HLCV

Augmentation using Generative Adversarial Network

Team # 17

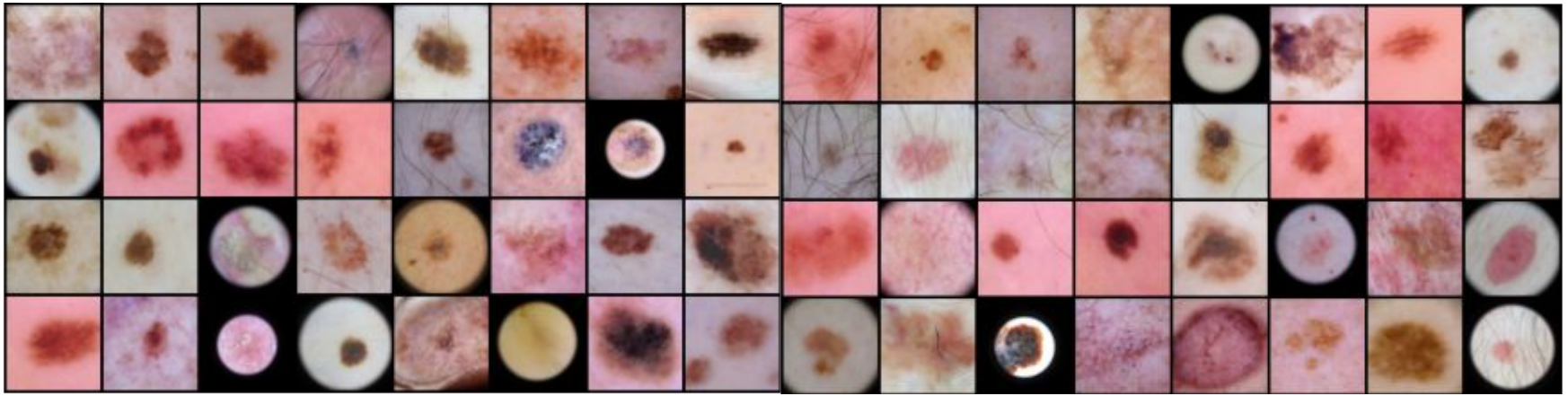
- **Elena Kosheleva (Presenter)**
- Hasan Md Tusfiqur Alam
- Georgii Ivannikov

Project Motivation

- Generative Adversarial Networks have been improving significantly over the years and can now generate very realistic images from little training data.
- Our idea was to use GANs to improve classification for datasets with unbalanced classes.
- We decided to run our experiments on medical images, since there is always insufficient data in medical imaging tasks.
- Our hypothesis is that by adding generated images to the training data of low-frequency classes, we can achieve accuracy improvement on those classes.
- The aim of this project is to prove or disprove this hypothesis.

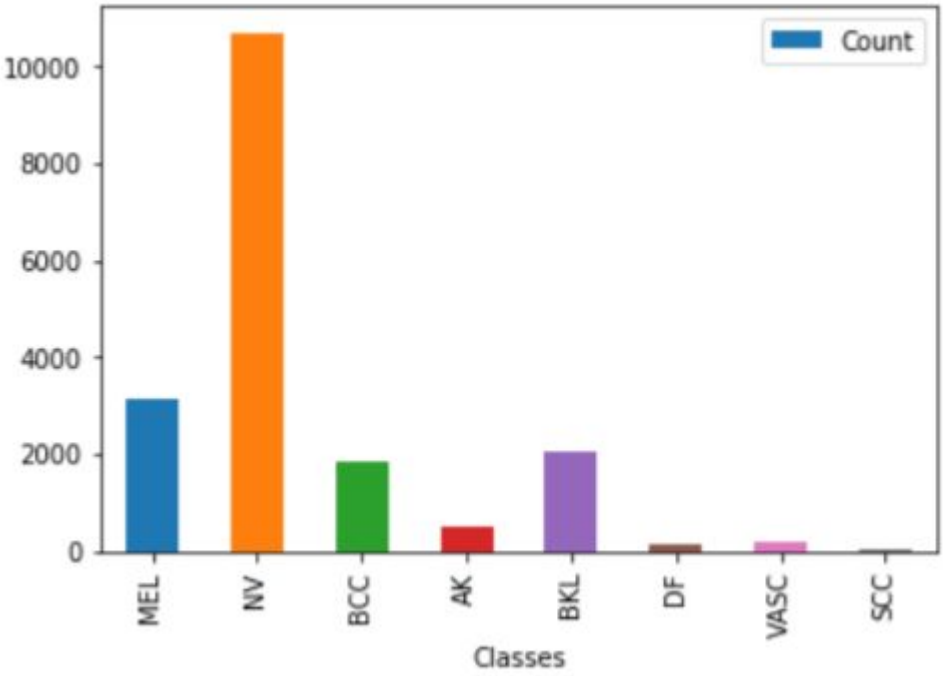
Dataset

We've selected the ISIC2019 Dataset (https://challenge2019.isic-archive.com/) with 25,331 dermoscopic images among **8 different diagnostic categories**



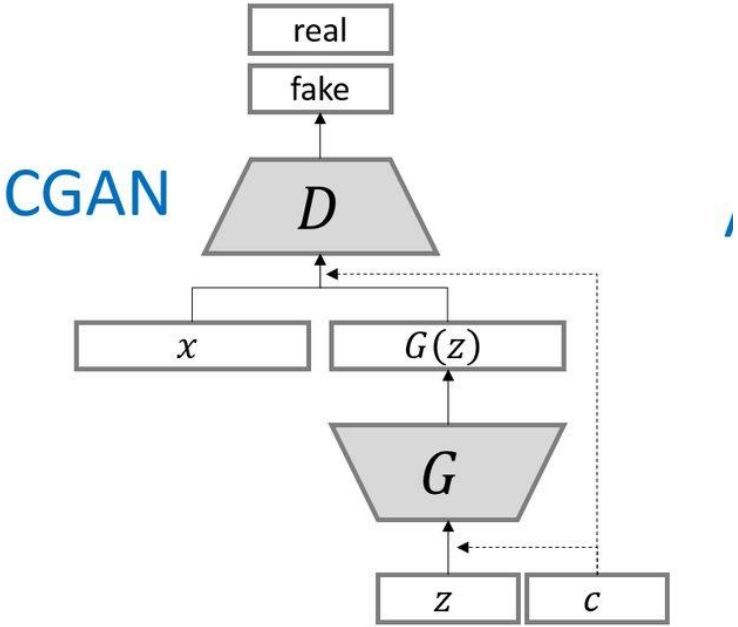
ISIC Class Distribution

- We can Observe the imbalances of classes for the dataset



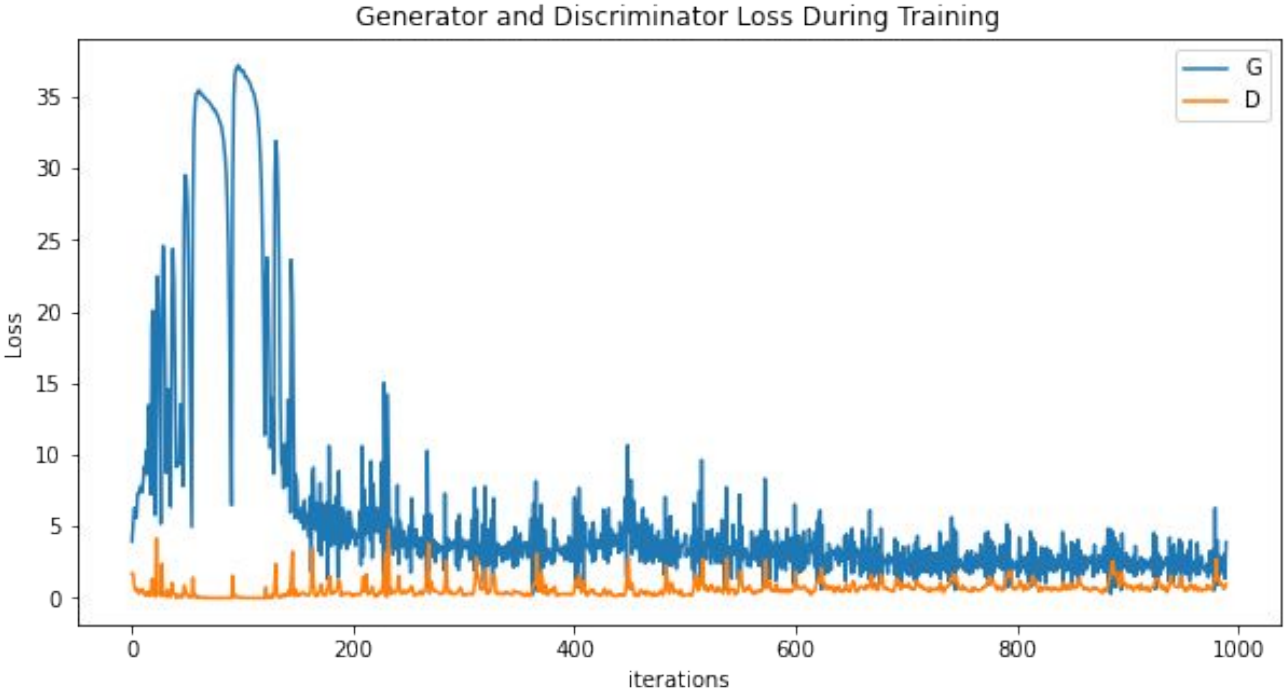
label	count
MEL	4521
NEV	12875
BCC	3223
AK	867
BKL	2624
DF	239
VASC	253
SCC	179

GAN



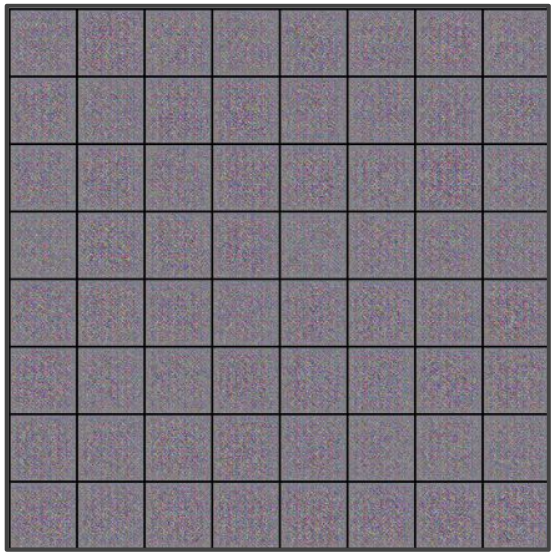
DCGAN Implementation

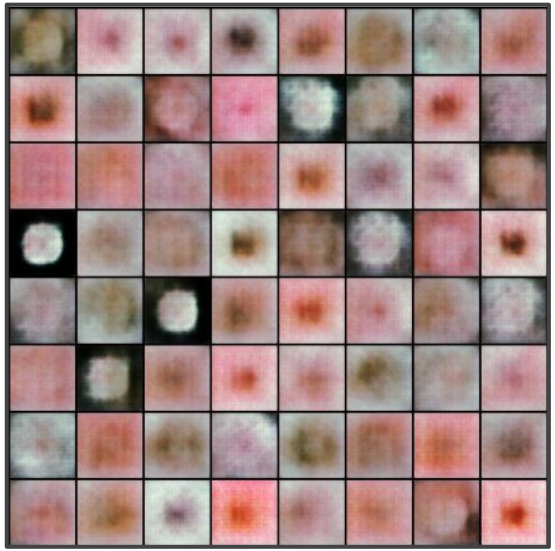
DCGAN (https://pytorch.org/tutorials/beginner/dcgan_faces_tutorial.html) trained on ISIC images without labels

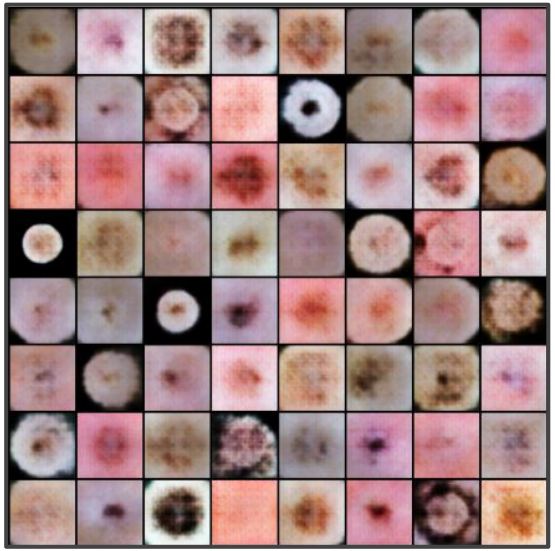


DCGAN Generated Images

Images generated disregarding the classes

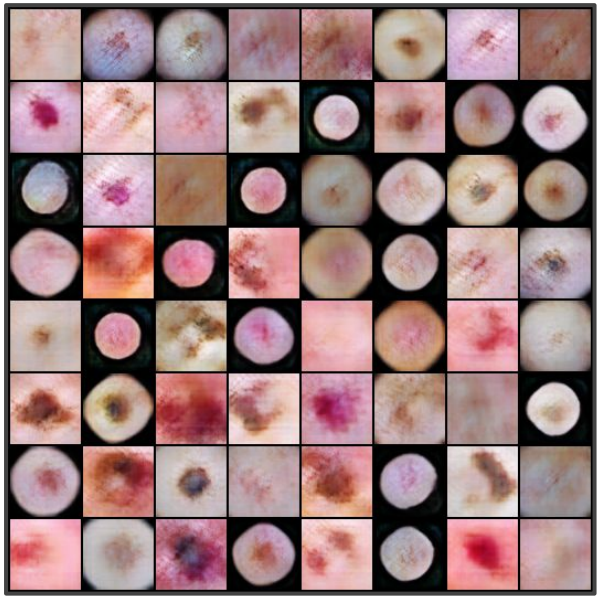






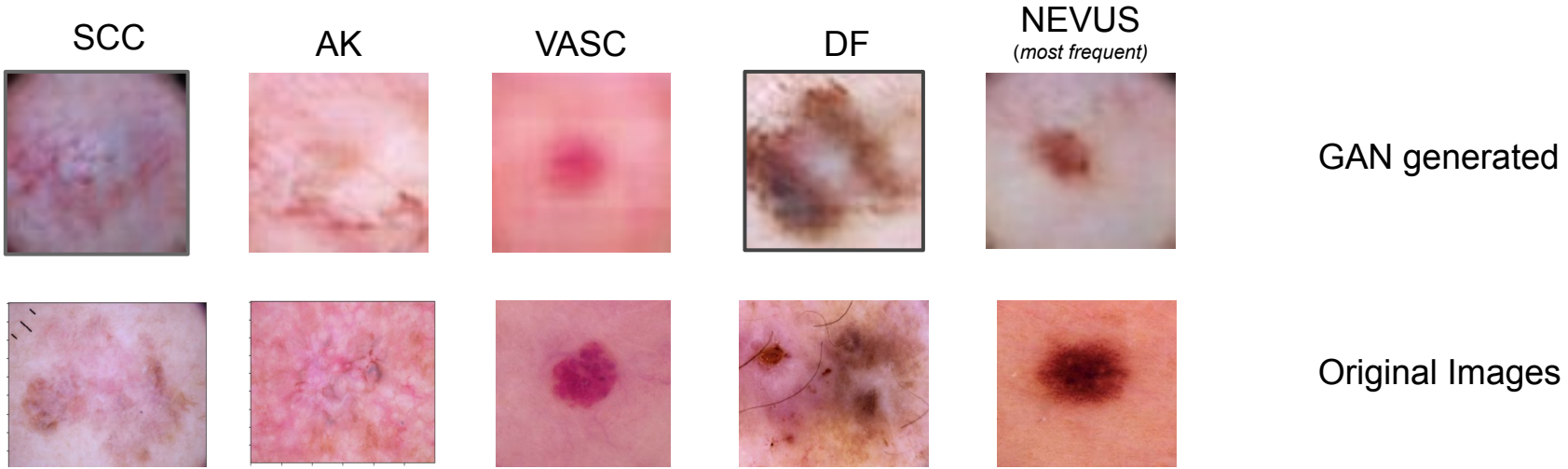
Wasserstein GAN

- We tried several models of conditional GAN
- Best result was achieved with Wasserstein GAN.
- We also oversampled low frequency classes.
- Thus, we could generate quality images of a specific class



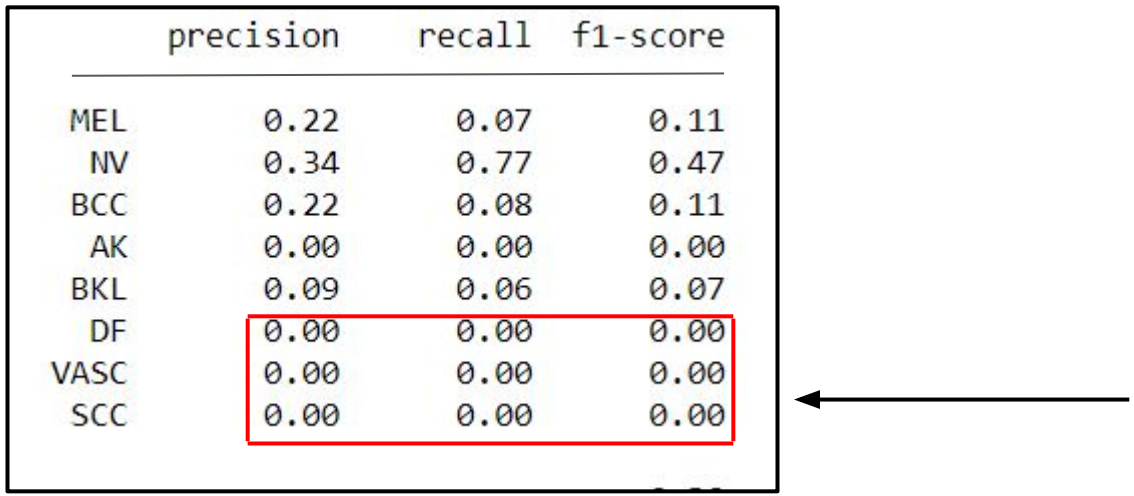
Generated images examples

- The comparison of generated(upper image) and real images of the same class.
  - We chose low-frequency classes for comparison, since generating them is harder



Classification Model

- We used pretrained vgg11 with batch normalization (vgg11_bn) as the base classification model
- **Classification Report for base model:**
  Accuracy: 0.33



	precision	recall	f1-score
MEL	0.22	0.07	0.11
NV	0.34	0.77	0.47
BCC	0.22	0.08	0.11
AK	0.00	0.00	0.00
BKL	0.09	0.06	0.07
DF	0.00	0.00	0.00
VASC	0.00	0.00	0.00
SCC	0.00	0.00	0.00

Classification Result with GAN Augmentation

- 128 images per classes was generated
- Generated images for 4 least frequent classes (**Ak, DF, VASC, SCC)** are augmented in trainset
- **Classification Report with GAN Augmentation on 4 least four classes**
  Accuracy: 0.39

	precision	recall	f1-score
MEL	0.16	0.02	0.04
NV	0.40	0.92	0.56
BCC	0.14	0.01	0.02
AK	0.05	0.01	0.02
BKL	0.00	0.00	0.00
DF	0.06	0.00	0.00
VASC	0.04	0.02	0.03
SCC	0.05	0.00	0.00

Classification Result with GAN Augmentation

- 4 class Classification result for generated images

	precision	recall	f1-score
AK	0.24	0.22	0.23
DF	0.19	0.68	0.29
VASC	0.32	0.11	0.16
SCC	0.16	0.03	0.05

Future work

- As a result of our project, we were able to slightly increase classification accuracy on low-frequency classes using GAN augmentation.

- Based on this promising results, we could try different strategies of mixing real and augmented images (i.e. in proportion 1:2 or 1:3) and run a series of experiments to see which strategy works best.

- Unfortunately, due to limited time we were only able to present proof of concept, which can serve as a basis for future work.

Packages used

Pytorch

Scikit image

Matplotlib

Pandas

Tensorboardx

Click

Scikit-learn

Gpustat

Tensolflow

Jupyter notebook

Reference

- https://github.com/jalola/improved-wgan-pytorch
- https://challenge2019.isic-archive.com

- https://medium.com/health-data-science/using-generative-adversarial-networks-gans-for-data-augmentation-in-colorectal-images-565deda07a22
- https://www.isic-archive.com/#!/topWithHeader/onlyHeaderTop/gallery
- https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia
- https://github.com/sarankrish/gans-data-augmentation
- https://github.com/eriklindernoren/PyTorch-GAN

Thank You