Report

Title: Let's Make Pottery: GANs for Voxel Completion in Ancient Pottery Dataset

Authors:

平行 2200013129

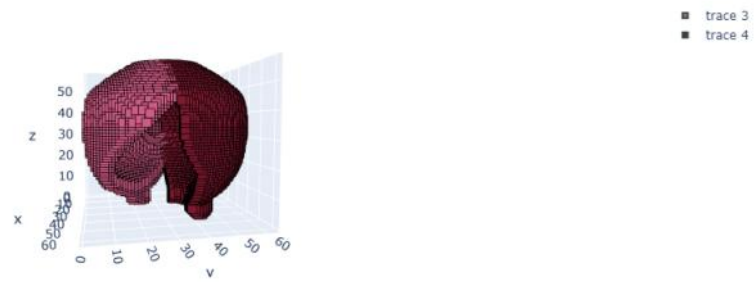
孙新冕 2200013218

Introduction

We utilize the Generative Adversarial Net (GAN) to assist the reconstruction of archaeological artefacts. The problem demands that complete objects be recovered from part of its voxels. We implement a GAN network of the canonical autoencoding architecture, on top of a dataset class specifically tailored for processing the 3D voxel data we're presented with. Further efforts are dedicated to training, optimizing, and engineering the GAN.

1. **Dataset visualization**

Magicavoxel type file can be read as a whole as well as by a fragment of a designated index. A model can be plotted without considering the fragment labels, or coloring the fragments according to their labels, and two models can be plotted together as a joined figure.



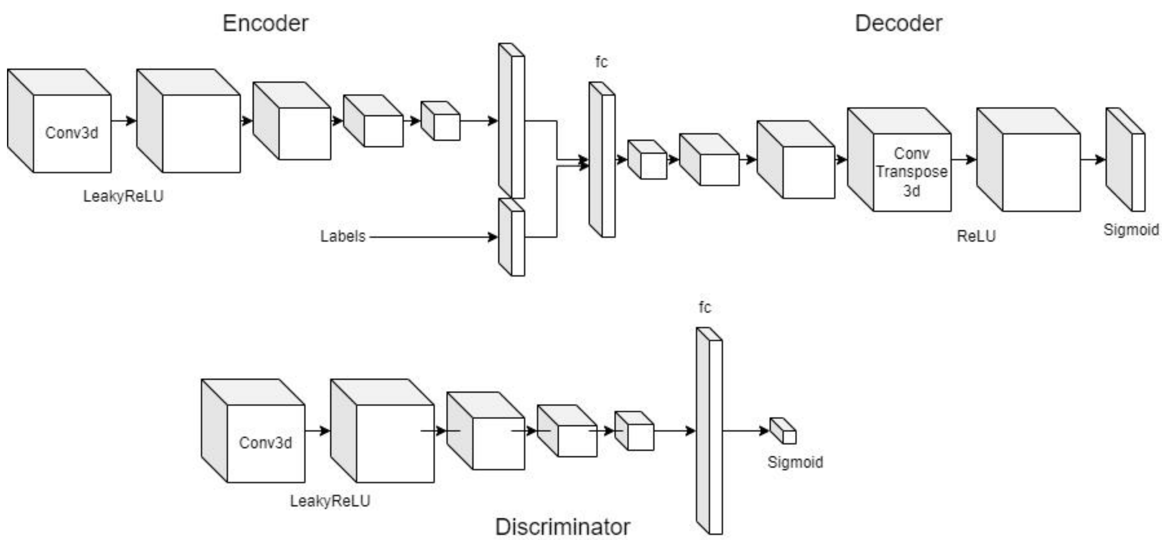
2. **Fragment Dataset class**

A custom Fragment dataset is built for further data processing. The voxel files can be rea
d
by required resolution. When getting the item, we can either randomly or specifically sel
ect a label and extract corresponding fragments in order to customize the need of trainin
g and testing. Fragment data, voxel data excluding the fragments and the class label will b
e returned.

3. **3D AE-GAN**

A Conditional Variational Autoencoder GAN (CVAE-GAN) is implemented. The network co
mprises of a variational autoencoder (VAE) and a discriminator. In order to regularize the
latent space, the encoded data is reparametrized by the mean and log variation of its dist
ribution. Following the approach of CGAN, a labels embedding layer is concatenated to th
e latent space before it being decoded. To mitigate potential gradient vanishing or explod
ing problems, the weights of the decoder are initialized by the Kaiming initializer.

The model's architecture can be illustrated by the diagram below.



The model takes hyperparameters such as input resolution, latent space size and number of classes, and can adapt itself to changes of these parameters.

4. **Training**

The custom training dataset is loaded by the dataloader and 3 instances of models are cr eated: the generator G, the discriminator D, and a classifier C, which is a discriminator ins tantiated with 11 channels of output. During training, the 3 models are trained alternatel y with their respective optimizers and learning rates on odd and even epochs.

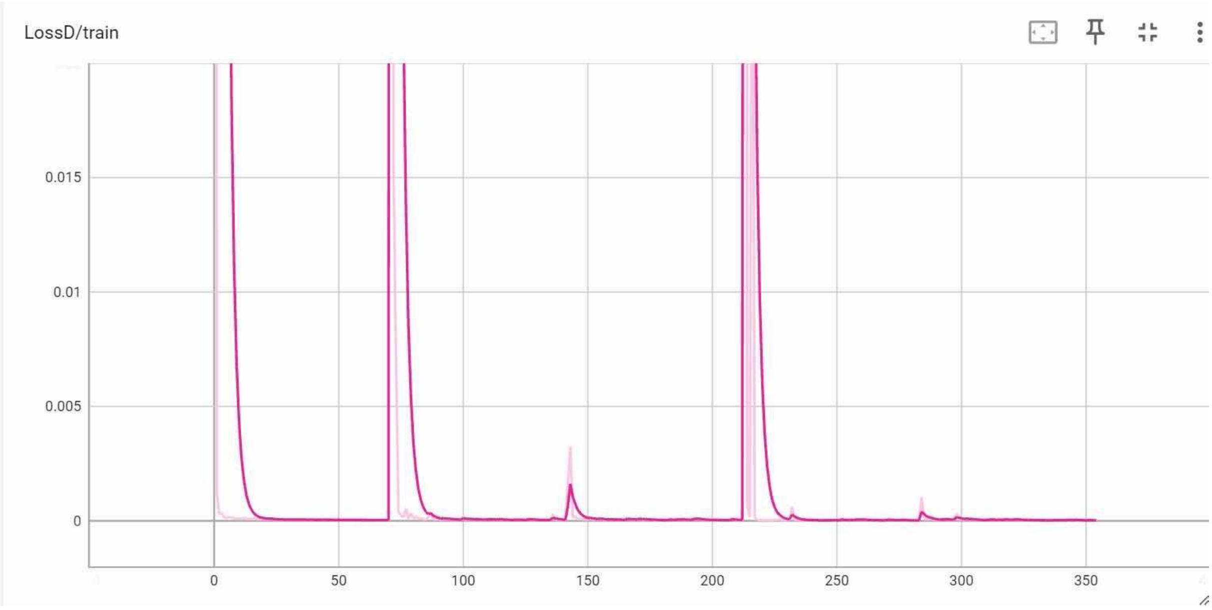
The 3 models have disparate loss functions. The losses are for D a BCE loss (consisting of t he loss for declining a real fragment and the loss for accepting a fake fragment generated by the generator), for C a cross-entropy loss, and for G a composite loss made up of 3 pa rts. Part 1 is the completion loss, the sum of a MSE loss between the reconstructed data and the ground truth and a KL-Divergence loss between the data distribution on the late nt space and the standard normal distribution. Part 2 is the loss for the discriminator atte mpting to discern the fake output. Part 3 is the loss for the classifier attempting to classif y the reconstructed data. Each part is moderated by a scaling factor. The overall loss func tion can be expressed by the following formula.

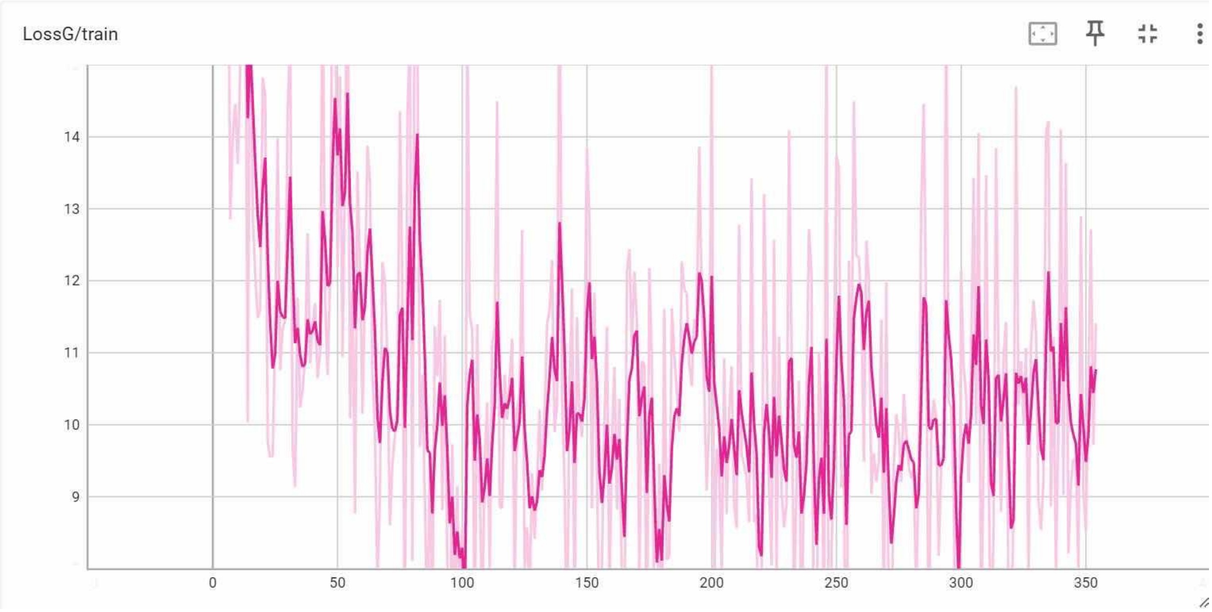
loss_D = -log(D(x))-log(1-D(G(z)))

loss_G = k1*loss_completion+k2*log(1-D(G(z)))+k3*loss_classification

The hyperparameters adopted are: G_lr = 2e-4, D_lr = 2e-4, C_lr = 2e-4, betas (0.9, 0.999) for the AdamW optimizers, a 0.1 ratio for the KLD, and k1, k3 being the k2*1e-4.

The loss curves during training:





5. **Test**

During the training process, the generator's loss is logged by a summary writer at certain intervals. At the end of certain epochs, the model's accuracy is tested on the testing data set.

Testing metrics include DSC and Jaccard-Distance, both of which measure the similarity o f the ground-true fragment and the generated output with values between 0 and 1. 1 rep resents the maximum similarity for DSC but the minimum for JD, so 1 - JD is adopted. The n, the average of the distance is calculated as accuracy. According to the test result, the a verage value of DSC is about 0.983 while JD is about 0.011.

When testing is complete the accuracy is also saved by the summary writer. A voxel file is selected from the testing dataset to produce a joined plot with the reconstructed data fo r qualitative evaluation purposes.

6. **Task assignment**

平行

is responsible for implementing the CVAE-GAN model, conducting the training function and loss function, and tuning the parameters with the training set on a GPU.

孙新冕

is responsible for implementing the dataset class, performing visualizations, conducting the test function and testing the model's accuracy on the test set.