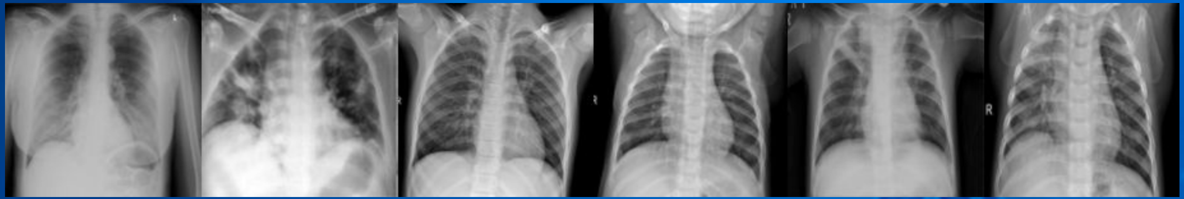
Politecnico di Torino

Uncertainty-based Loss for Chest X-rays data generation



Giammarinaro Silvia
Manunza Gilberto

Repository

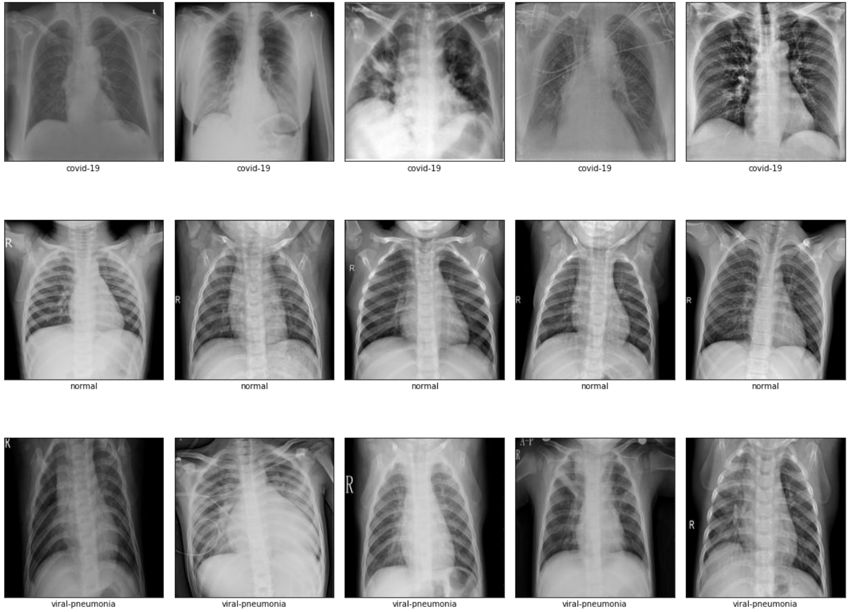
Bioinformatics
AY 2020/2021

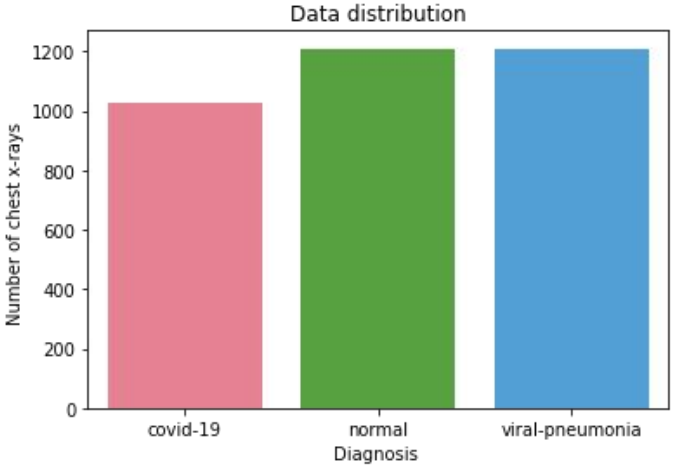
Roadmap

1. Synthetic images generation using GAN
2. Exploration of various conditional and unconditional GAN methods
3. Improve generated images quality using uncertainty
4. Using the Chest X-rays dataset for classification benchmark
5. Evaluation of our generated samples for a classification task

The dataset

COVID-19 Radiography Database.





1. Generative Adversarial Networks

Generate synthetic images of COVID-19 Chest X-rays

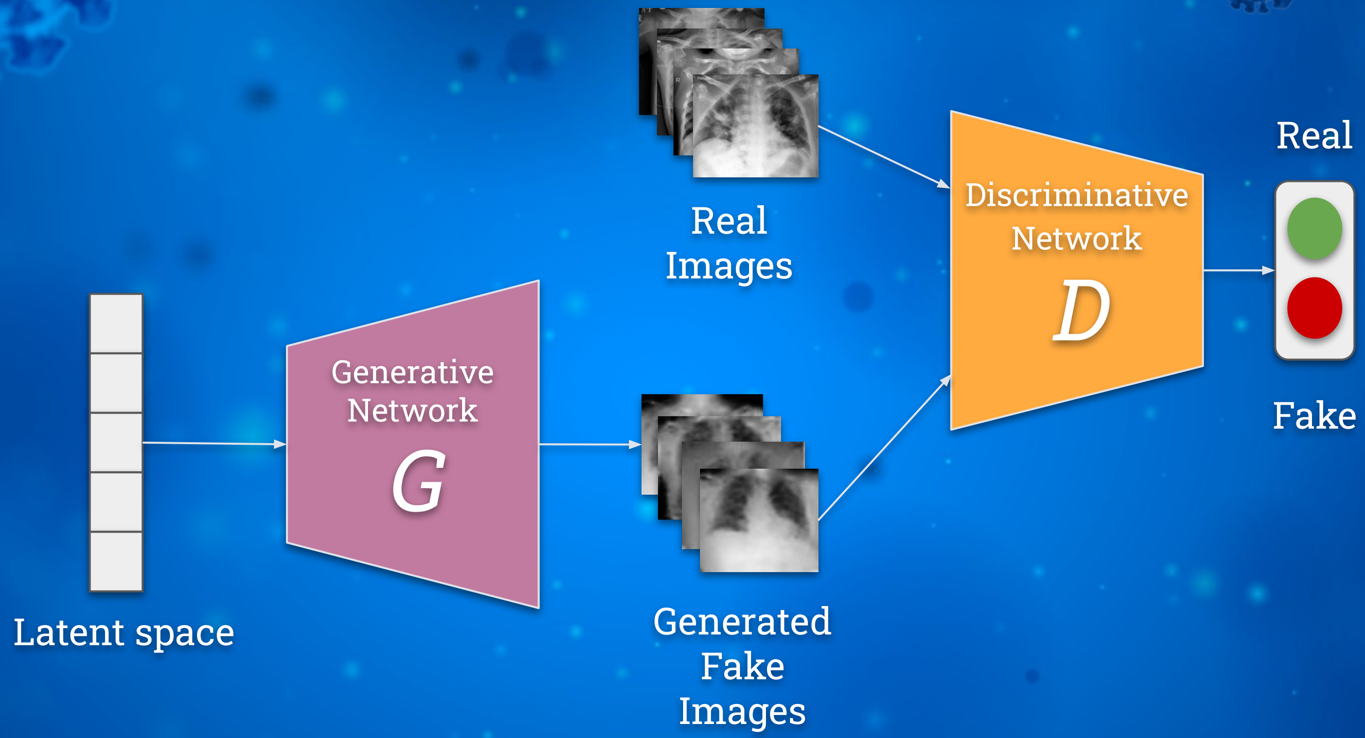
Evaluation metric used

Frechet Inception Distance

$$d_{FID}(x,g) = \|\mu_x - \mu_g\|^2 + Tr\left[\Sigma_x + \Sigma_g - 2(\Sigma_x \Sigma_g)^{\frac{1}{2}}\right]$$

where μ_x and μ_g are the feature-wise mean of real and generated images respectively, Σ_x and Σ_g are the covariance matrix of real and generated images repectively, Tr is the trace whichis the sum of the elements along the main diagonal of the square matrix, $X_x \sim \mathcal{N}(\mu_x, \Sigma_x)$ and $X_g \sim \mathcal{N}(\mu_g, \Sigma_g)$ are the 2048-dimensional activation's of the Inception-V3 pool3 layer for real images and generated images respectively.

covidGAN architecture



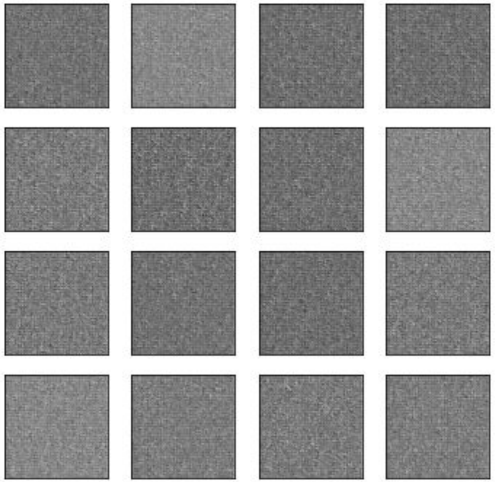
covidGAN: definition and results

$$\mathcal{L}_D = -E_x(log(D(x)) - E_z(log(1 - D(G(z))))$$
$$\mathcal{L}_G = -E_z(log(D(G(z))))$$

where:

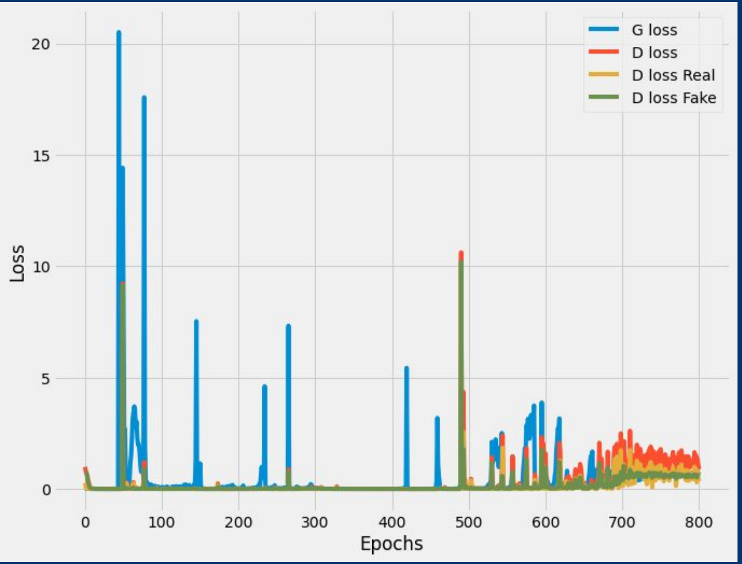
- $D(x)$ is the discriminator's estimate of the probability that a real data instance x is real.
- E_x is the expected value over all real data instances.
- $G(z)$ is the generator's output when given noise z.
- $D(G(z))$ is the discriminator's estimate of the probability that a fake instance is real.
- E_z is the expected value over all random inputs to the generator.

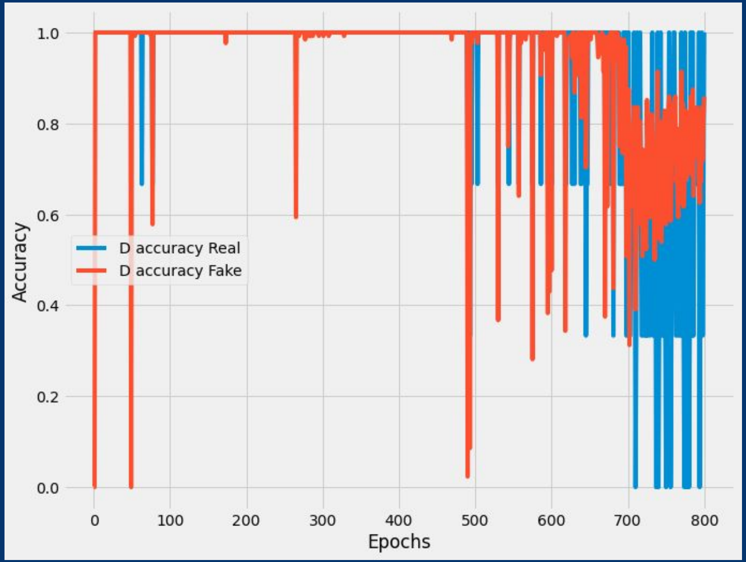
FID: 313.84 ± 2.48



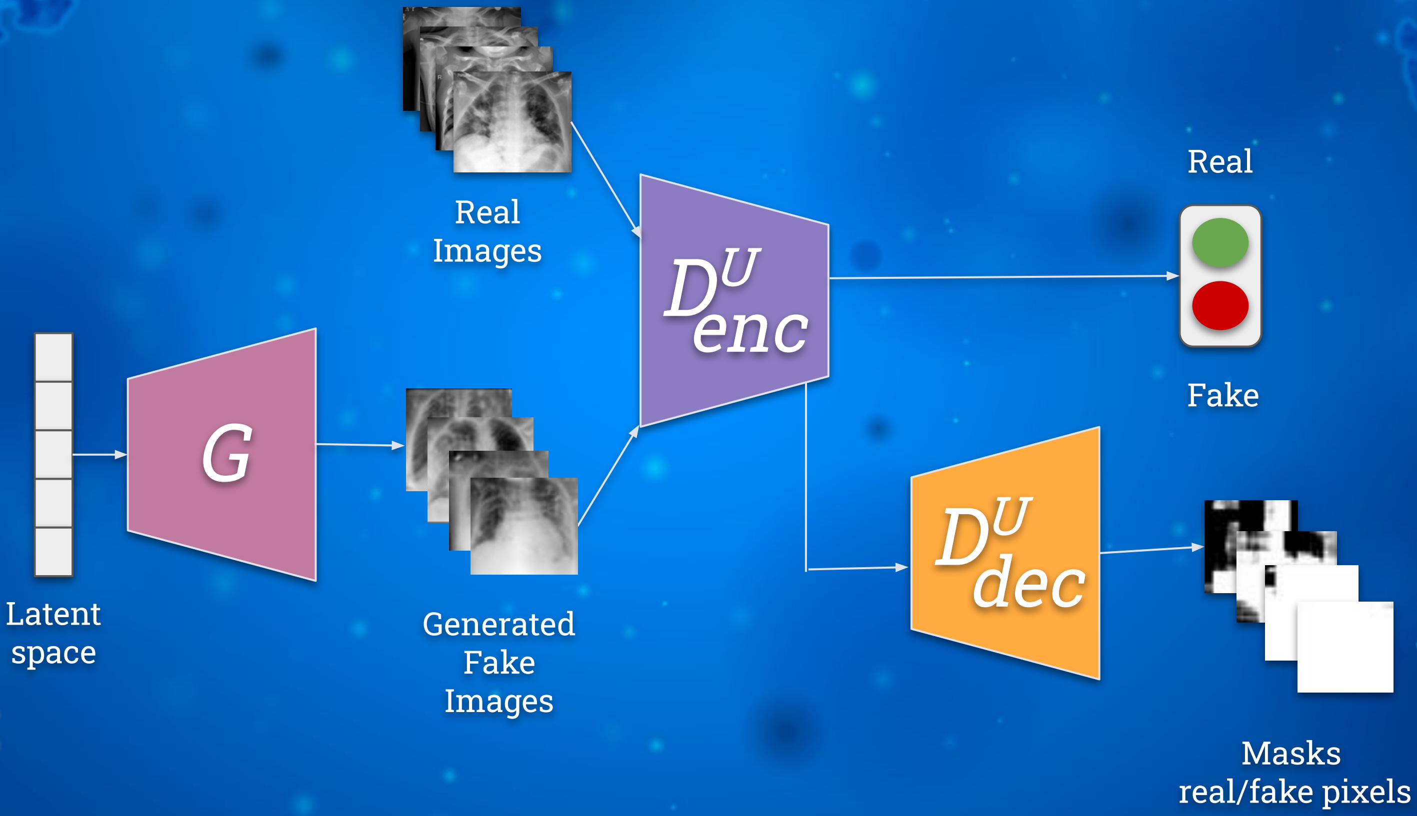
Generated images from covidGAN

covidGAN: loss and accuracy





covidUnetGAN architecture



covidUnetGAN: definition and results

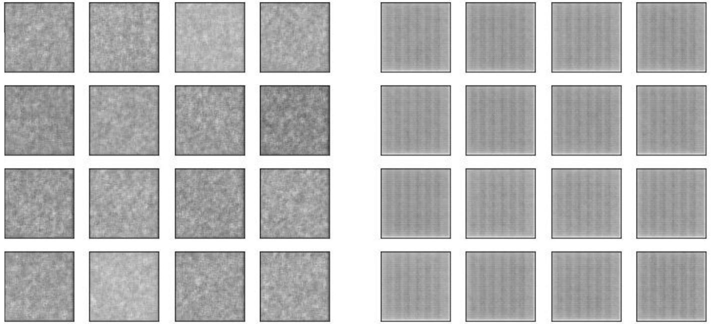
$$\mathcal{L}_{D^U_{enc}} = -E_x(log(D^U_{enc}(x)) - E_z(log(1 - D^U_{enc}(G(z))))$$

$$\mathcal{L}_{D^U_{dec}} = -E_x(\sum_{(i,j)} log[D^U_{dec}(x)]_{(i,j)}) - E_z(\sum_{(i,j)} log[1 - D^U_{dec}(G(z))]_{(i,j)})$$

$$\mathcal{L}_D = \mathcal{L}_{D^U_{enc}} + \mathcal{L}_{D^U_{dec}}$$

$$\mathcal{L}_G = -E_z(log(D^U_{enc}(G(z))) + \sum_{(i,j)} log[D^U_{dec}(G(z))]_{(i,j)})$$

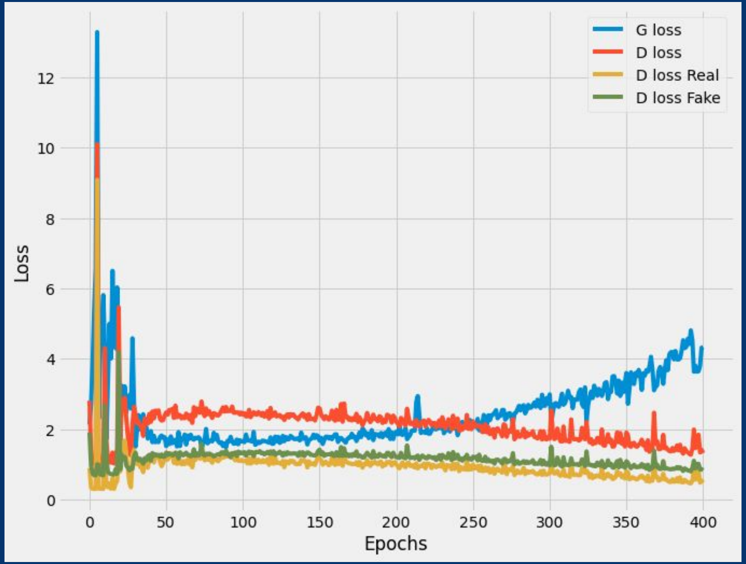
$[D^U_{dec}(x)]_{(i,j)}$ and $[D^U_{dec}(G(z))]_{(i,j)}$ refer to the discriminator decision at pixel (i, j).

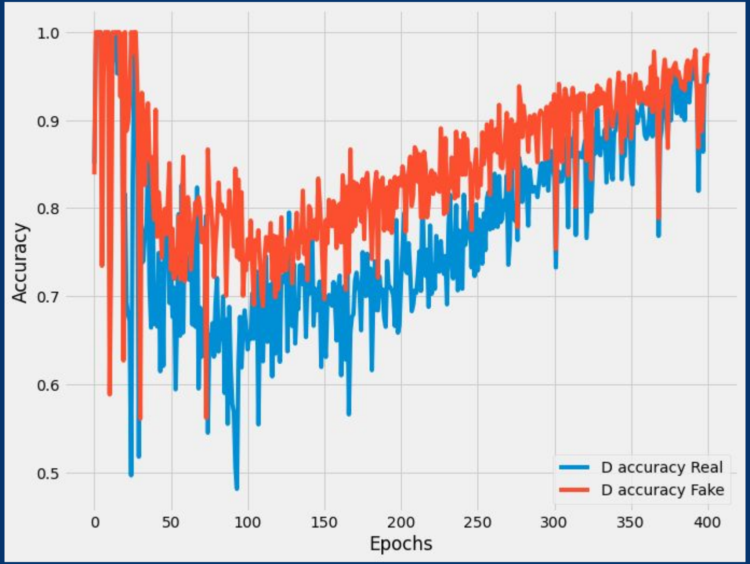


Generated images and masks from covidUnetGAN

FID: 188.48 ± 3.84

covidUnetGAN: loss and accuracy

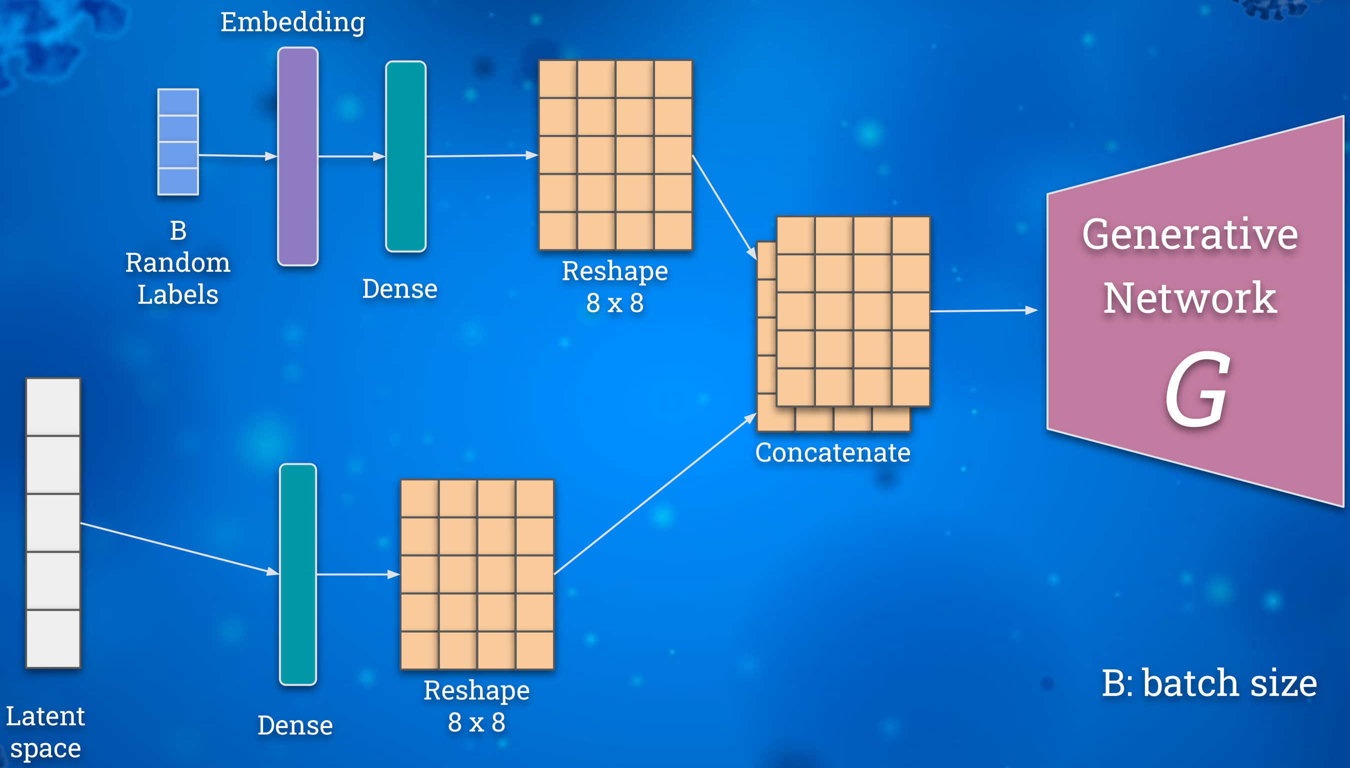




2. Conditional GANs

Generate synthetic images of Chest X-rays with different pathologies

Generator Input for cGAN



B: batch size

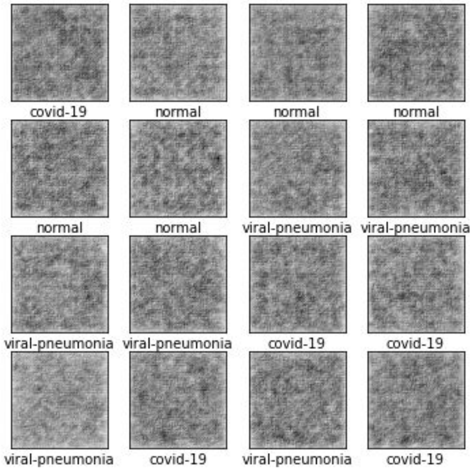
The same input concatenation is used for the discriminator, where real images are given

cGAN: definition and results

$$\mathcal{L}_D = -E_x(log(D(x|y)) - E_z(log(1 - D(G(z|y))))$$
$$\mathcal{L}_G = -E_z(log(D(G(z|y))))$$

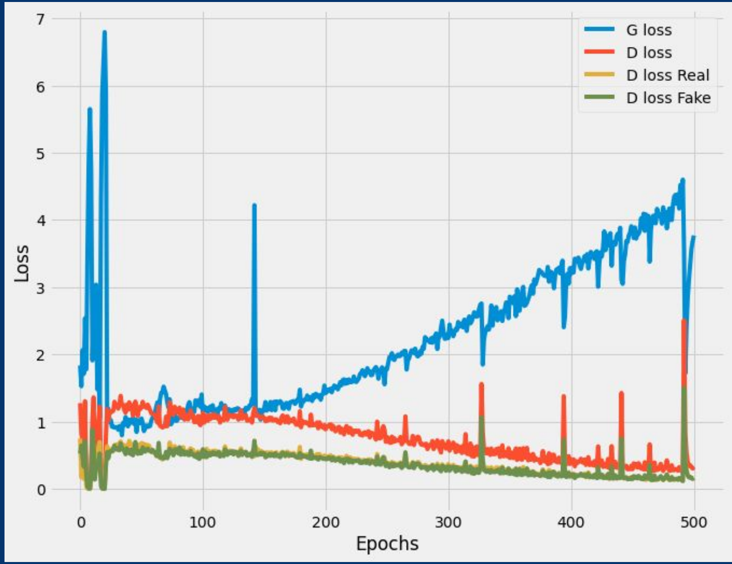
where y is the class label.

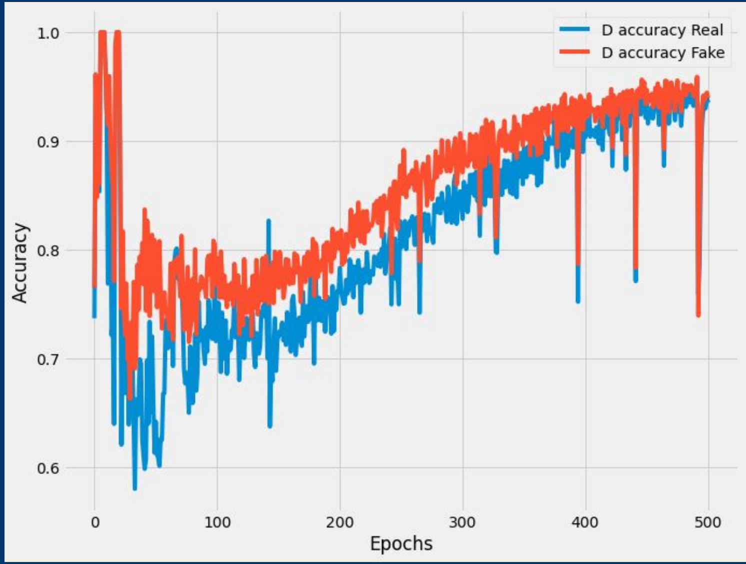
FID: 80.65 ± 1.27



Generated images from cGAN

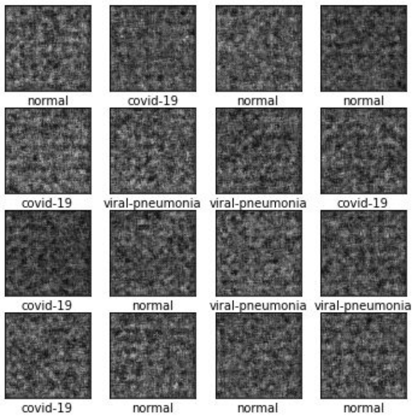
cGAN: loss and accuracy

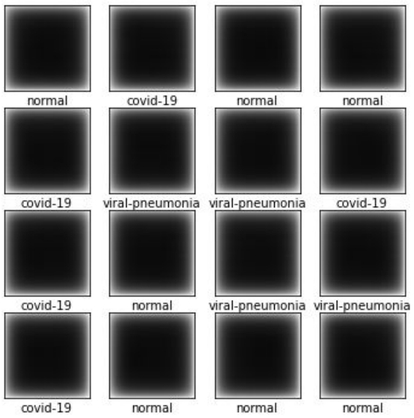




UnetCGAN: definition and results

The class label is inserted into the covidUnetGAN loss as done for the cGAN case.

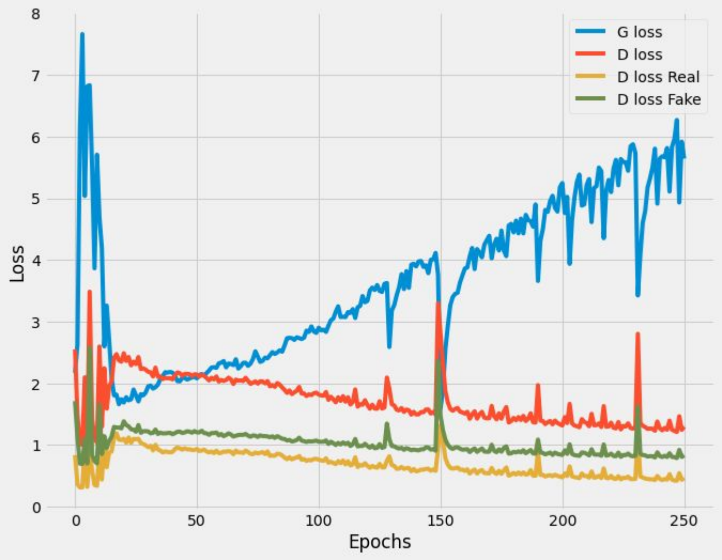


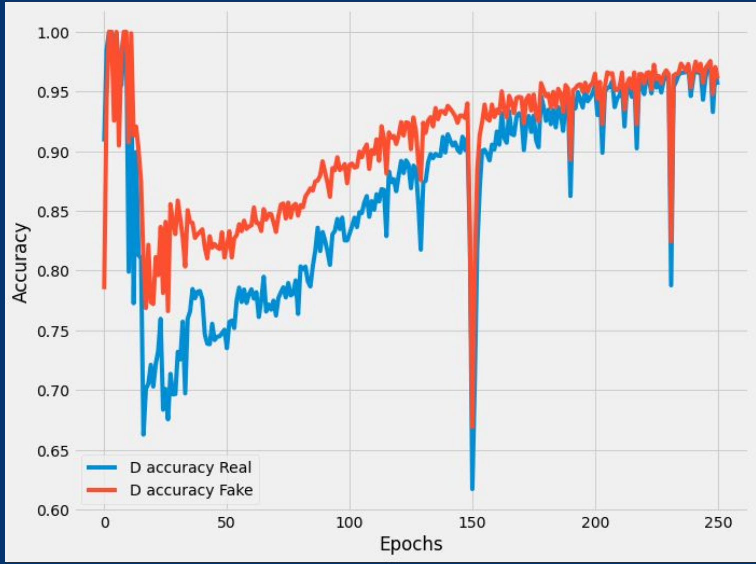


FID: 89.76 ± 2.03

Generated images and masks from UnetCGAN

UnetCGAN: loss and accuracy

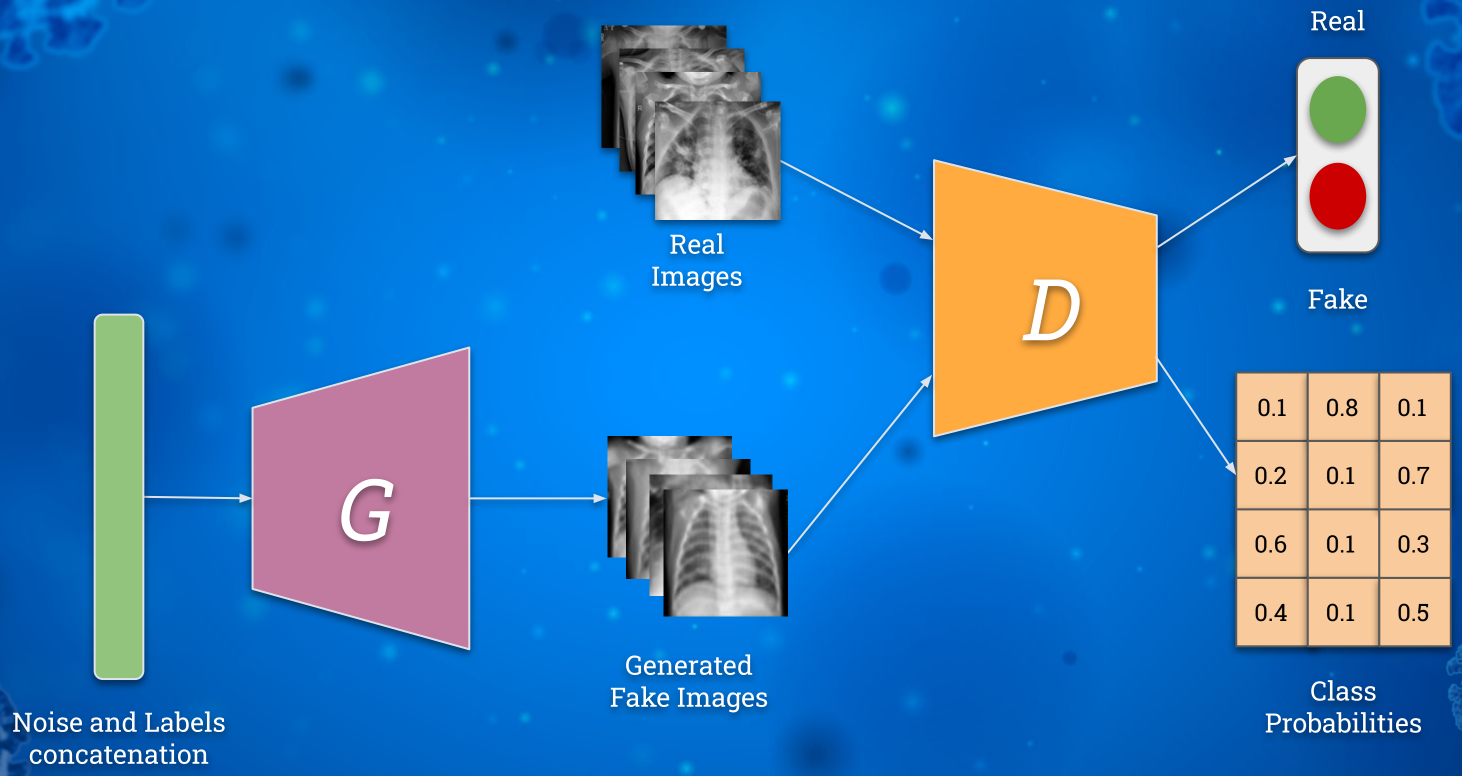




3. Auxiliary Classifier Conditional GAN

An extension of conditional GANs

AC-CGAN architecture



AC-CGAN: definition and results

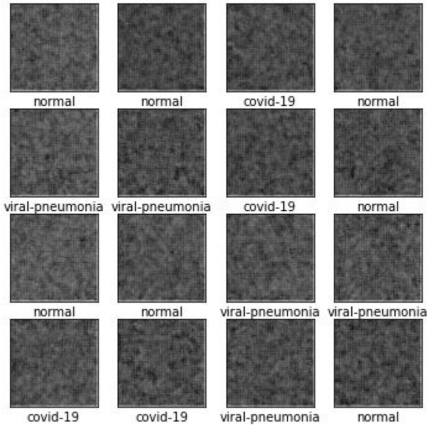
$$\mathcal{L}_{disc} = -E_x(log(D_{disc}(x)) - E_z(log(1 - D_{disc}(G(z|y))))$$

$$\mathcal{L}_{class}(x) = -E_x(\sum_{c \in \mathcal{C}} y_c log(D_{class}(x)_c)$$

$$\mathcal{L}_D = \mathcal{L}_{disc} + \mathcal{L}_{class}(x) + \mathcal{L}_{class}(G(z))$$

$$\mathcal{L}_G = -E_z(log(D(G(z)))) + \mathcal{L}_{class}(G(z))$$

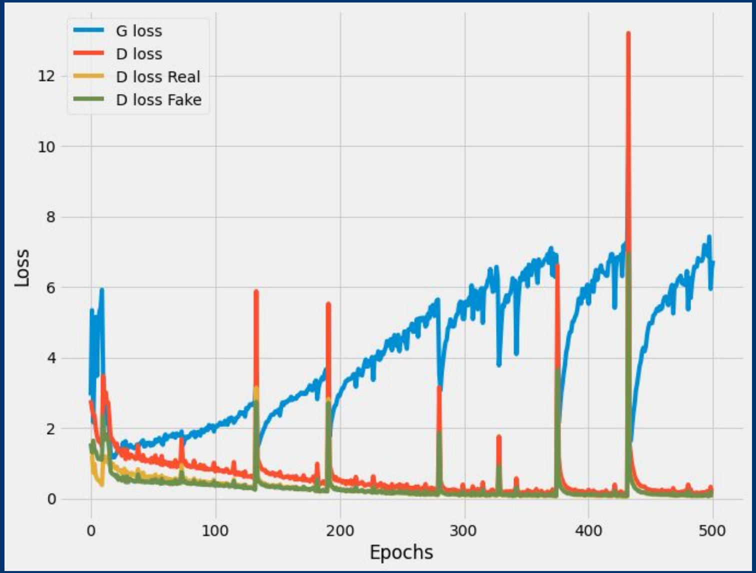
where D_{disc} is the binary discriminator output (like in the standard GAN), D_{class} is the classification discriminator softmax output and $\mathcal{C}$ is the the set of classes.

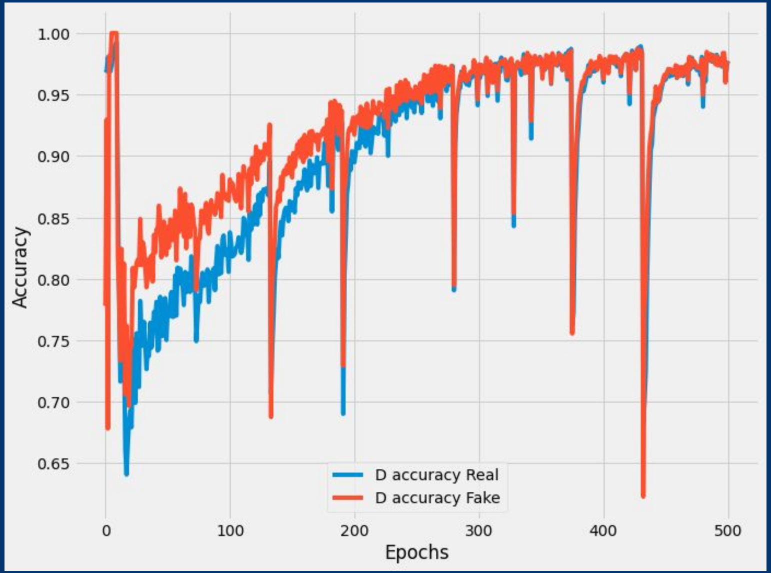


FID: 81.87 ± 1.85

Generated images from ACCGAN

AC-CGAN: loss and accuracy





4. Uncertainty regularization

Our proposed method

Uncertainty in Deep Learning

In the Monte Carlo Dropout setting, we compute for T times the predictions of the net. In this case the uncertainty is the following:

$$\underbrace{\frac{1}{T}\sum_{t=1}^{T} diag(\hat{p}_t) - \hat{p}_t^{\otimes 2}}_{aleatoric} + \underbrace{\frac{1}{T}\sum_{t=1}^{T}(\hat{p}_t - \overline{p})^{\otimes 2}}_{epistemic}$$

where $\bar{p} = \sum_{t=1}^{T} \hat{p}_t / T$ and $\hat{p}_t = p(\hat{\omega}_t) = \text{Softmax}\{f^{\hat{\omega}_t}(x^*)\}$.

cGAN uncertainty

$$D_{mean} = \frac{1}{T}\sum_{i=1}^{T} D_i(x|y)$$

$$\hat{p} = D(x|y),\ \bar{p} = \sum_{i=1}^{T} \hat{p}_i/T$$

$$U = \frac{1}{T}\sum_{i=1}^{T}(\hat{p}_i(1-\hat{p}_i)) + \frac{1}{T}\sum_{i=1}^{T}(\hat{p}_i-\bar{p})^2$$

$$\mathcal{L}_D = -E_x(log(D_{mean}(x|y))) - E_z(log(1-D_{mean}(G(z|y)))) + \alpha(U_{real}+U_{fake})$$

$$\mathcal{L}_G = -E_z(log(D_{mean}(G(z|y)))) - \alpha U_{fake}$$

where D_{mean} is the mean of the prediction using the discriminator model with Monte Carlo Dropout with T samples. U is the new uncertainty term added to the model and α is an hyper-parameter to weight the uncertainty contribution. In this formulation, we consider the max mode. For the min mode we have:

$$\mathcal{L}_D = -E_x(log(D_{mean}(x|y)) - E_z(log(1-D_{mean}(G(z|y)))) - \alpha(U_{real}+U_{fake})$$

$$\mathcal{L}_G = -E_z(log(D_{mean}(G(z|y)))) + \alpha(U_{fake})$$

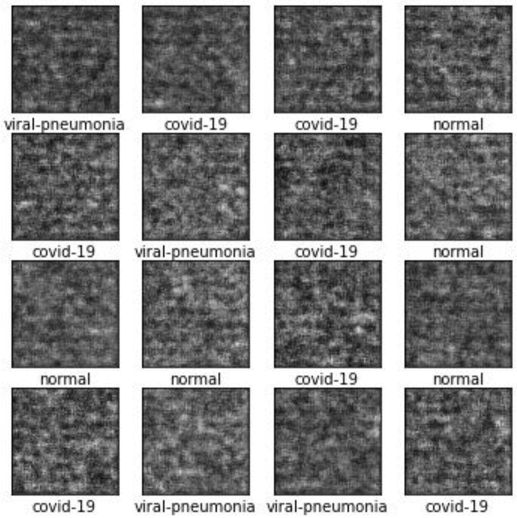
The different uncertainty modes

- **Min uncertainty case:**
  The generator (G) wants to fool the discriminator (D) and minimize its doubts. On the other hand D wants to maximize its own “critic spirit” and judge taking into account multiple possibilities.

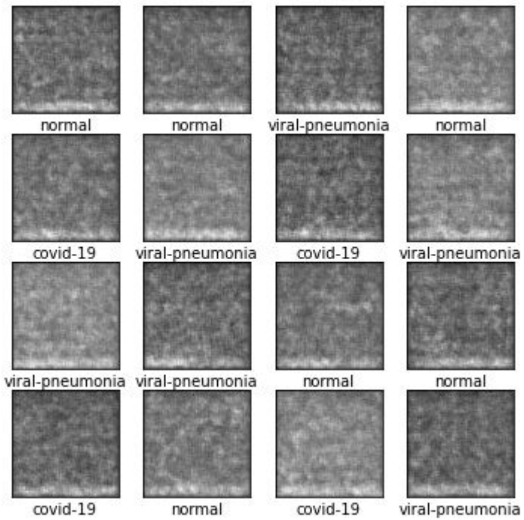
- **Max uncertainty case:**
  G wants to make D uncertain about its evaluations so that G can learn better from its mistakes. It is like G is asking to D a more complete explanation on where G should improve. On the other hand D wants to minimize its uncertainty and be more sure about its predictions.

cGAN uncertainty: results

*Note: cGAN FID: 80.65 ± 1.27



Generated images from CGAN uncertainty min
FID: 72.68 ± 0.92

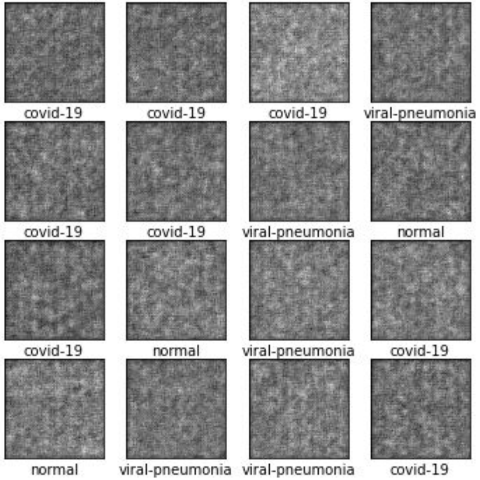


Generated images from CGAN uncertainty max
FID: 62.59 ± 0.61

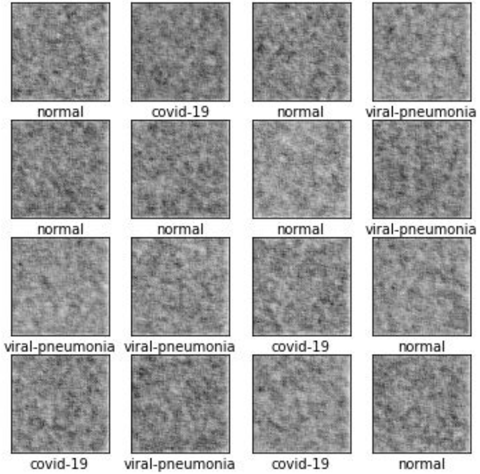
AC-cGAN uncertainty

The uncertainty loss has been applied to the AC-cGAN model as well in the same way as in the cGAN case

Note: AC-cGAN FID: 81.87 ± 1.85



Generated images from AC-cGAN uncertainty min
FID: 89.65 ± 1.45

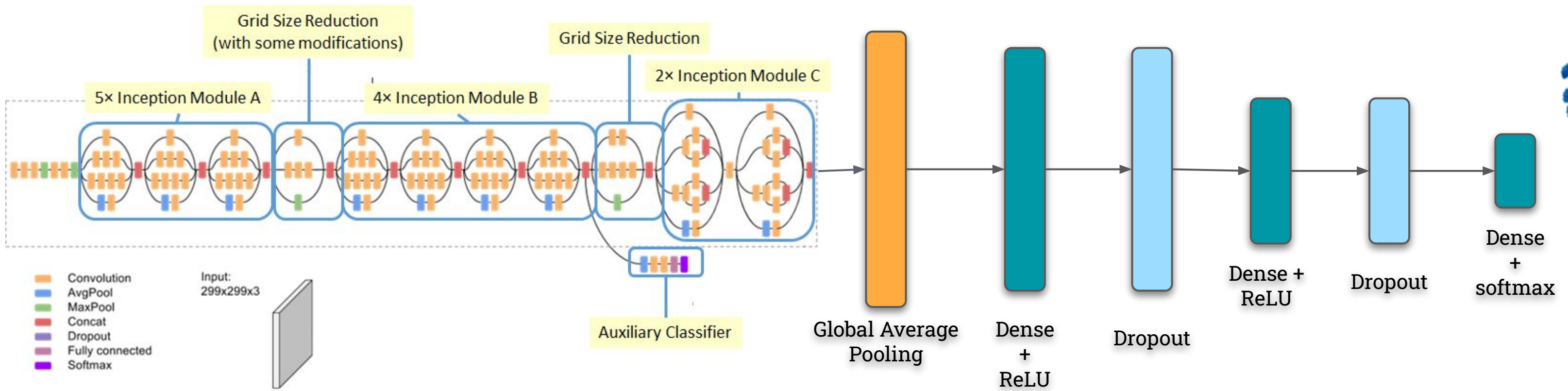


Generated images from AC-cGAN uncertainty max
FID: 78.54 ± 1.59

5. Classification task

Comparing a deterministic and a Monte Carlo Dropout method

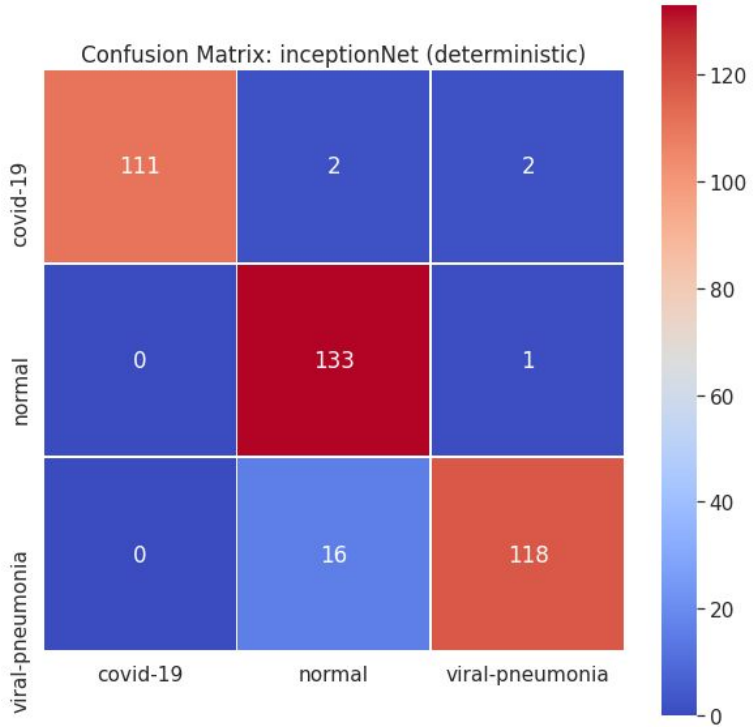
Modified Inception V3 net



Feature extractor

Classifier

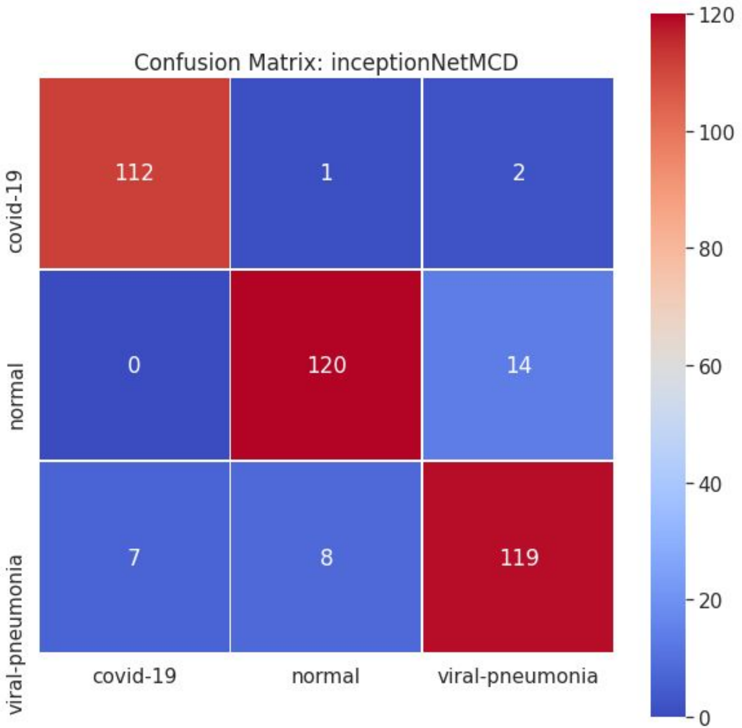
inceptionNet results



Accuracy	Recall Covid	Recall Normal	Recall VP	Precision Covid	Precision Normal	Precision VP
0.944 ± 0.026	0.939 ± 0.066	0.966 ± 0.025	0.928 ± 0.046	1.000 ± 0.000	0.917 ± 0.060	0.936 ± 0.020

Results are averaged over 5 runs and reported as mean ± std

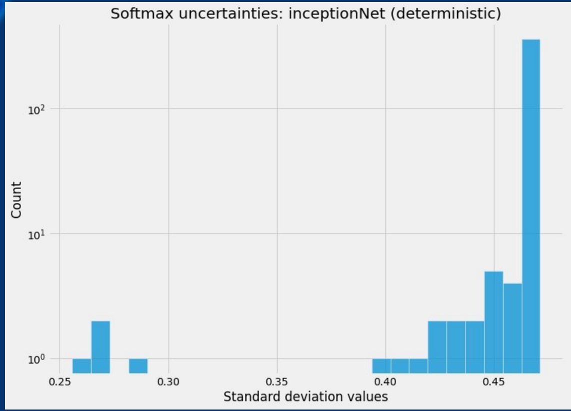
InceptionNetMCD results

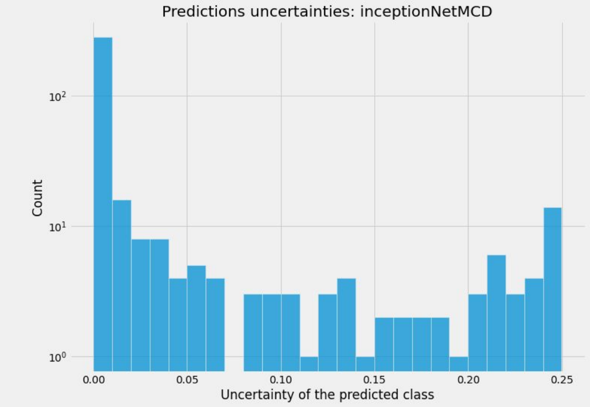


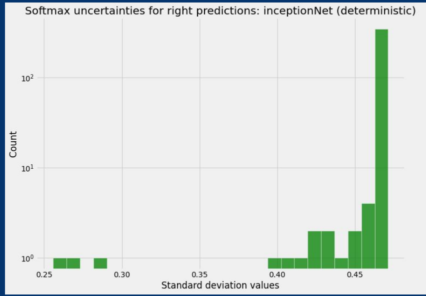
Accuracy	Recall Covid	Recall Normal	Recall VP	Precision Covid	Precision Normal	Precision VP
0.907 ± 0.007	0.974 ± 0.007	0.894 ± 0.009	0.861 ± 0.010	0.933 ± 0.009	0.922 ± 0.011	0.878 ± 0.010

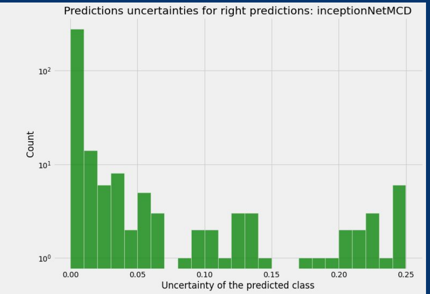
Results are are obtained by sampling five times and reported as mean ± std

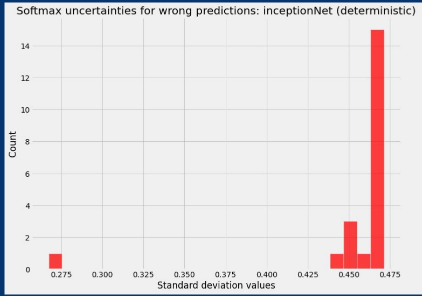
InceptionNet VS InceptionNetMCD

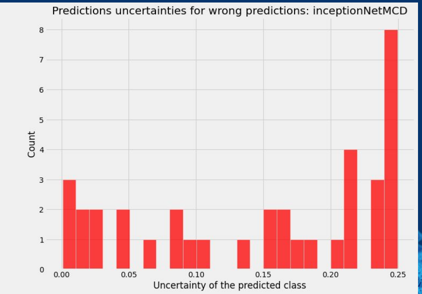












6. Generative Classification

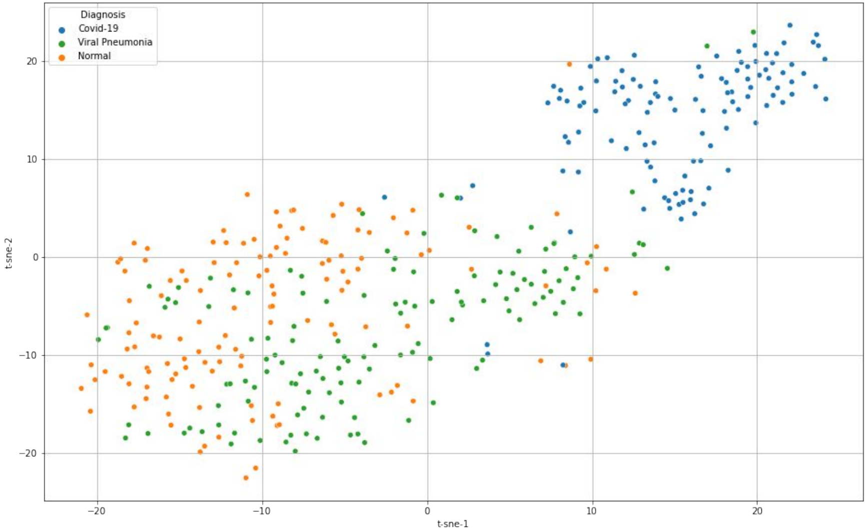
Using the GAN images for classification

Half data training

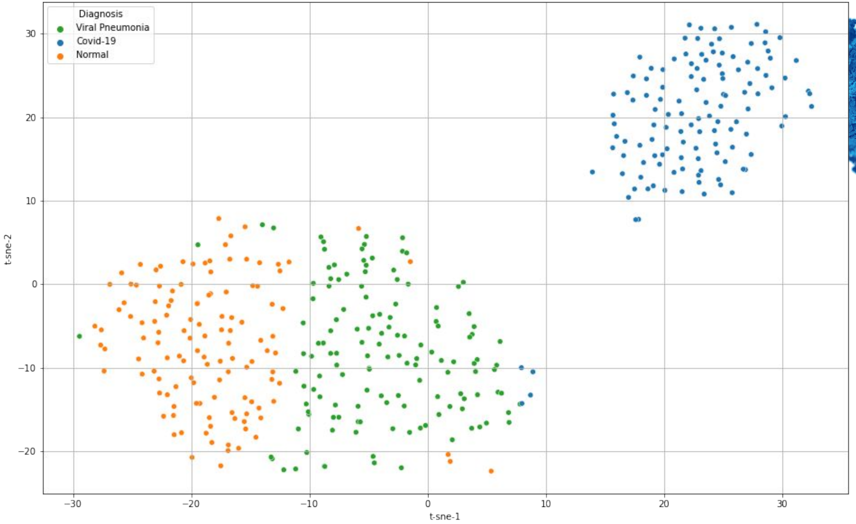
We used the deterministic inceptionNet. Half batch comes from the training set, half batch from the generated images. The baseline is the model trained only on real data.

Model	Accuracy	Recall Covid	Recall Normal	Recall VP	Precision Covid	Precision Normal	Precision VP
Baseline	0.944 ± 0.026	0.939 ± 0.066	**0.966 ± 0.025**	0.928 ± 0.046	**1.000 ± 0.000**	0.917 ± 0.060	0.936 ± 0.020
cGAN	0.959 ± 0.007	0.974 ± 0.012	0.930 ± 0.019	0.973 ± 0.017	**1.000 ± 0.000**	**0.972 ± 0.016**	0.917 ± 0.023
cGAN + uncertainty (min)	**0.968 ± 0.008**	**0.981 ± 0.007**	0.960 ± 0.023	0.966 ± 0.010	0.997 ± 0.007	0.966 ± 0.009	**0.948 ± 0.019**
cGAN + uncertainty (max)	0.965 ± 0.004	0.979 ± 0.009	0.951 ± 0.013	0.967 ± 0.012	**1.000 ± 0.000**	0.965 ± 0.014	0.937 ± 0.007
AC-cGAN	0.948 ± 0.016	0.979 ± 0.009	0.894 ± 0.039	**0.975 ± 0.007**	0.995 ± 0.004	0.971 ± 0.014	0.897 ± 0.037
AC-cGAN + uncertainty (min)	0.950 ± 0.008	0.977 ± 0.017	0.921 ± 0.017	0.954 ± 0.027	0.995 ± 0.004	0.955 ± 0.028	0.912 ± 0.011
AC-cGAN + uncertainty (max)	0.956 ± 0.011	0.977 ± 0.009	0.922 ± 0.024	0.970 ± 0.016	**1.000 ± 0.000**	0.961 ± 0.023	0.921 ± 0.029

t-SNE visualization

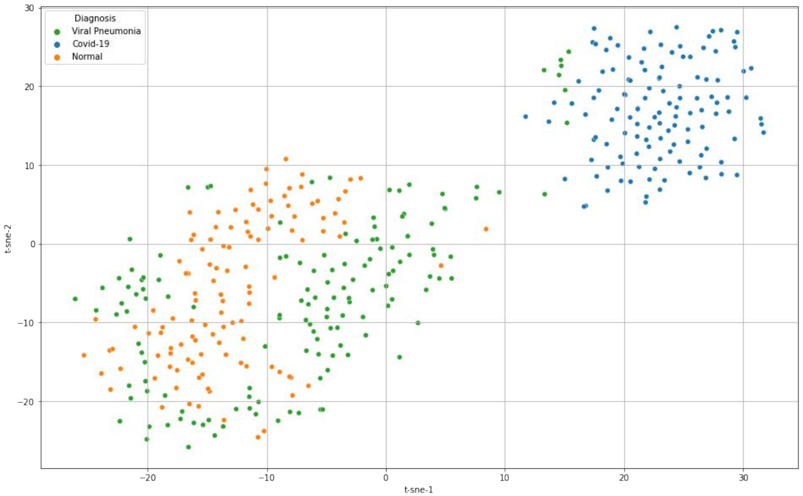


Real Images t-SNE

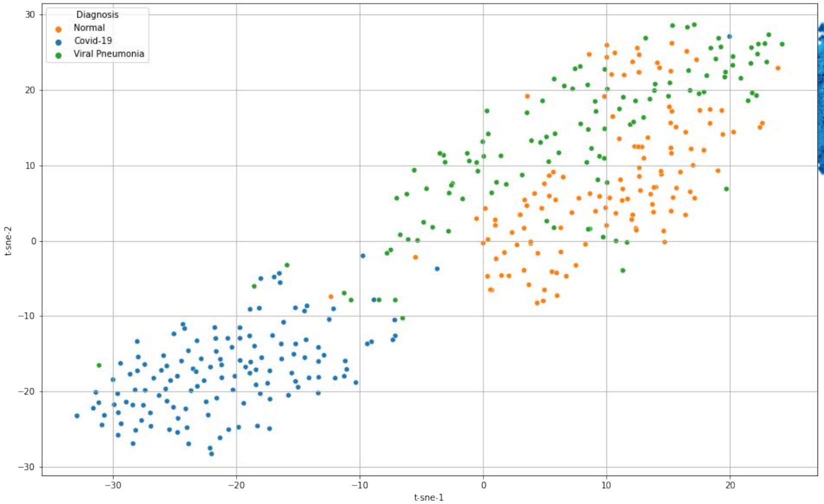


cGAN generated images t-SNE

t-SNE visualization



cGAN uncertainty **max** generated images t-SNE



cGAN uncertainty **min** generated images t-SNE

THANKS FOR YOUR ATTENTION!

Any questions?