# Conditional Poisson Flow Generative model: a class conditional image generator

Ioannis Georgiades [1], Lydia Chen[1] and Zilong Zhao [1]

[1]TU Delft

## Research Question

How to introduce the concept of conditional sampling images using the Poisson Flow Generative Model (PFGM) [1][2]?

1. How can we introduce the content of the label of every image to the model/underlying neural network?
2. How should the existing loss function needs to be modified to account for conditional image generation
3. What changes need to be made in terms of the overall code structure to support training, such conditional model?

## Background and Related works

This research is based on the paper introducing the PFGM [2]. This study introduces a novel Poisson flow generative model (PFGM), utilizing a high-dimensional electric field analogy to map a uniform distribution into any data distribution. Offering accelerated performance in image generation tasks and robustness to various estimation errors, PFGM shows state-of-the-art results on CIFAR-10 and competes effectively with cutting-edge stochastic differential equation (SDE) techniques.

## Methodology

Our proposed model is an extension of the existing PFGM by introducing two key ideas.

1. Augment the color channels of every image by including the label information in the form of one-hot like encoding color channels. See figure 1 for example.

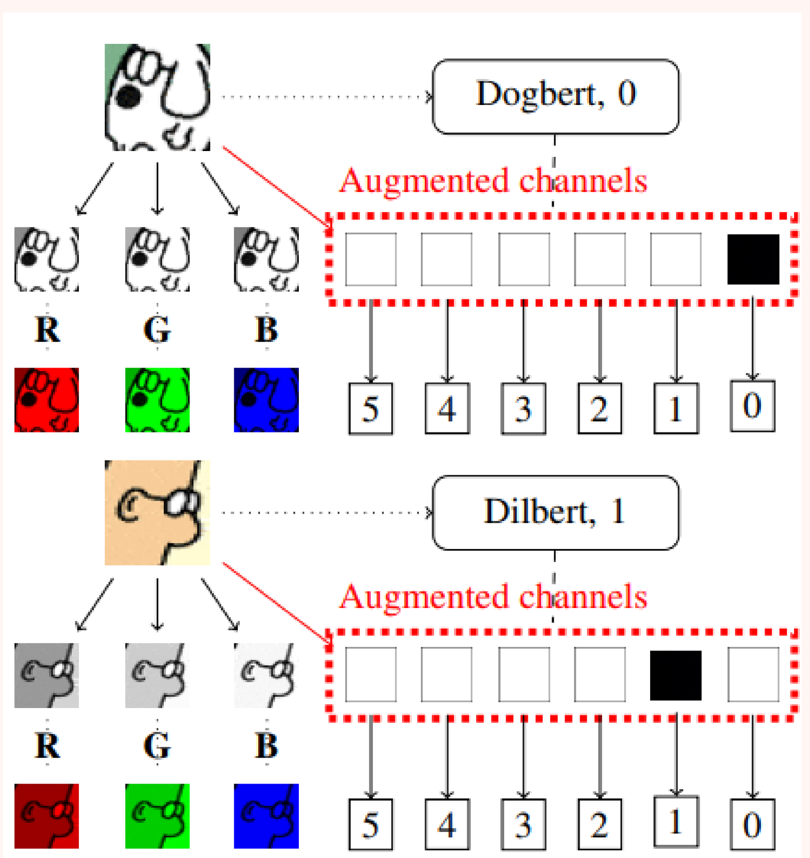


Figure 1. Visualization of the augmented channels for images of Dogbert and Dilbert. All channels are grids, for the RGB channels, we are showing both the grayscale interpretation and the actual color. For the augmented channels that correspond to either only white or black pixel grids.

## Methodology

2. Update the existing Loss Function to incorporate the Cross-Entropy Loss of the images generated during training. Cross-Entropy Loss is based on the predicted class probabilities for each image and their true class. Such loss aims not only to predict the correct class but also for the image to have a high probability of belonging to that class.

But, to achieve such a thing, we need to update the overall architecture of our training model as it is shown in figure 2. Where in red are the parts we have added, and in **bold** are the updated components of the previous model.

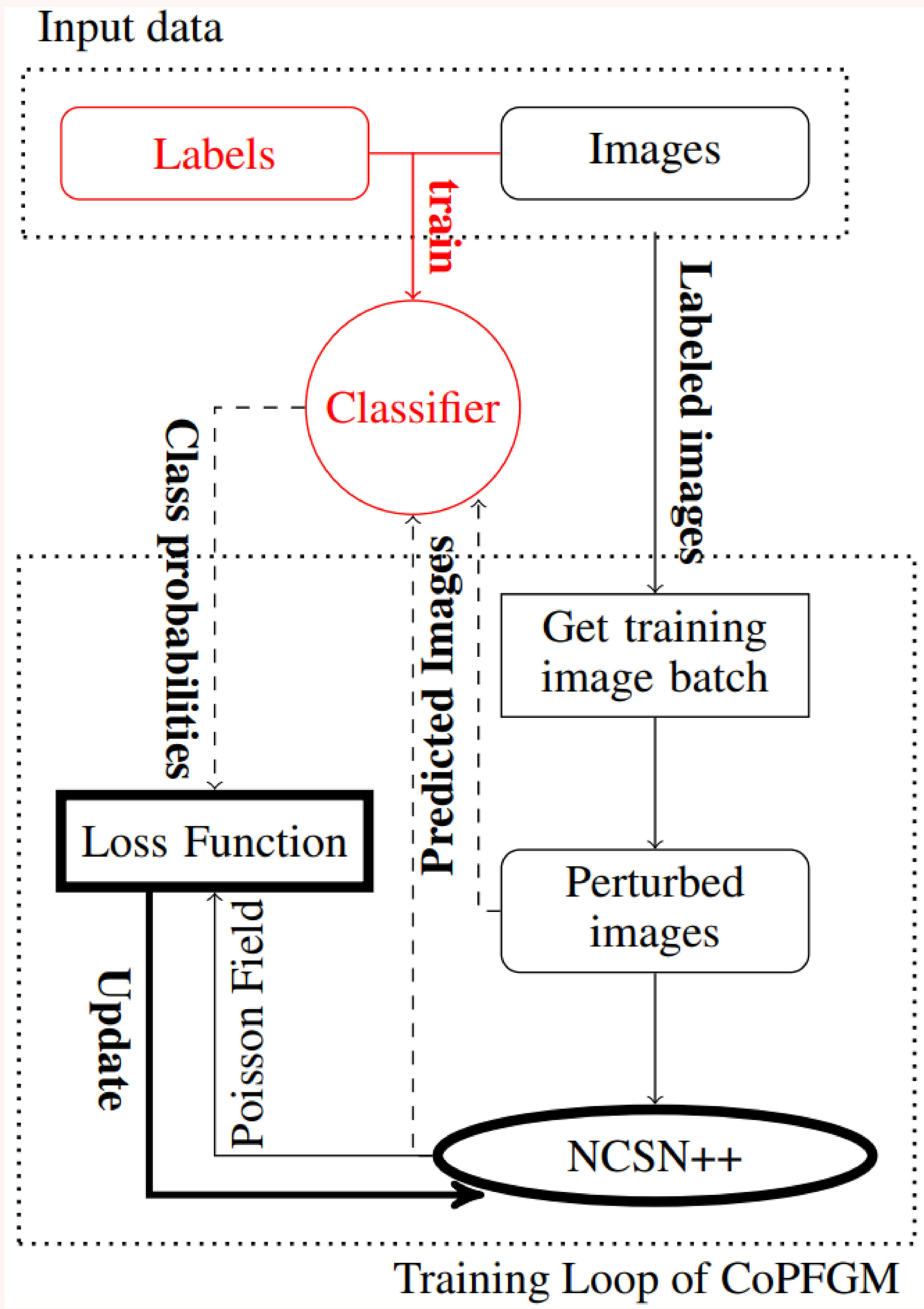


Figure 2. Updated code structure regarding training CoPFGM, which includes the classifier and updated loss function.

## Results

Here, in the figure 3, we show a combined representation of the model, generating all 6 classes from the Dilbert-faces dataset, as well as for the 10 classes of the MNIST dataset.

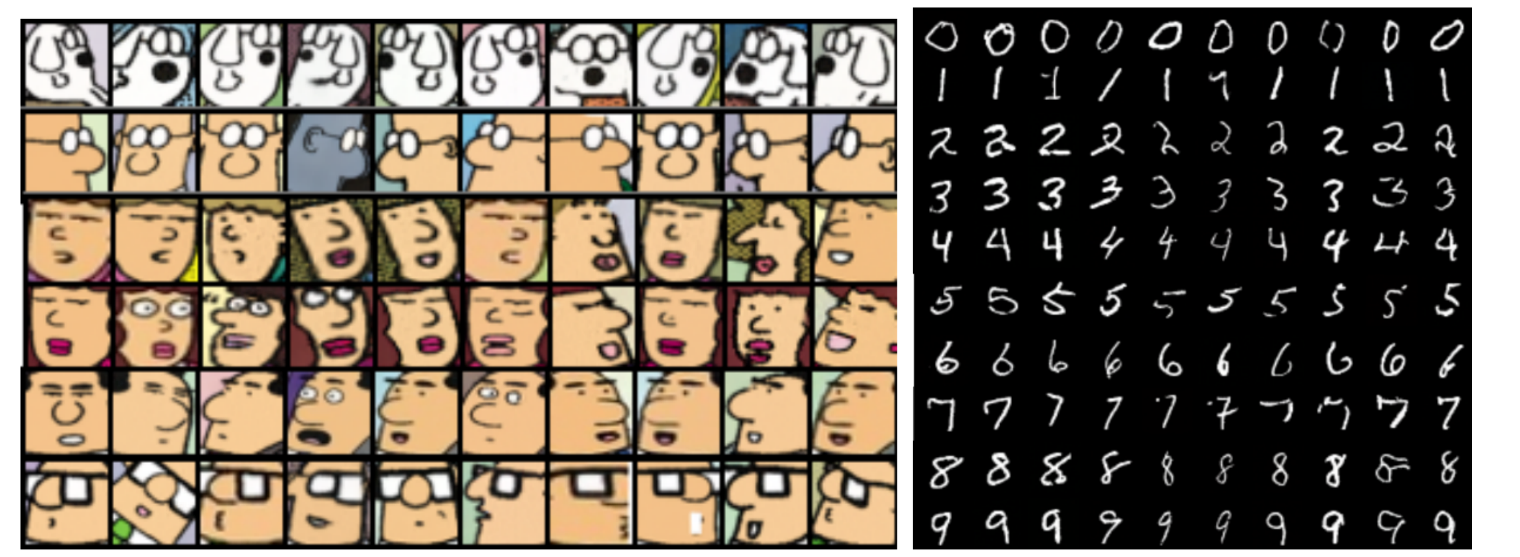

Figure 3. Results from the 2 datasets that were tested with the CoPFGM.

| Model | Dataset | FID score * | Inception score* |
|---|---|---|---|
| CoPFGM | Dilbert | $1.4930 \times 10^2$ | 2.5904 |
| CoPFGM-Without classifier | Dilbert | $1.4608 \times 10^2$ | 2.9499 |
| CoPFGM-Without channel augmentation | Dilbert | $1.0077 \times 10^2$ | 2.8267 |
| CoPFGM | MNIST | $3.1364 \times 10^2$ | 1.7850 |
| CoPFGM-Without classifier | MNIST | $3.0943 \times 10^2$ | 1.8149 |
| CoPFGM-Without channel augmentation | MNIST | $2.1500 \times 10^2$ | 2.8150 |

The CoPFGM was also tested in the form of an ablation study against other versions of itself but each without one of the introduced ideas. Results, about image quality can be seen in the above table.

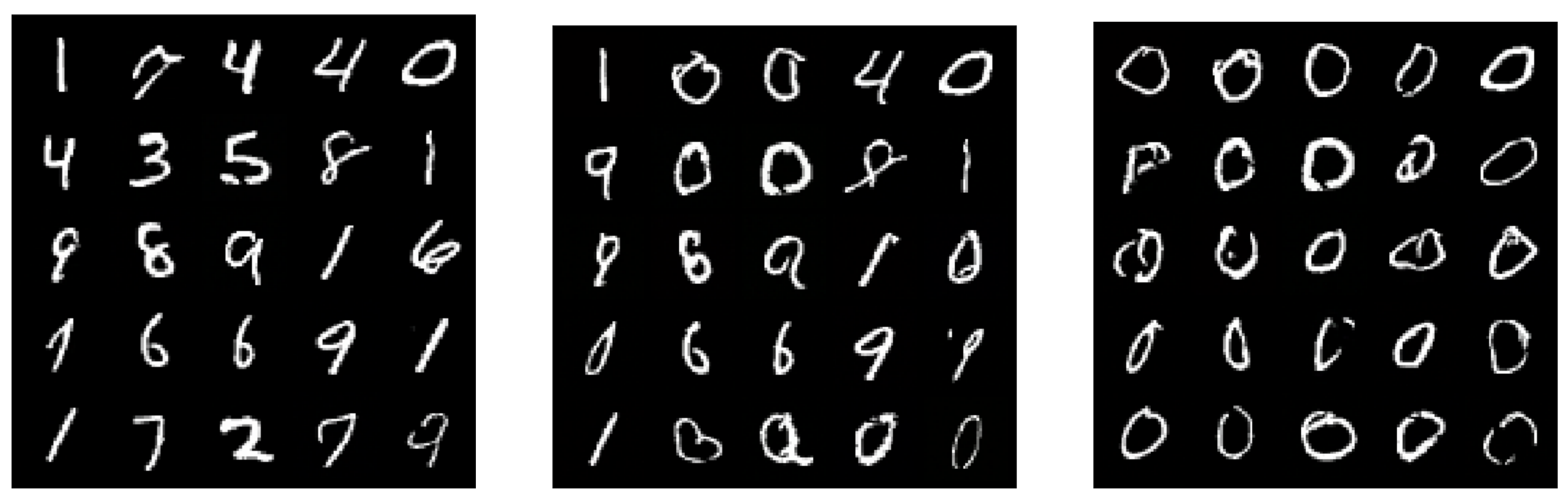

Figure 4. Samples From left to right, we have (1) CoPFGM without Channel Augmentation, (2) CoPFGM without Cross Entropy Loss, and (3) the proposed CoPFGM.

## References

[1] Newbeeer. Poisson flow, 2022.

[2] Yilun Xu, Ziming Liu, Max Tegmark, and Tommi Jaakkola. Poisson flow generative models, 2022.

igeorgiades@student.tudelft.nl