GSoC’25 ML4SCI EXXA1

General Test: Unsupervised Image Clustering

Jupyter Notebook : here

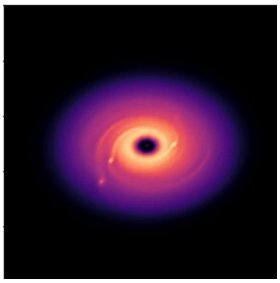
Approach & Discussion:

- **Architecture:**
    - Trained a D(8)-truly Equivariant encoder (8 rotations + 2 flips) with escnn library that mimics E(2) equivariance, then fed it into a mirrored D(8)-Equivariant decoder for reconstruction. For latent vectors applied group pooling along with global average pooling in the output of encoder to get invariant latent vectors with respect to camera orientations.
    - Traditional Autoencoders (VGG like & resnet-18 encoder based) failed miserably, they didn't capture fine structures, kinks and turbulences and also were not equivariant to E(2) transformations so adopted an almost E(2) equivariant one.
    - For entire training (Self-Supervised based pretraining didn't help) used MSE and pretrained ResNet-18 perceptual loss for reconstruction and added a term (inspired from ViCReg paper) to encourage orthogonality of latent vectors to make full use of latent space.

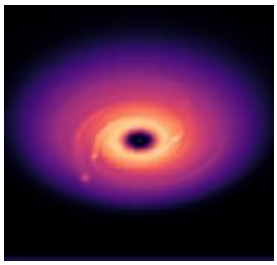
- **Augmentation**:

  During Training, used Affine transformations - random rotations, flipping, scaling and shearing to mimic possible camera orientations while imaging these disks in 3D space (didn't use perspective transforms in training here,

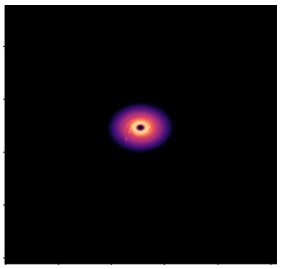
doesn't perform well for 256*256 images due to interpolations) , also added gaussian smoothing, to smoothen out some very grainy images



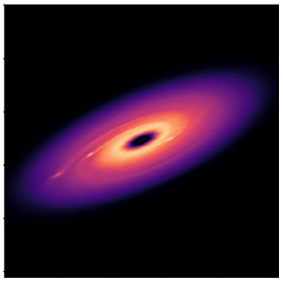
Original Disk



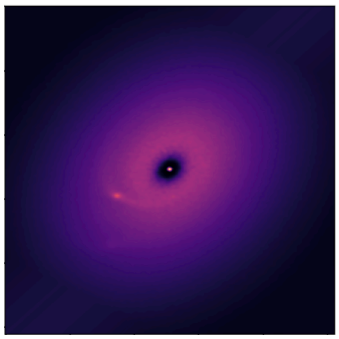
Perspective Transformed



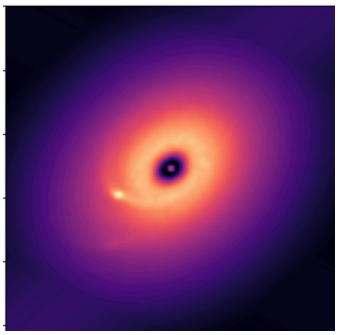
Scaled down



Sheared + Rotated



Grainy Image

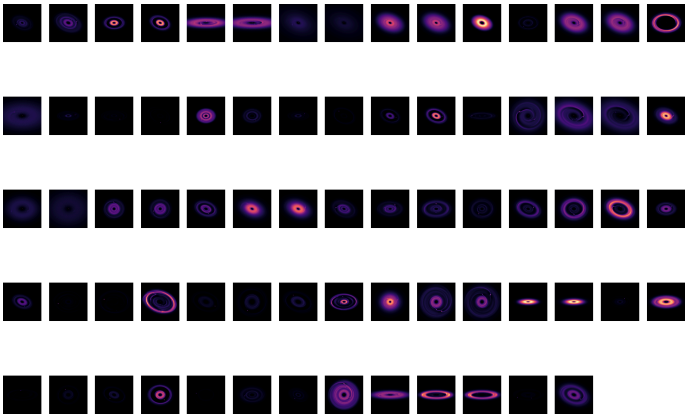


Gaussian Blurred

- **Clustering:**
    - Using the group pooled vector and applying global average pooling to squeeze out the spatial dimensions to get latent vectors which I clustered through Spectral Clustering from sk-learn, I got clusters

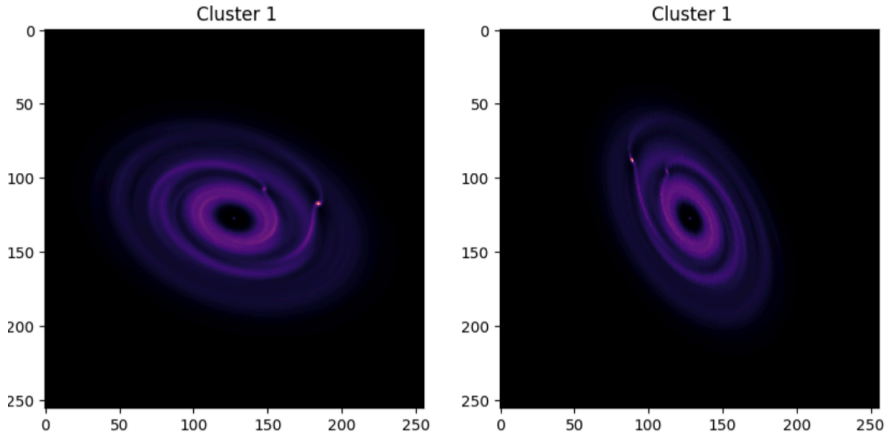
which did to some extent cluster images that had clear planets and images with clearly no planets separately .

Cluster 0
(almost all are not having any kinks/disturbances)

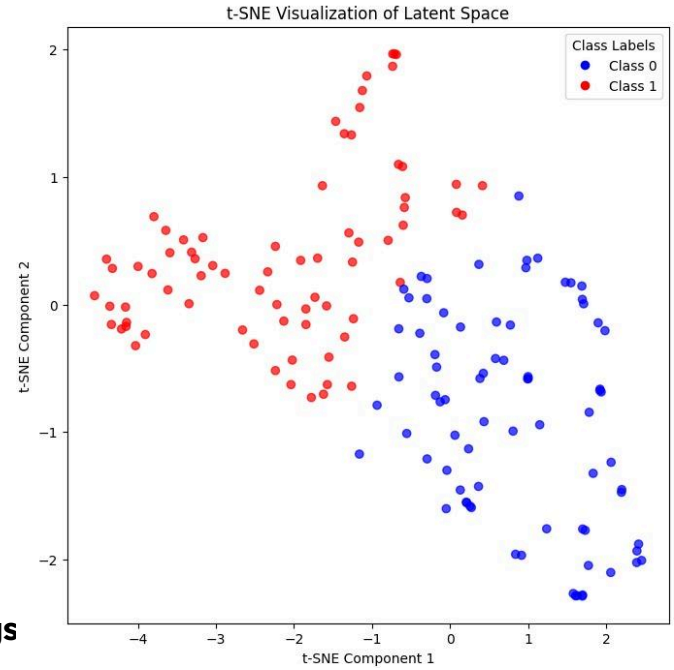


Cluster 1
(almost all are having some kinks/disturbances
indicating planet presence/self-gravitating disk etc.)

- One thing interesting to note was on affine transformation of any image , its cluster index did not change indicating invariant nature of latent vector with respect to affine transformations.

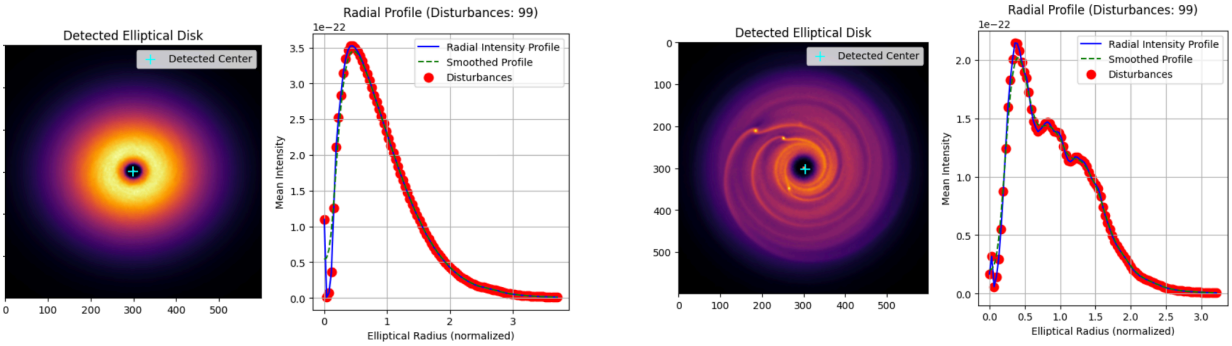


- t-SNE visualization for clusters :

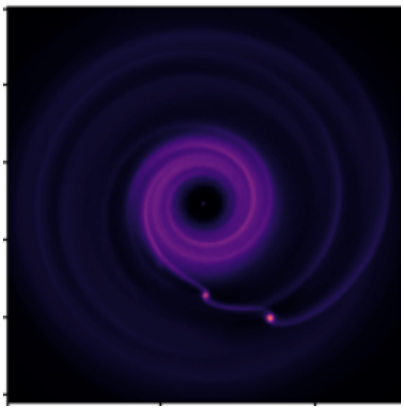


- **Other things**
  - One last thing I want to add , I tried to compute elliptical radial intensity profiles of these images which did provided information

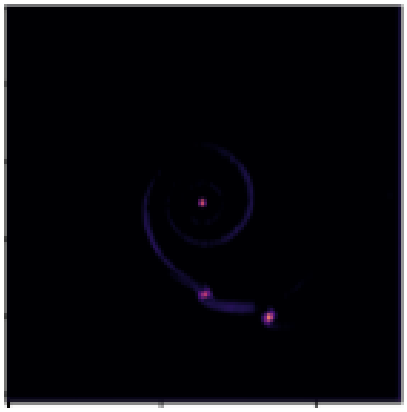
about these kinks/turbulences and whole problem was bring down much smaller space from 2D to 1D frequency domain but discontinued my approach as I didnt felt doing this now for tests.



- Computed Spectral Laplacian filtering, also called Fourier-Laplacian filtering and Difference of Gaussian to highlight edges and globs in the images which narrowed down the information needed for planet prediction from the images , after applying threshold and connected-components I could sort out images with no blobs.



Original Image



Transformed Image

Image Based Test: Autoencoder with accessible latent space

Final Results:

- **Achieved MSE 0.091, 0.043 (single, ensemble x10) & MSSSIM 0.971 with ensemble of 10 K-folds.**
-

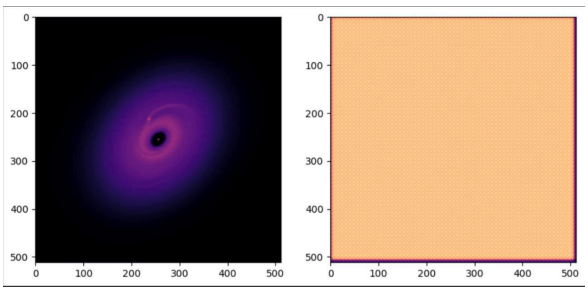
Discussion of Approach:

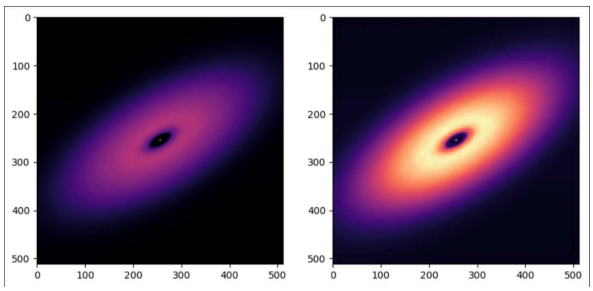
- **Augmentation:**

  Same as above

- **Architecture:**

  Same as above but here increased more weightage to perceptual loss to encourage structure of image more than MSE or orthogonal of loss , didn't use MSSSIM in training loss as after visual inspection didn't gave good results model was finding patterns to cheat loss reductions rather capturing structures (memorising over learning accurate images see below images) . This latent space collapsing problem was solved with thinning down architecture + perceptual loss (similar to human eye) over MSE & MS-SSIM + ensembling , also tried dropout in decoder but did not affect much.





- **Analysing Latent space** :-

    - I found that E(2)-transformed versions of images were very close in latent space, implying invariant nature of latent vectors towards the E(2) group.

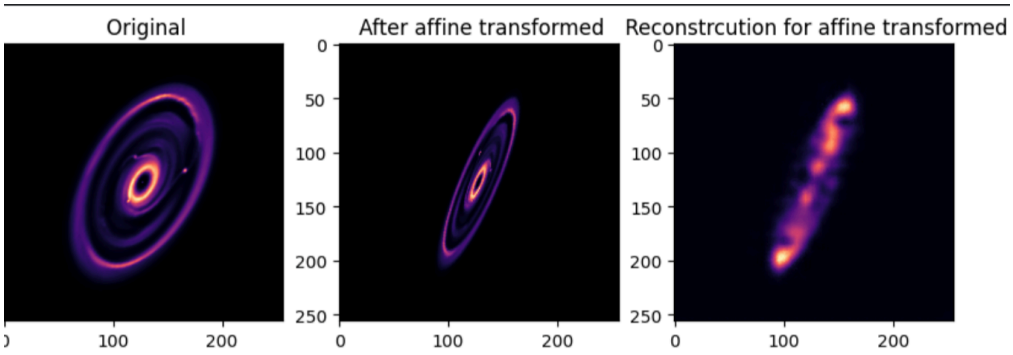
- **Why Ensembling:**

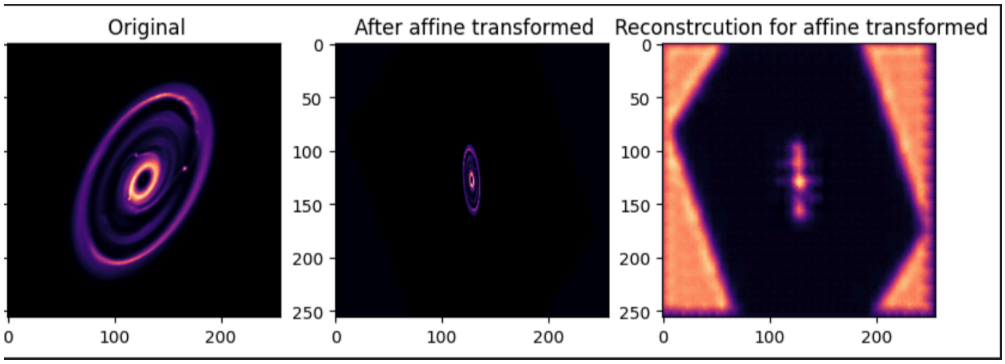
    - Due to extremely small size of dataset , model was proned to overfit if right architectures and losses were not chosen , single models were finding global patterns that minimised overall loss over the dataset hence losing to capture all variety of disks. With ensembling , each model was an expert to its own training set , meaning an expert in his domain , so in average voting effects overfitting were nullified due to a collection of models (like a jury) giving much better results.

- **Other things I tried:**

    - **Other Architectures I tried that didn't help:**

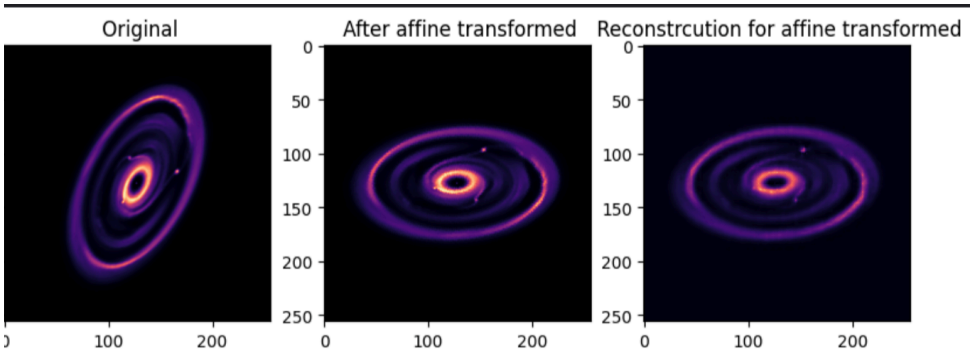
      Traditional AE didn't help , they were failing miserably in reconstruction, probably data limitation was the issue , so adopted an equivariant one.

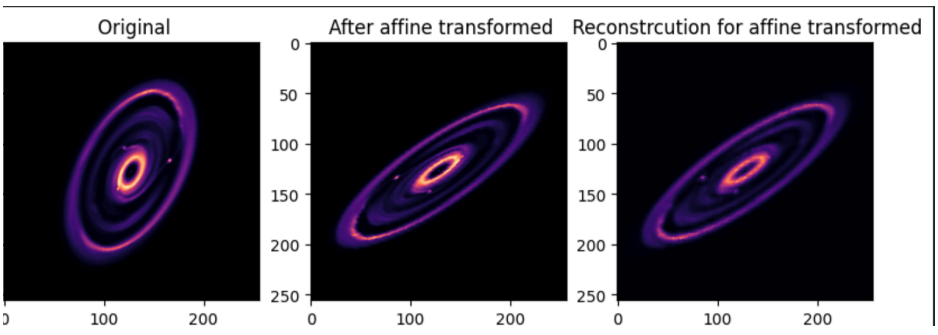




Reconstructions given by Traditional AE

Traditional AE failed to capture details when Affine transformed Images was fed , probably because even with data augmentations during training the model is failing in equivariance because of lack of diversity in the training data in such a small 150 image data.

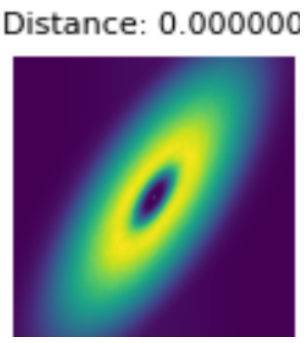




Reconstructions by my D(8) equivariant AE

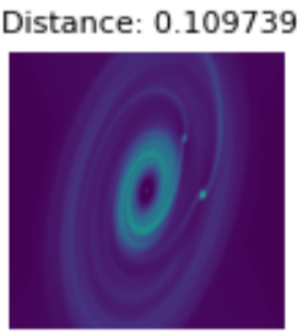
- While equivariant AE on the other hand , explicitly devises kernels for D(8)-group transformed version of feature maps not relying on self-organising nature of traditional vision networks when trained on huge diverse data with augmentations hence perfect for small datasets with known possible symmetries.

- I tried contrastive learning for pre training because euclidean distances in latent space were large between similar disks at different inclination angles probably because they largely differ in shapes but it didn't help maybe because of maybe wrong hyperparameters (choice of augmentations, temperature, latent dim, small dataset etc), so in future I would like to try equivariances more groups like SIM(2) , PGL(3) etc to generalize even more because with augmentations to achieve equivariance is hard with small less diverse datasets and also low resolution images suffer loss in characteristics due to interpolations with such transformations like perspective transformations.



Original Image

Disk with similar Architecture
But different inclination angle



Disk with totally
different architecture