ Training Data: Simulated beam background events, nominal phase 3, Bkgx2.
  - ☑ Objective: Generation of sensor-dependent images to capture all intra-event correspondence among images



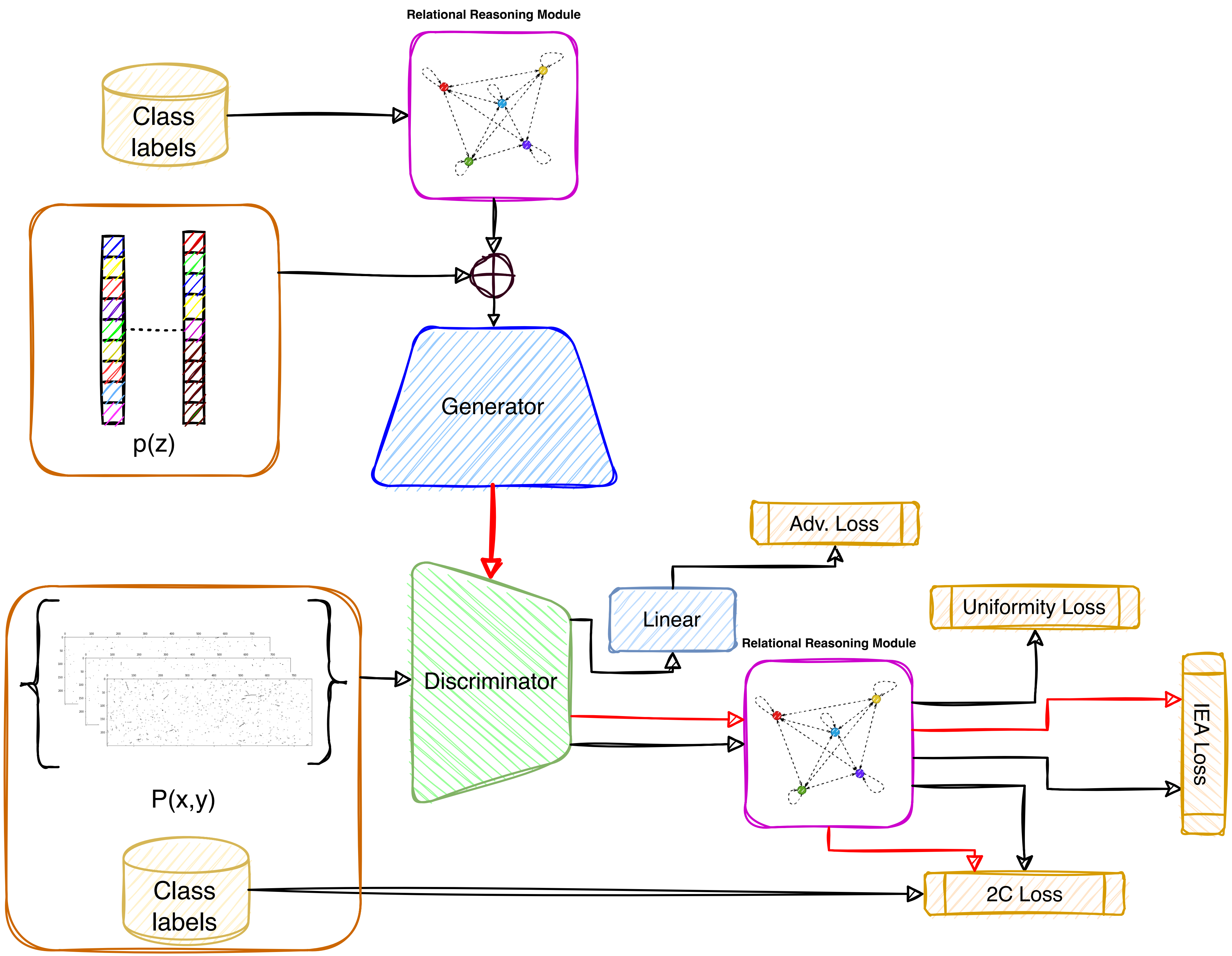


IEA-GAN Model (prologue)

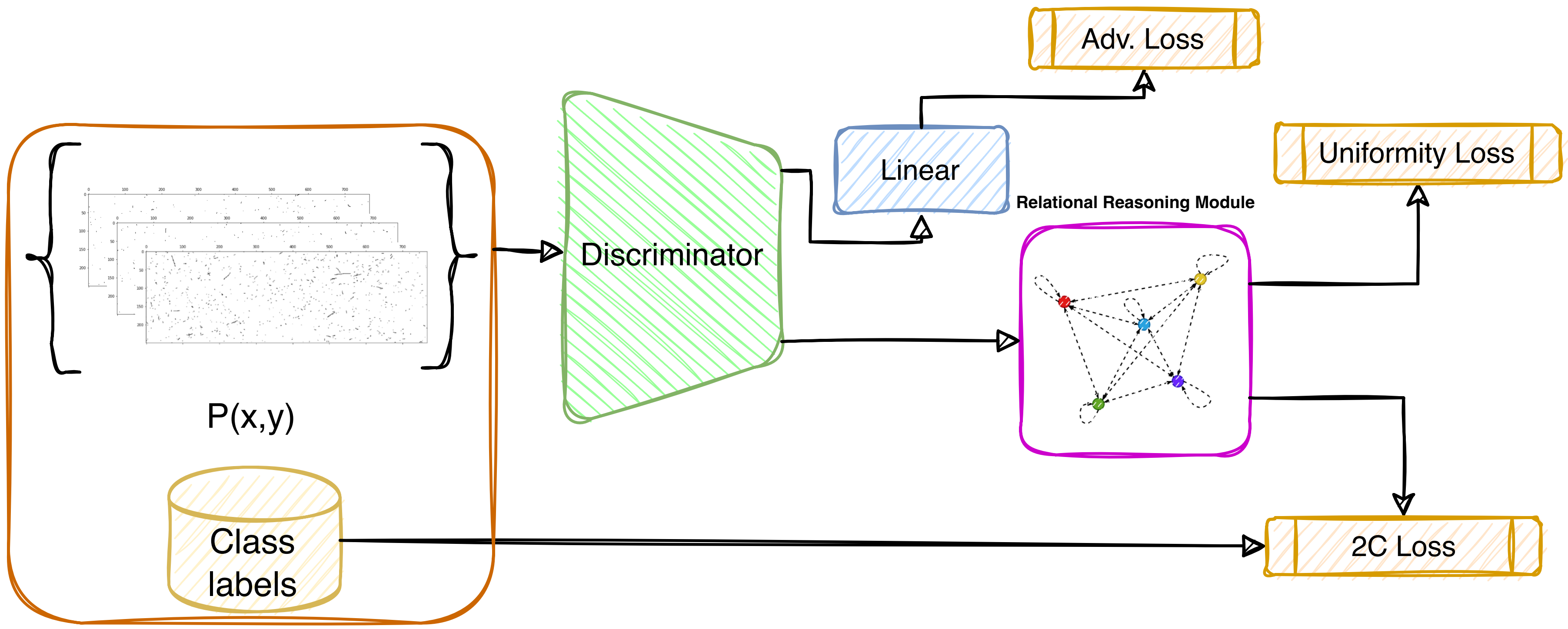
- **Approximating the concept of an "Event":**
  - Defining an image per class sampler (**generating event by event**) and shuffling within each batch (event).
  - **Intra-event relational reasoning:** Considering intra-event correlation for both the discriminator's image embeddings and the generator's class embedding. Using a Relational Reasoning Module over the samples in an event to weight the importance of each image with respect to each other. With this, **the model will understand if each image has any relation to one another in a single event.**



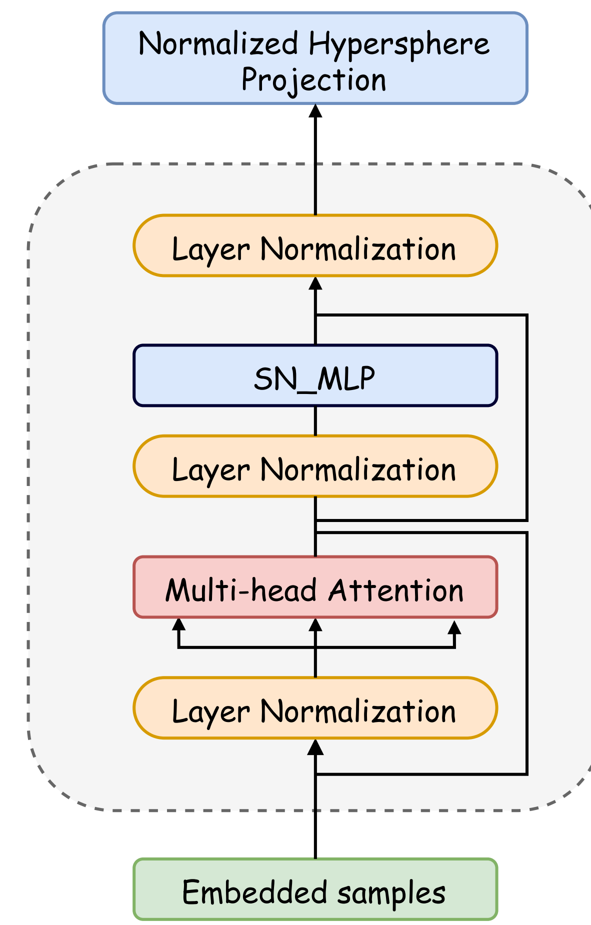
IEA-GAN Model



IEA-GAN Model (Discriminator)



Relational Reasoning Module



Hypersphere dimension: 1024
SN-MLP dimension: 512
Number of Heads: 4
Number of Layers: 1

$h(.)$: *Relational embedding*
$e(.)$: *proxy (class embedding)*

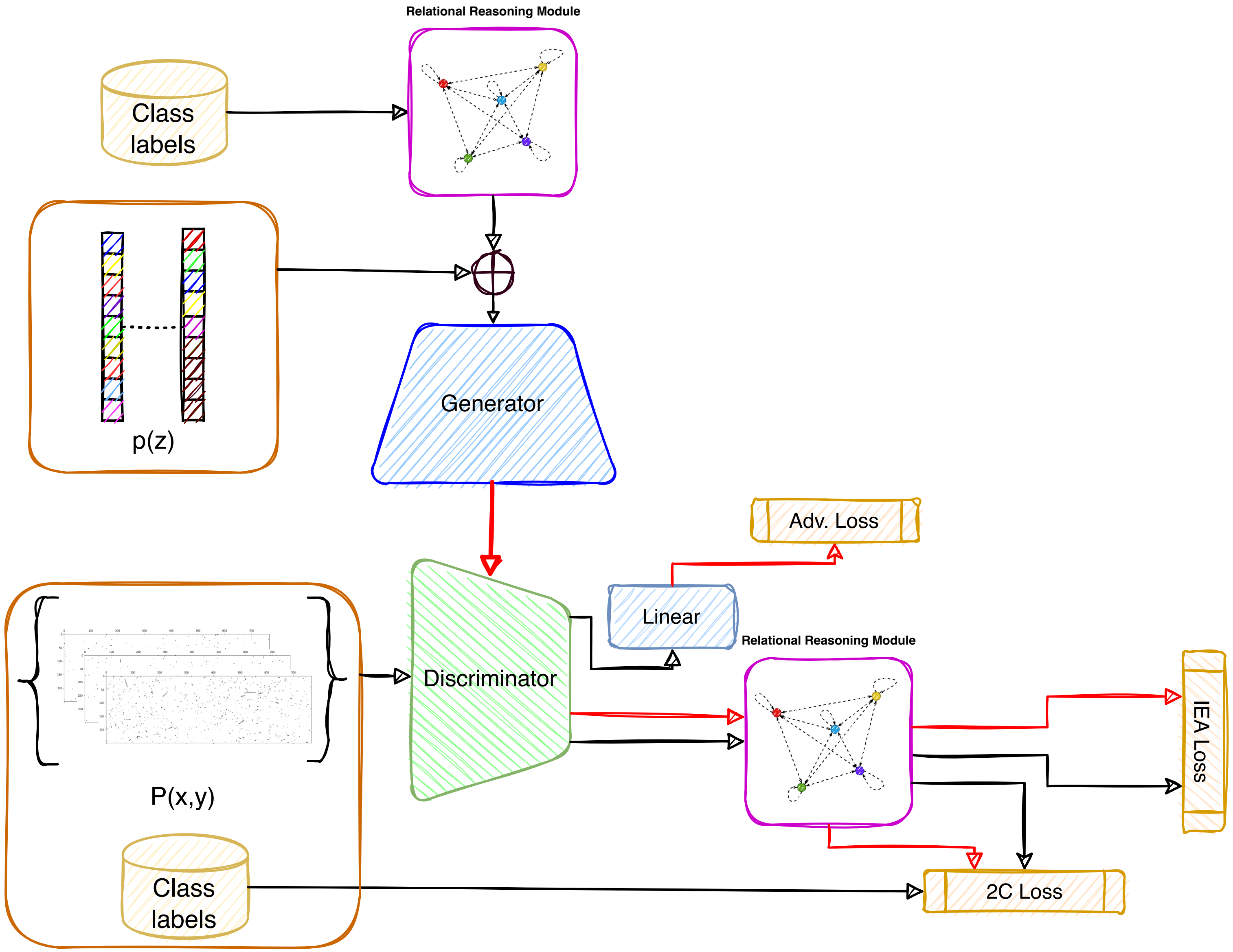
$$L_{dis} = L_{Adv} + \lambda_{2C} L_{2C} + \lambda_{uniform} L_{uniform}$$

$$L_{2C}(x_i, y_i; t) = -\frac{1}{N} \sum_{i=1}^{N} \log \left(\frac{\exp(h(x_i)^T e(y_i)/t)}{\exp(h(x_i)^T e(y_i)/t) + \sum_{k=1}^{m} \mathbf{1}_{k \neq i} \,.\, \exp(h(x_i)^T h(x_k)/t)} \right)$$

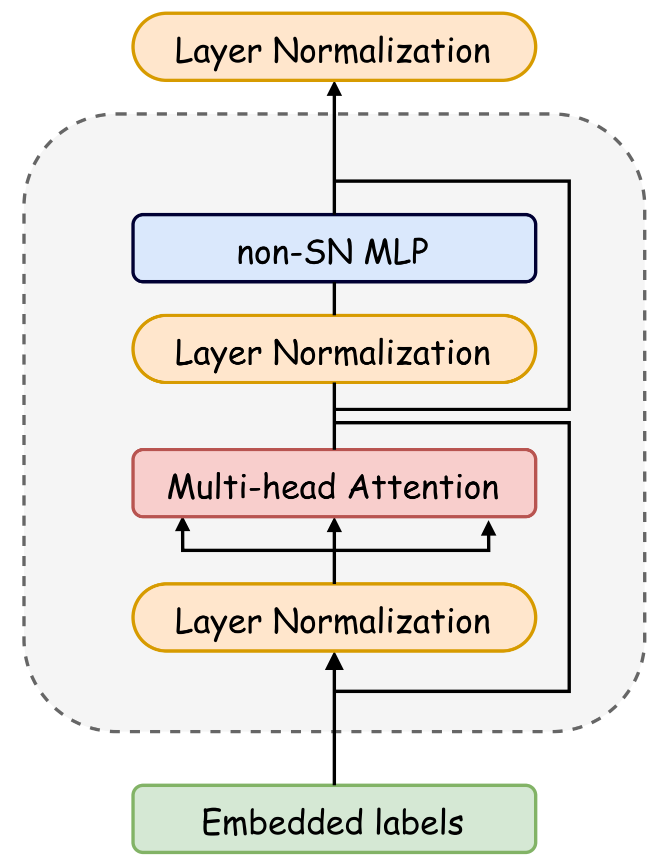
$$L_{uniform}(h; t) = \log \mathbb{E}_{x_i, x_j \sim p_{data}} [\exp(-t \| h(x_i) - h(x_j) \|_2^2)]$$

- **By imposing uniformity condition over the feature vectors on the unit hypersphere, they preserve as much information as possible since the uniform distribution carry high entropy.**

IEA-GAN Model (Generator)



Relational Reasoning Module



$$L_{gen} = L_{Adv} + \lambda_{2C} L_{2C} + \lambda_{IEA} L_{IEA}$$

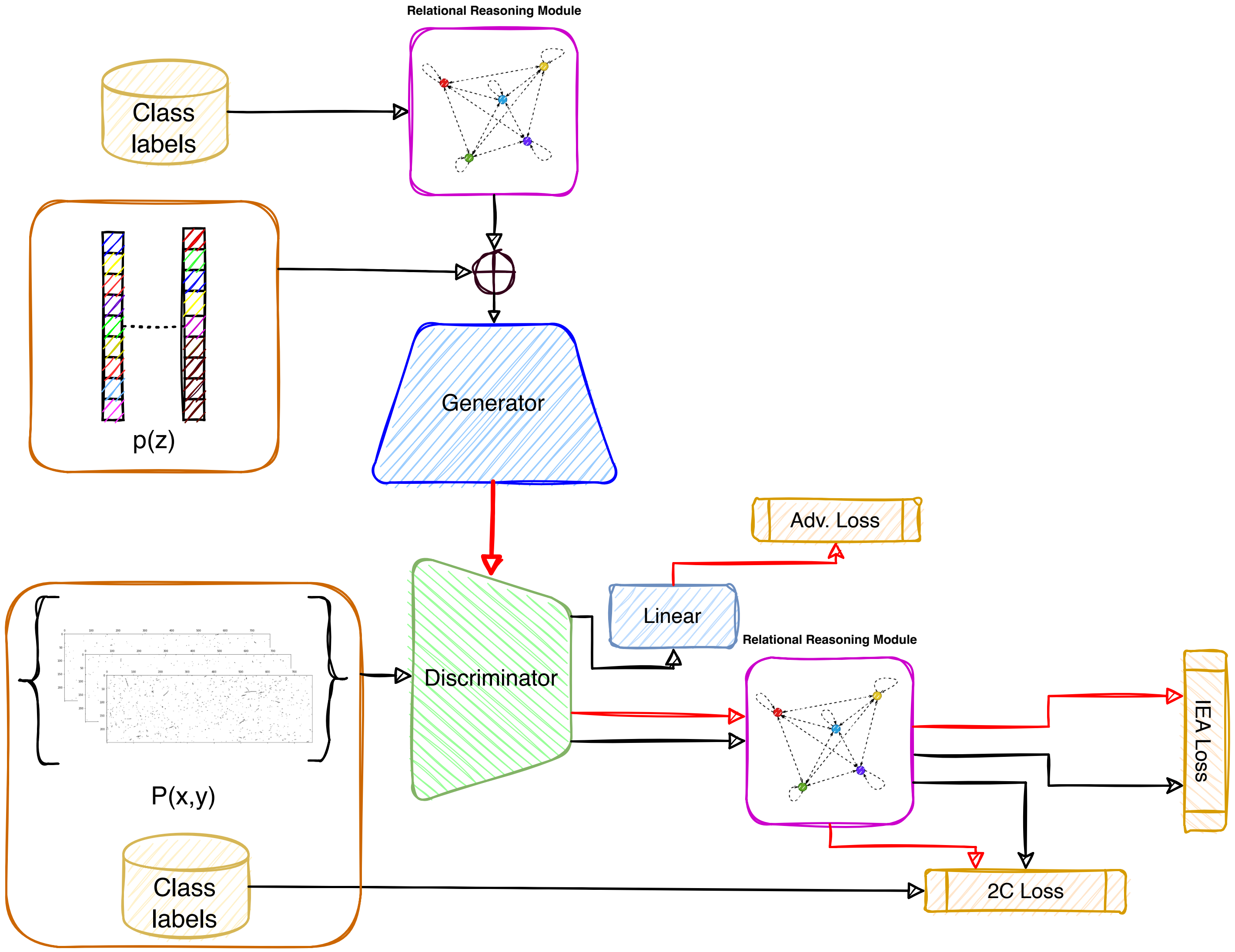
$$L_{2C}(x_i, y_i; t) = -\frac{1}{N} \sum_{i=1}^{N} \log \left(\frac{\exp(h(x_i)^T e(y_i)/t)}{\exp(h(x_i)^T e(y_i)/t) + \sum_{k=1}^{m} \mathbf{1}_{k \neq i} \, . \exp(h(x_i)^T h(x_k)/t)} \right)$$

$$L_{IEA}(x_f, x_r) = D_{KL}\Big(\sum_{i,j} \sigma(h(x_i^{(r)}) h(x_j^{(r)})^\top) \,|\, \sigma(h(x_i^{(f)}) h(x_j^{(f)})^\top)\Big)$$

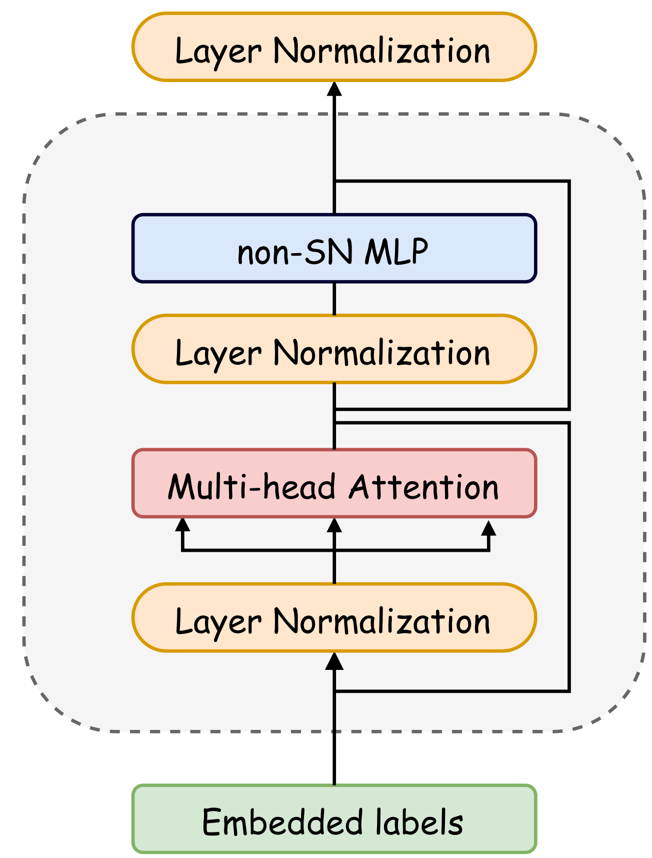
Hypersphere dimension: 128
MLP dimension: 128
Number of Heads: 2
Number of Layers: 1

$h(.)$: *Relational embedding*
$e(.)$: *proxy (class embedding)*
$\sigma(.)$: *Softmax function*
$x^{(f)}$: *generated images*
$x^{(r)}$: *real images*

IEA-GAN Model (Generator)



Relational Reasoning Module



$$L_{IEA}(x_f, x_r) = D_{KL}\Big(\sum_{i,j} \sigma(h(x_i^{(r)})h(x_j^{(r)})^\top) \,|\, \sigma(h(x_i^{(f)})h(x_j^{(f)})^\top)\Big)$$

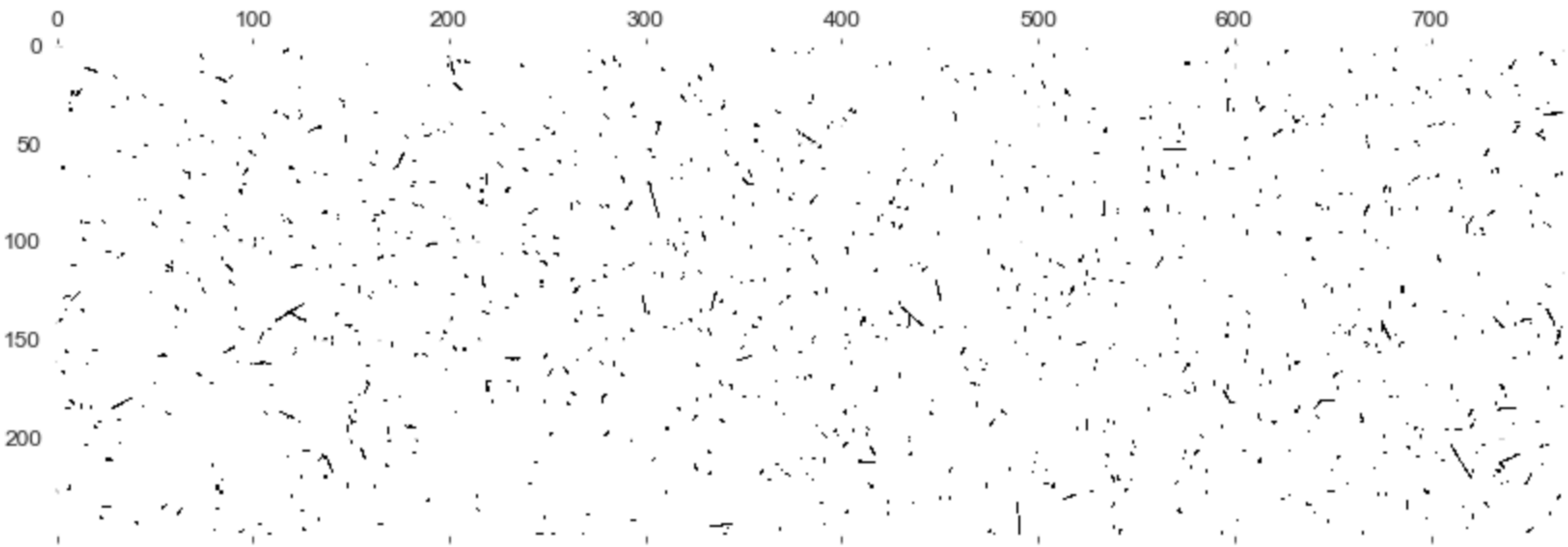
- **Upon minimising it, we are putting a self-supervised penalising system over the intra-event awareness of the the generator by encouraging it to look for more detailed connections among the images and be sensitive to even a small differences.**
- **In the end we want to maximise the agreement of data points on two unit hyperspheres of real image and generated image embeddings.**

Hypersphere dimension: 128
MLP dimension: 128
Number of Heads: 2
Number of Layers: 1

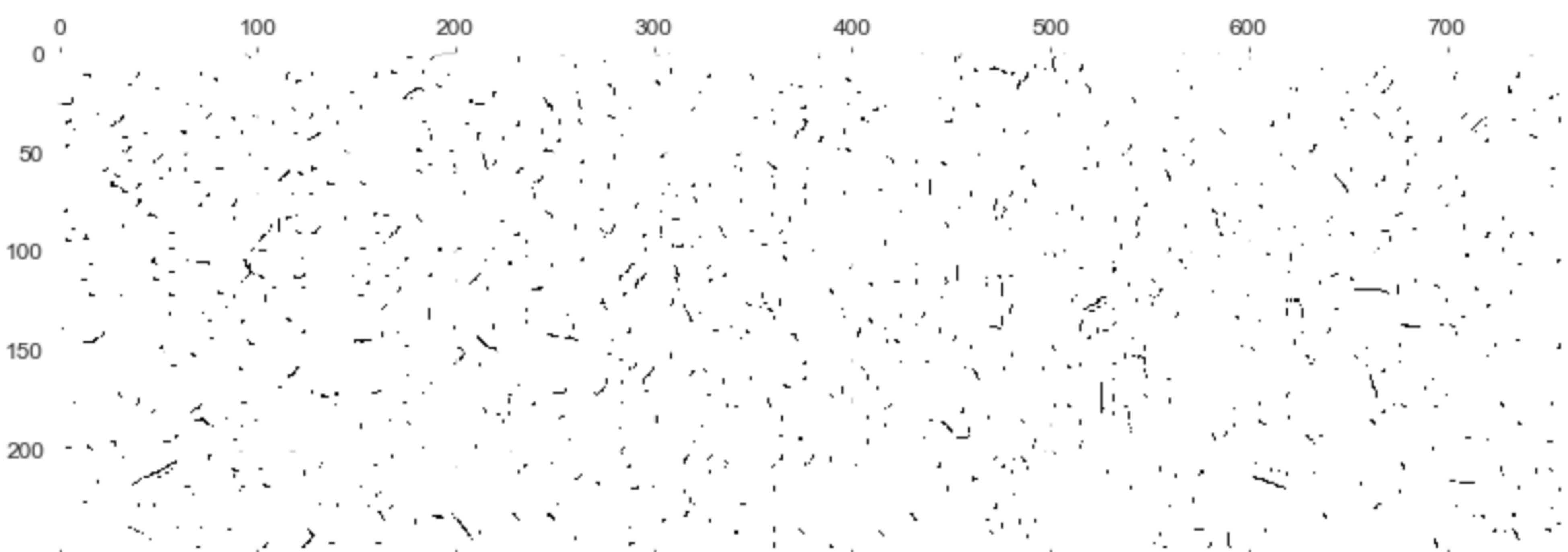
$h(.)$: *Relational embedding*
$e(.)$: *proxy (class embedding)*
$\sigma(.)$: *Softmax function*
$x^{(f)}$: *generated images*
$x^{(r)}$: *real images*

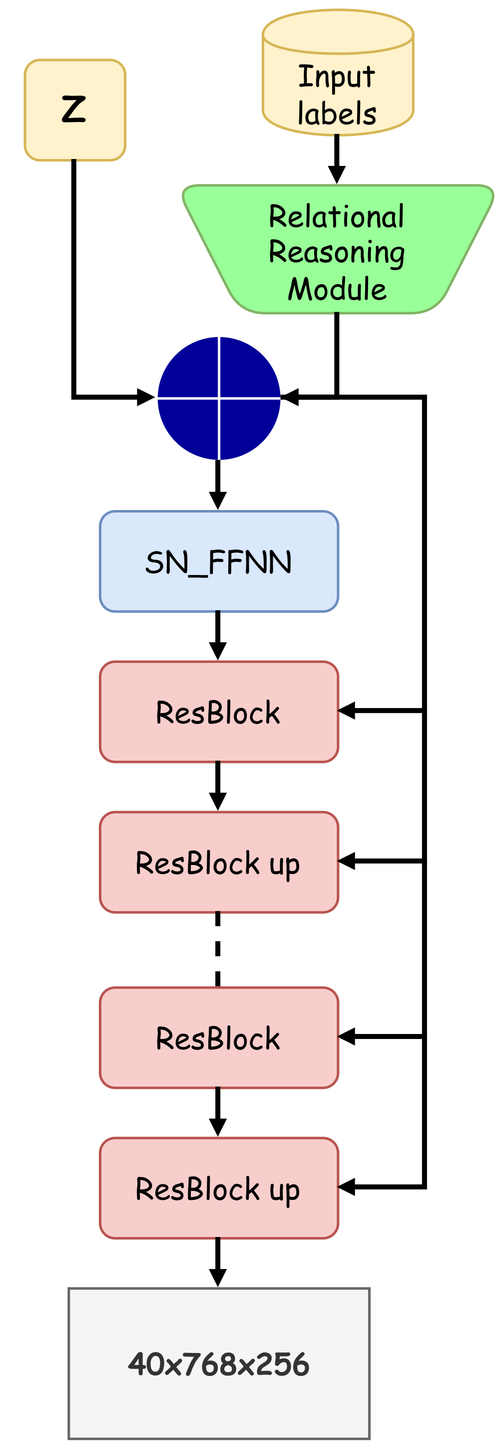
IEA-GAN Model

Colour-reversed Geant4 simulated image



Colour-reversed IEA-GAN generated image

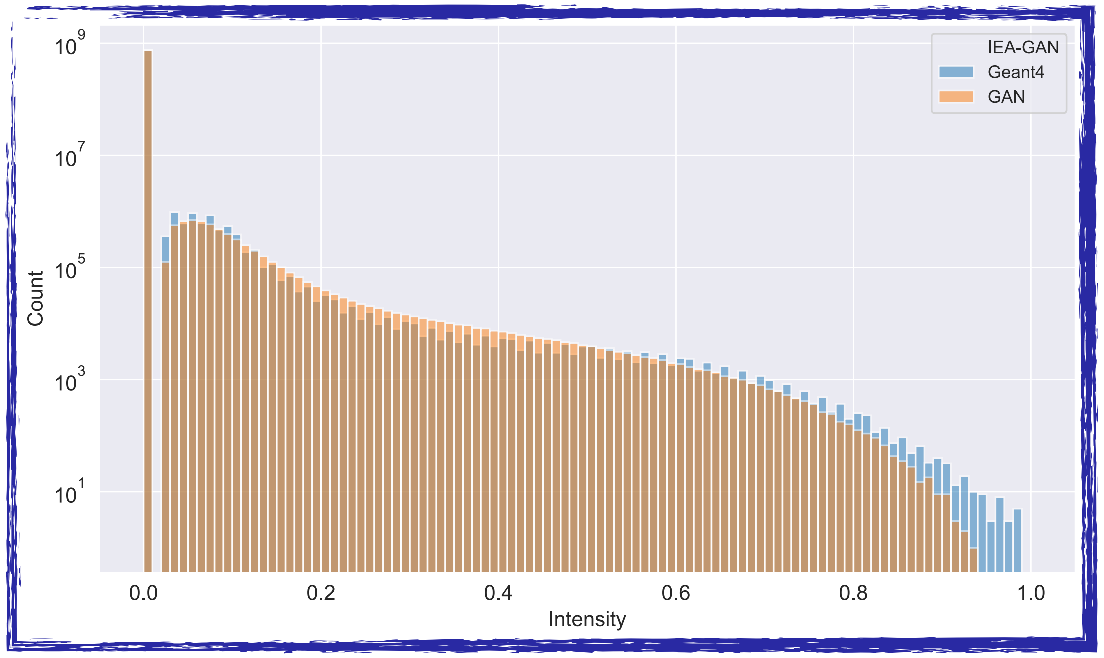


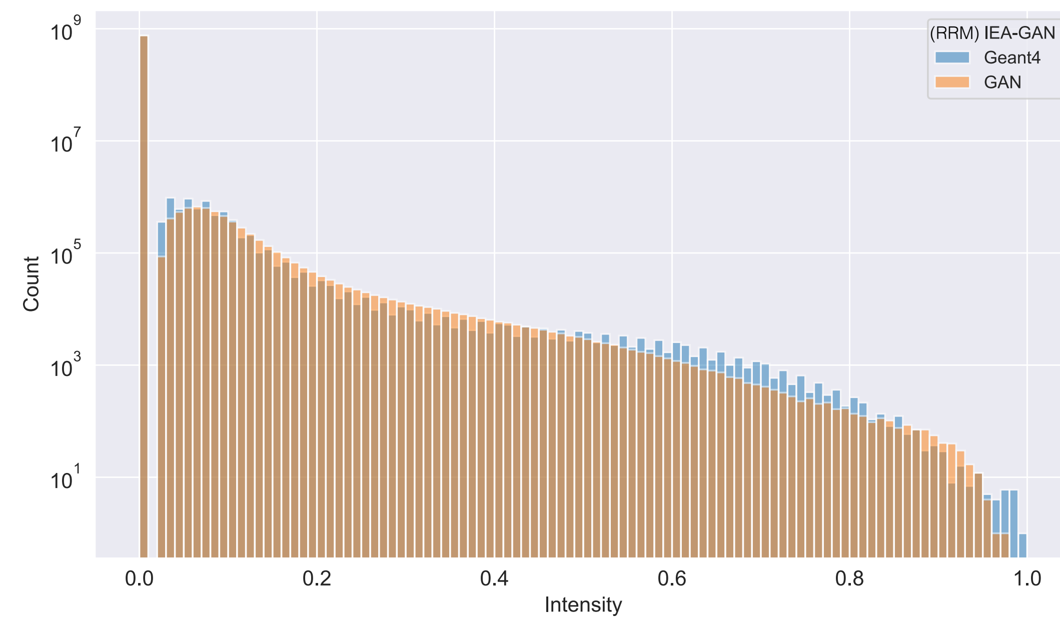


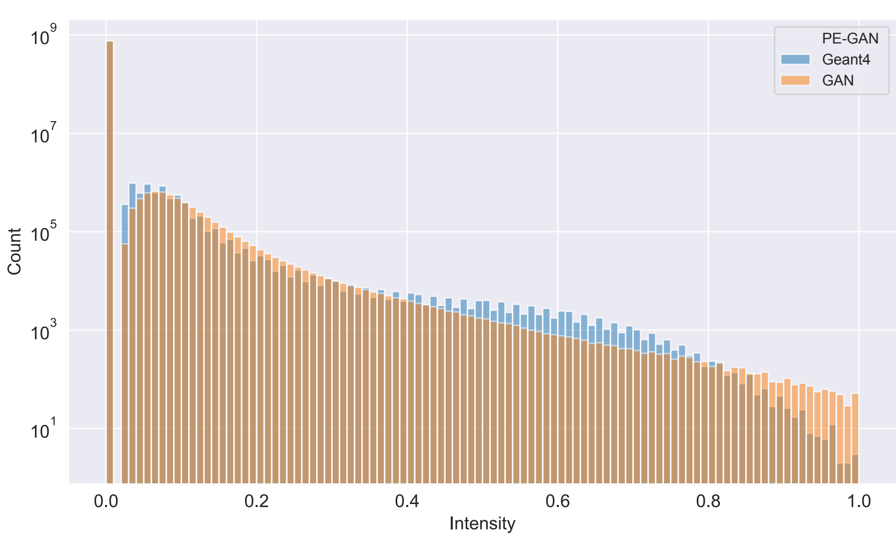
Validation of generated PXD images

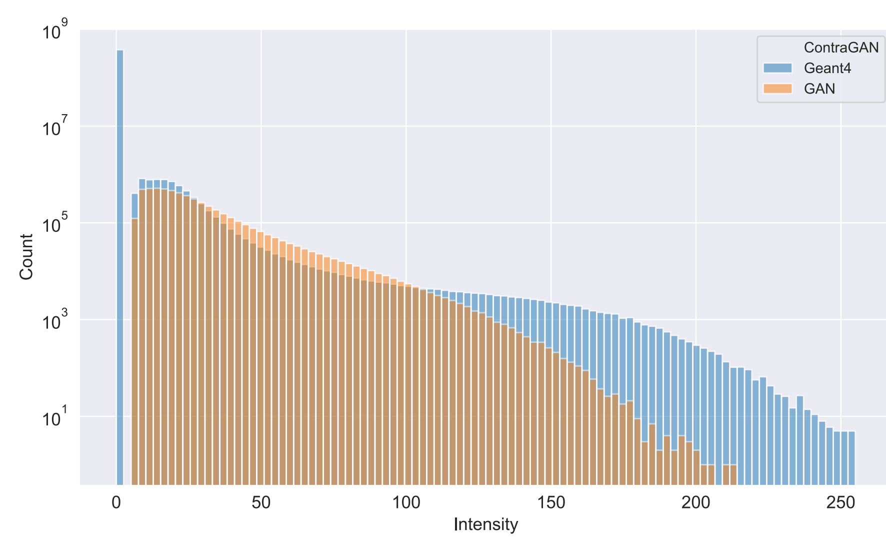
- **Validation Metrics over the test set:**
  - ☑ **Pixel Energy above the threshold:**

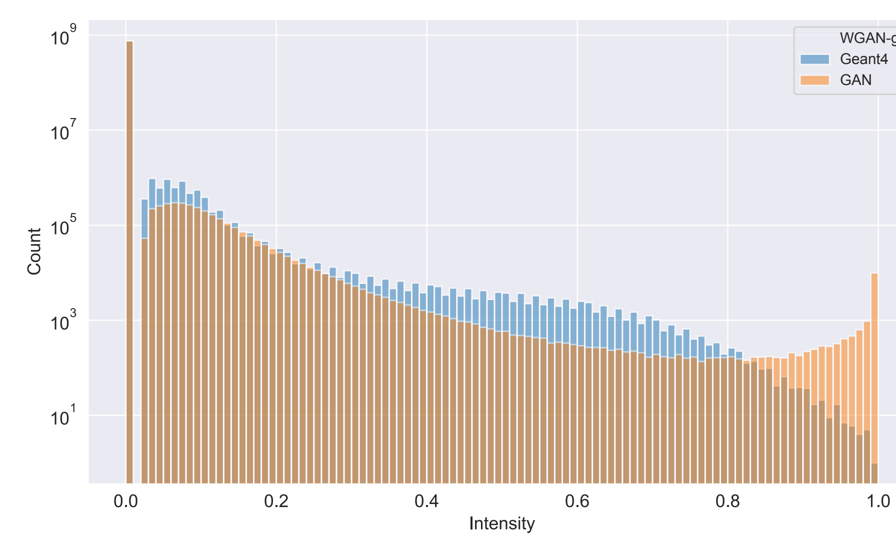
- Pixel value 0 means complete darkness.
- The fired pixels are only read out if their value exceeds a threshold, 7 ADU.











Validation of generated PXD images

- **Validation Metrics over the test set:**
  - **FID:**
    - FID is one of the most popular metrics for measuring the feature distance between the real and the generated images. Frechet Distance is used to compute the distance between two "multivariate" normal distribution. For a "univariate" normal distribution Frechet Distance is given as

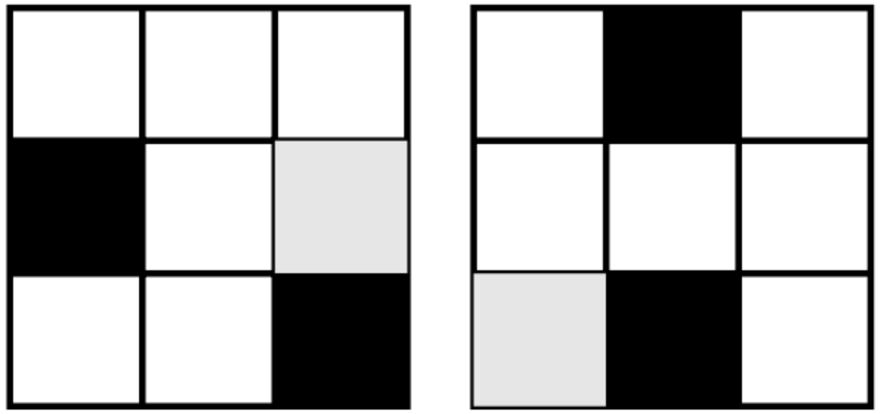
$$d^2(x_r, x_f) = (\mu_r - \mu_f)^2 + (\sigma_r - \sigma_f)^2.$$

    - The use of activations of the last layer from the Inception-V3 model trained on the PXD images to summarise each image, gives the score. The lower the FID the better the image diversity and Fidelity.

	WGAN	PE-GAN	ContraGAN	IEA-GAN (RRM)	IEA-GAN (RRM and IEAloss)
FID	12.09	5.84	2.84	2.16	**1.7**

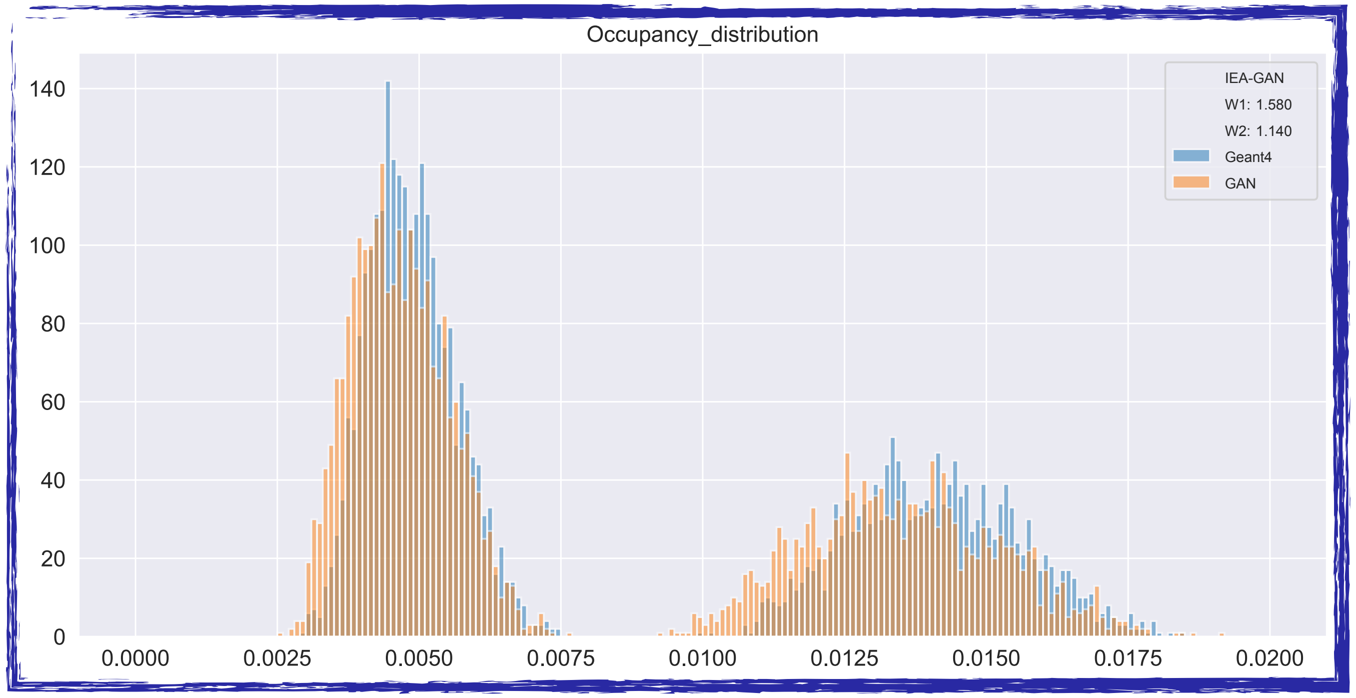
Table 1: FID comparison between models and ablation of modules

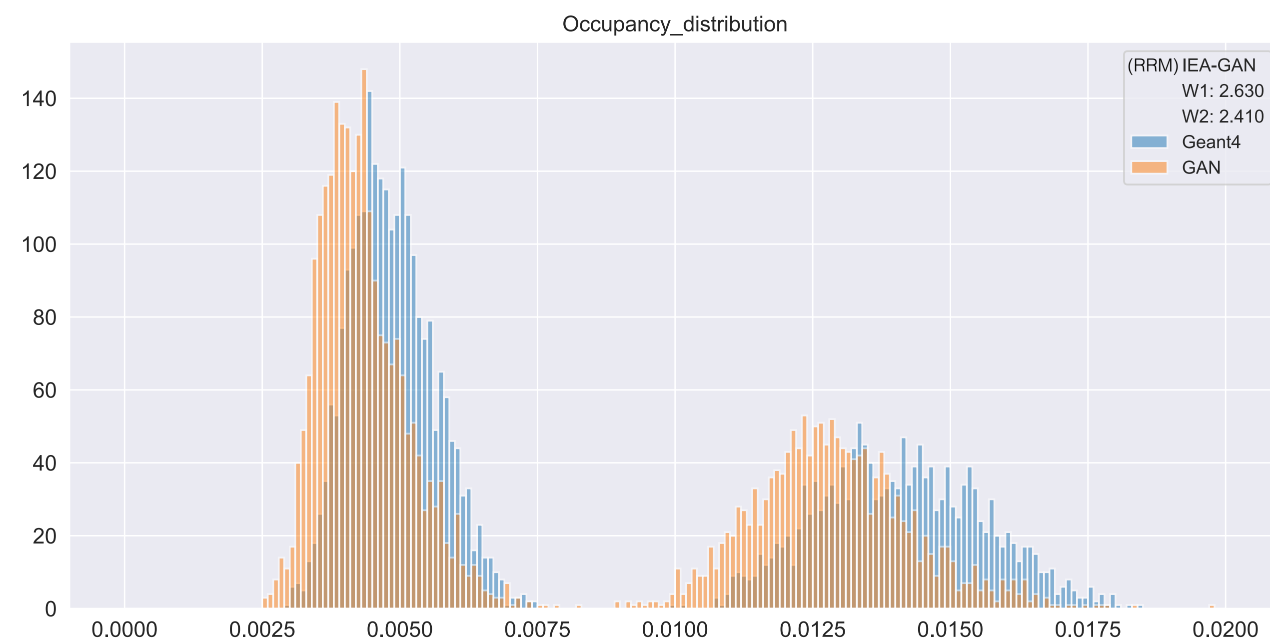
    - Possible interpretation of FID in pixel level:

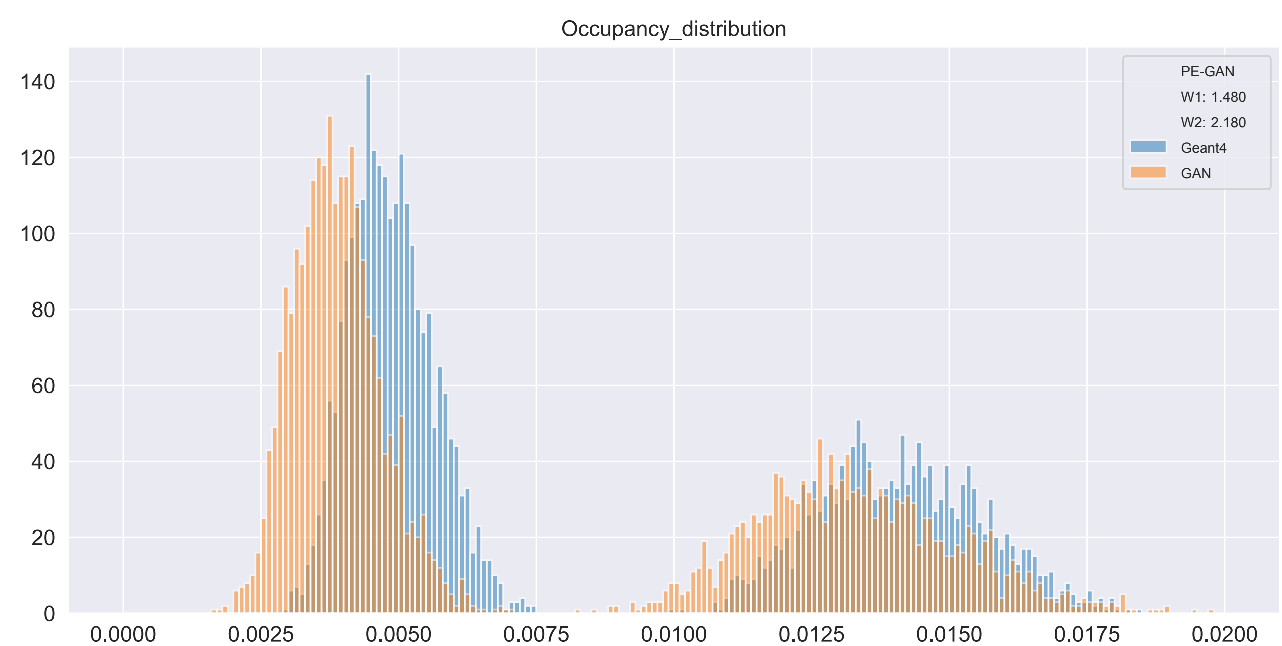


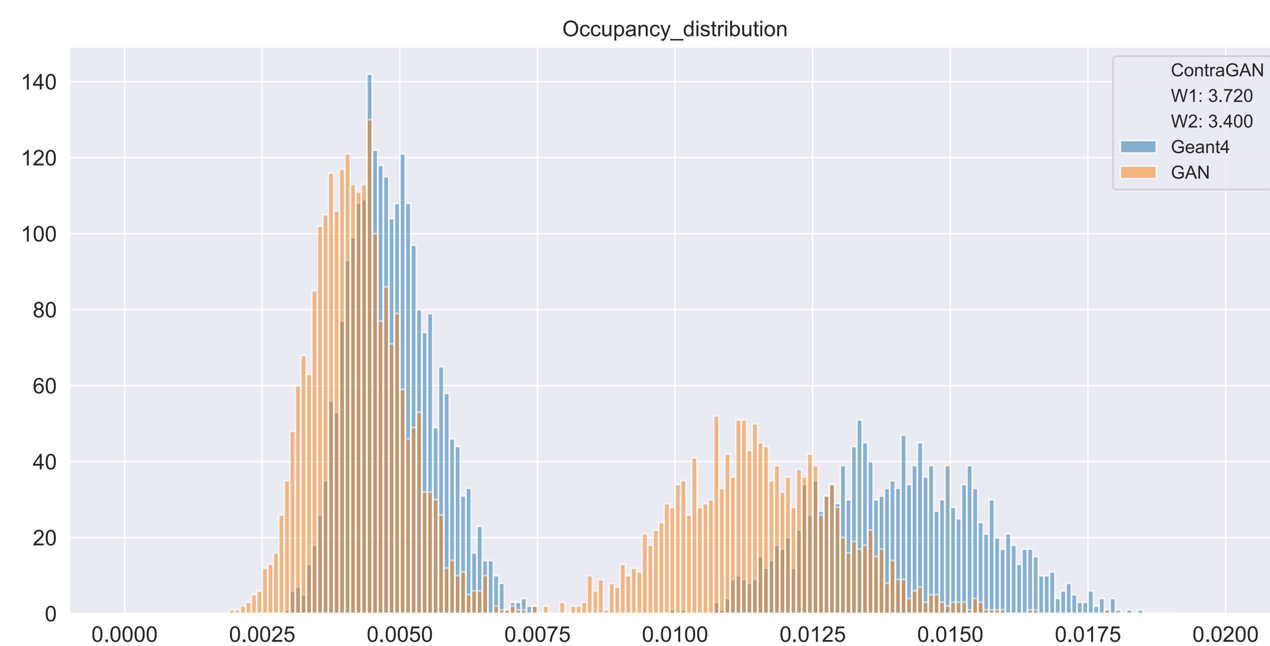
Validation of generated PXD images

- Validation Metrics over the test set:
  - ☑ Occupancy Distribution:



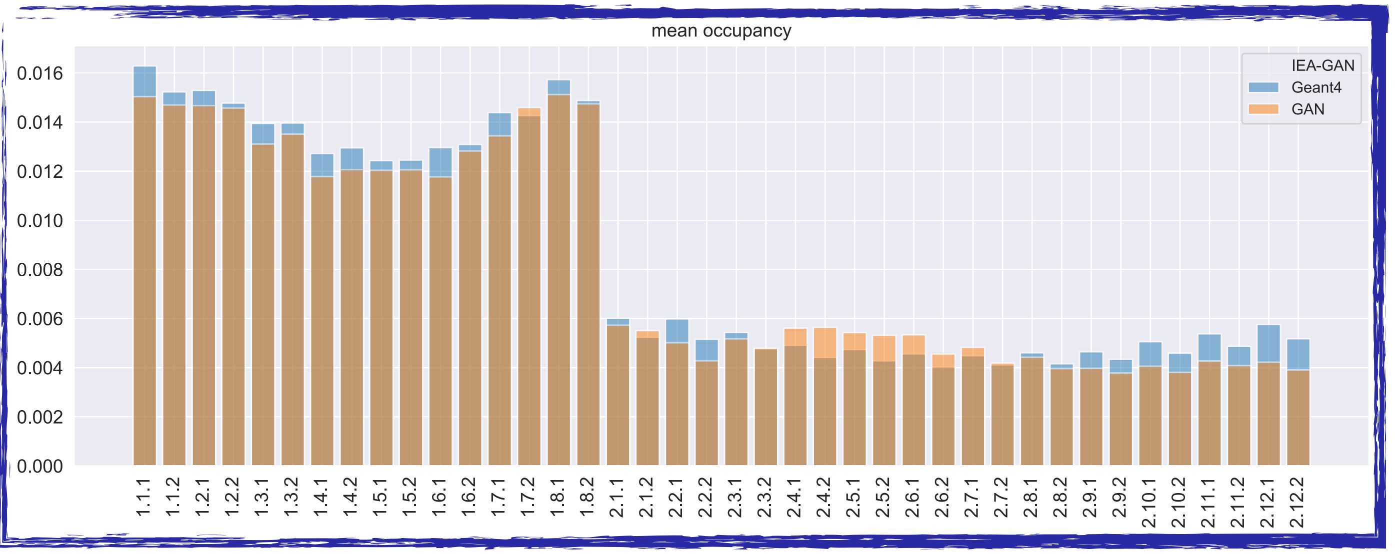


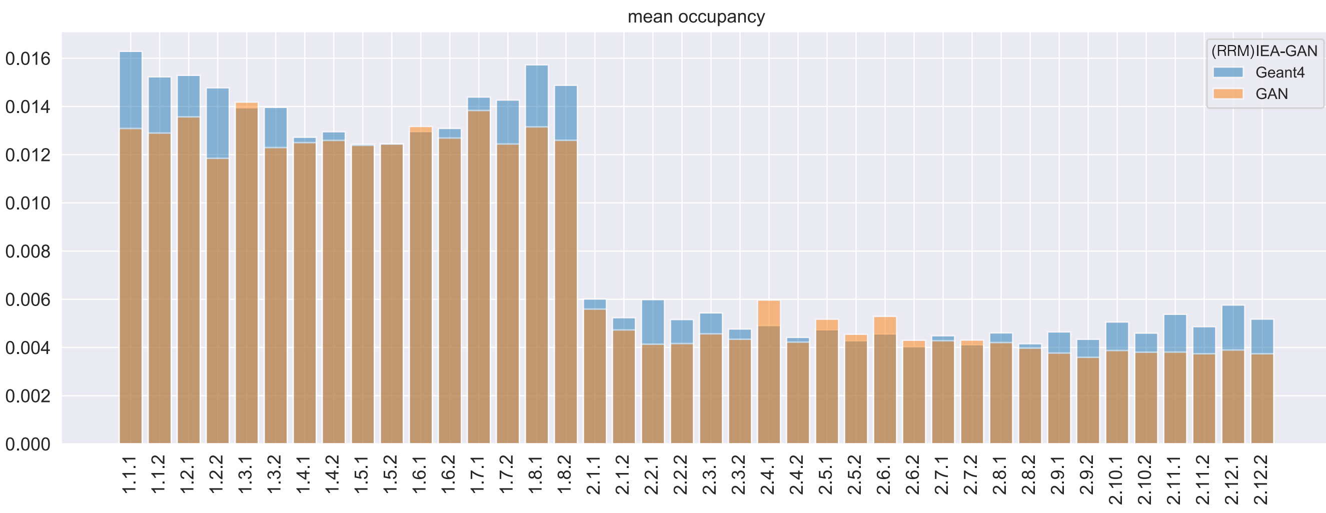


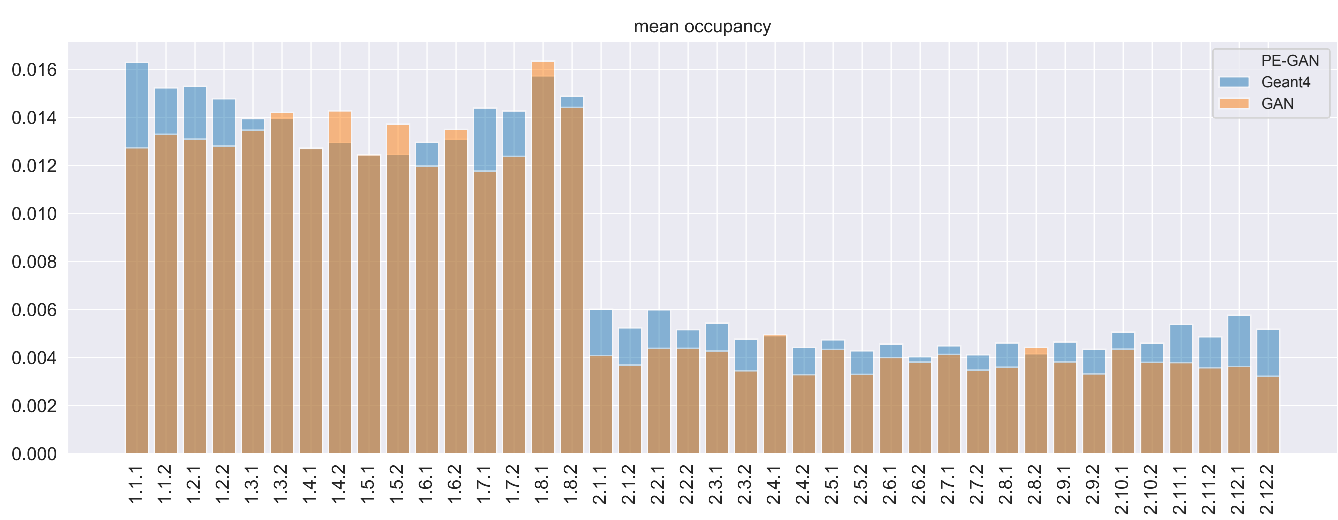


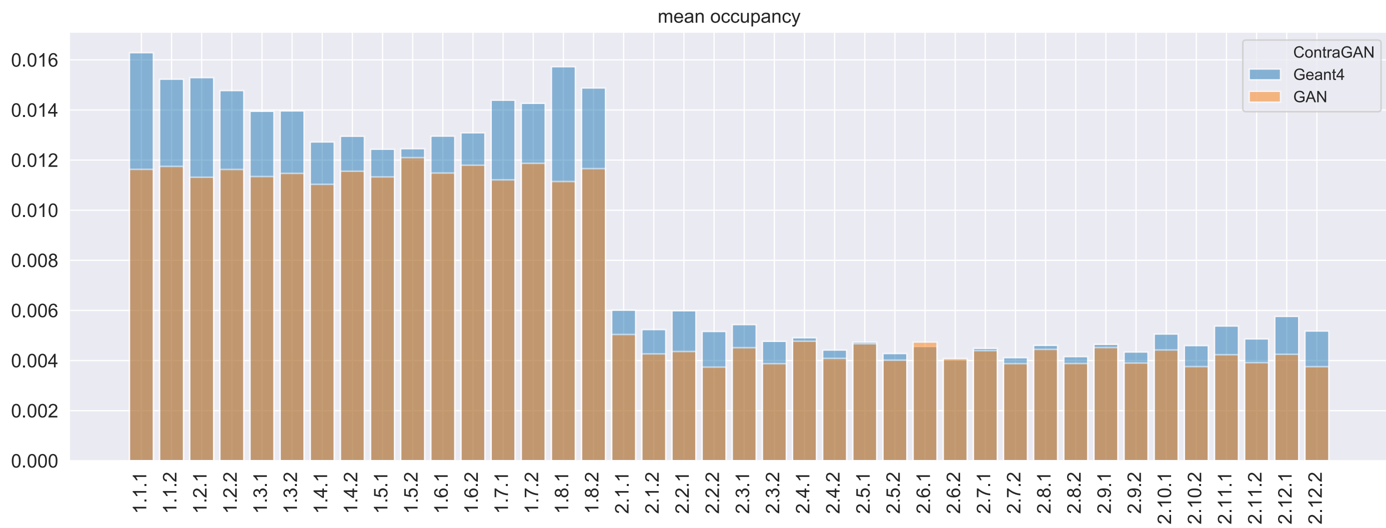
Validation of generated PXD images

- Validation Metrics over the test set:
  - Mean Occupancy:









Summary and Outlook

- **IEA-GAN:**
  - Successful generation of PXD images based on the sensor number in an end-to-end manner
  - Capturing class-to-class relations among the images in image generation by introducing Relational Reasoning Module
  - Transferring these inter-class connections to the generator via IEA-loss
  - Improving the training stability by using a Uniformity loss for the discriminator
- Working on the real detector data by transferring the same structure with minor modifications to generate them.
- Using the GAN as another way to understanding the Background processes at BelleII through PXD hitmaps.
- Doing a comprehensive validation of generated hitmaps by estimating the systematic uncertainty on the tracking efficiency, fake rate and resolution.
- Simulation Software implementation.
- Investigating future applications of IEA-GAN in fine-grained image generation task over natural images.

LUDWIG-
MAXIMILIANS
UNIVERSITY
MUNICH

Thank You

References

- Kang, Minguk, and Jaesik Park. "Contragan: Contrastive learning for conditional image generation." Advances in Neural Information Processing Systems 33 (2020): 21357-21369
- Hashemi, Hosein, et al. "Pixel Detector Background Generation using Generative Adversarial Networks at Belle II." EPJ Web of Conferences. Vol. 251. EDP Sciences, 2021.
- Srebre, Matej, et al. "Generation of Belle II Pixel Detector Background Data with a GAN." EPJ Web of Conferences. Vol. 245. EDP Sciences, 2020.
- Heusel, Martin, et al. "Gans trained by a two time-scale update rule converge to a local nash equilibrium." Advances in neural information processing systems 30 (2017).
- Brock, Andrew, Jeff Donahue, and Karen Simonyan. "Large scale GAN training for high fidelity natural image synthesis." arXiv preprint arXiv:1809.11096 (2018).

Back up Slides

The Base Model:

- **Technologies:**
  - Residual blocks
  - Spectral Normalisation
  - Orthogonal Weight init.
  - Orthogonal regularisation
  - Contrastive Learning
  - Hinge Loss
  - Consistency Regularisation
  - Differentiable Augmentation
  - IEA Loss
  - 5x10^-5 lr for both G and D

Algorithm 1 Intra-Event Aware GAN

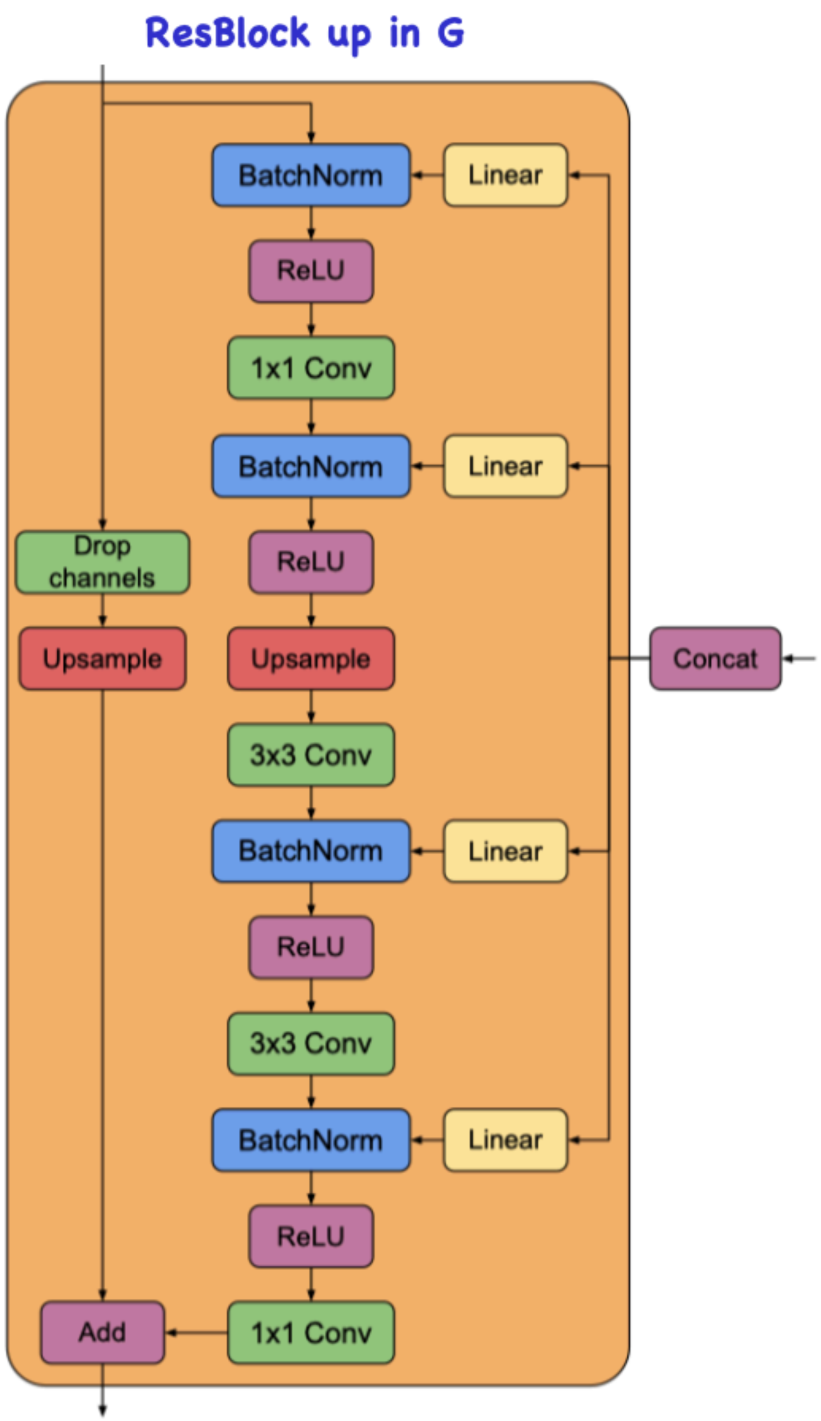
Require: generator and discriminator parameters θ_G, θ_D, Intra-Event-aware coefficient λ_{IEA}, Uniformity coefficient $\lambda_{uniform}$, Adam hyperparameters α, β_1, β_2, Event size M, number of discriminator iteration steps per generator iteration N_D

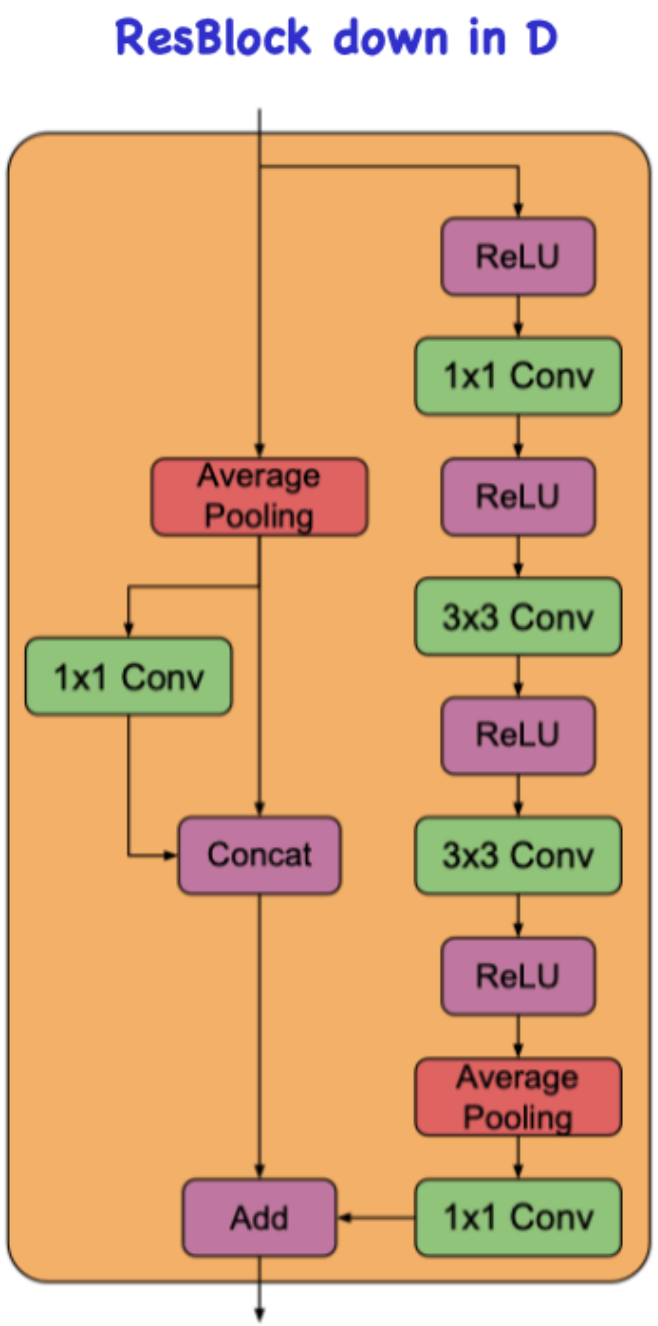
for number of training iterations **do**
    for $t = 1, ..., N_D$ **do**
        for $i = 1, ..., M$ **do**
            sample $z \sim p(z),\ x = p_{data}(x)$
            $L_D^{(i)} \leftarrow D[G(z)] - D(x)$
            $L_{uniform}^{(i)} \leftarrow \lambda_{uniform} L_{uniform}(h;\ t)$ ▷ The Uniformity Loss.
        end for
        $\theta_D \leftarrow Adam(\frac{1}{M}\sum_{i=1}^{M}(L_D^{(i)} + L_{uniform}^{(i)}), \alpha, \beta_1, \beta_2)$
    end for
    sample $\{z^{(i)}\}_{i=1}^{M} \sim p(z)$
    $x_{fake} = G(z)$
    $L_G^{(i)} \leftarrow -D[G(z)]$
    $L_{IEA}^{(i)} \leftarrow \lambda_{IEA} D_{KL}(h[G(z)], h(x))$ ▷ The Intra-Event KL Loss.
    $\theta_G \leftarrow Adam(\frac{1}{M}\sum_{i=1}^{M}(L_G^{(i)} + L_{IEA}^{(i)}), \alpha, \beta_1, \beta_2)$
end for

Back up Slides

The Base Model:

- **Technologies:**
  - Residual blocks
  - Spectral Normalisation
  - Orthogonal Weight init.
  - Orthogonal regularisation
  - Contrastive Learning
  - Hinge Loss
  - Consistency Regularisation
  - Differentiable Augmentation
  - IEA Loss
  - 5x10^-5 lr for both G and D





Back up Slides

- The fired pixels are only read out if their value exceeds a threshold, 7 ADU.
- How to capture this prior information about the image?
- Solution: To add this information to the training in a differentiable way???

Algorithm 1 Pixel-Aware regularization.

Input: generator and discriminator parameters θ_G, θ_D, pixel-aware regularization coefcient λ, Adam hyperparameters α, β_1 , β_2, batch size M, number of discriminator iterations per generator iteration N_D

for number of training iterations **do**
    for $t = 1, ..., N_D$ **do**
        for $i = 1, ..., M$ **do**
            sample $z \sim p(z)$, $x = p_{data}(x)$
            $L_D^{(i)} \leftarrow D[G(z)] - D(x)$
        end for
        $\theta_D \leftarrow Adam(\frac{1}{M} \sum_{i=1}^{M} (L_D^{(i)}), \alpha, \beta_1, \beta_2)$
    end for
    sample $\{z^{(i)}\}_{i=1}^{M} \sim p(z)$
    $x_{fake} = G(z)$
    $F[G(z)] : x_{fake} \mapsto x_{fake}^{cutoff}$ ▷ Threshold wrt. the pixel constraints.
    $L_{pr}^{(i)} \leftarrow ||G(z) - F[G(z)]||^2$
    $L_G^{(i)} \leftarrow -D[G(z)]$
    $\theta_G \leftarrow Adam(\frac{1}{M} \sum_{i=1}^{M} (L_G^{(i)} + L_{pr}), \alpha, \beta_1, \beta_2)$
end for