Image embedding in pseudorandom latent space

Master thesis defence - LM Computer Science

Leonardo Monchieri

12/12/2024

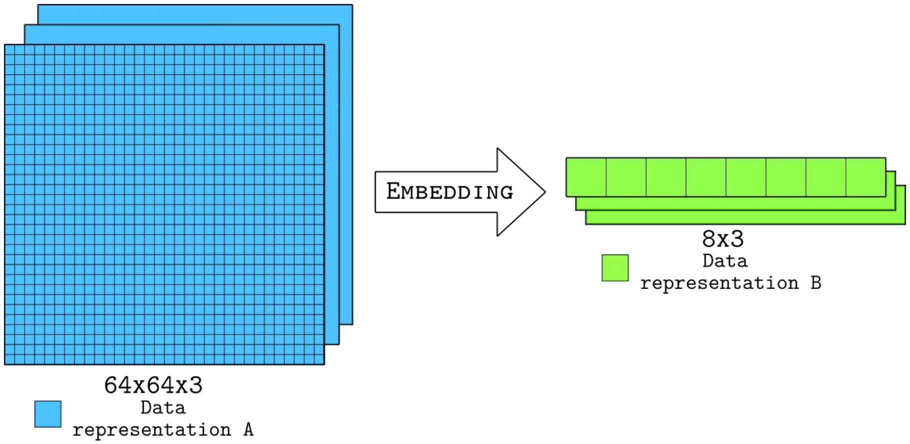
Supervisor: Prof. Simone Milani
Co-supervisor: Dott. Daniele Mari

Università degli Studi di Padova

- **Introduction**
- Latent codes
- Datasets
- Training and optimization
- Analysis and results
- Conclusion

Data Embedding

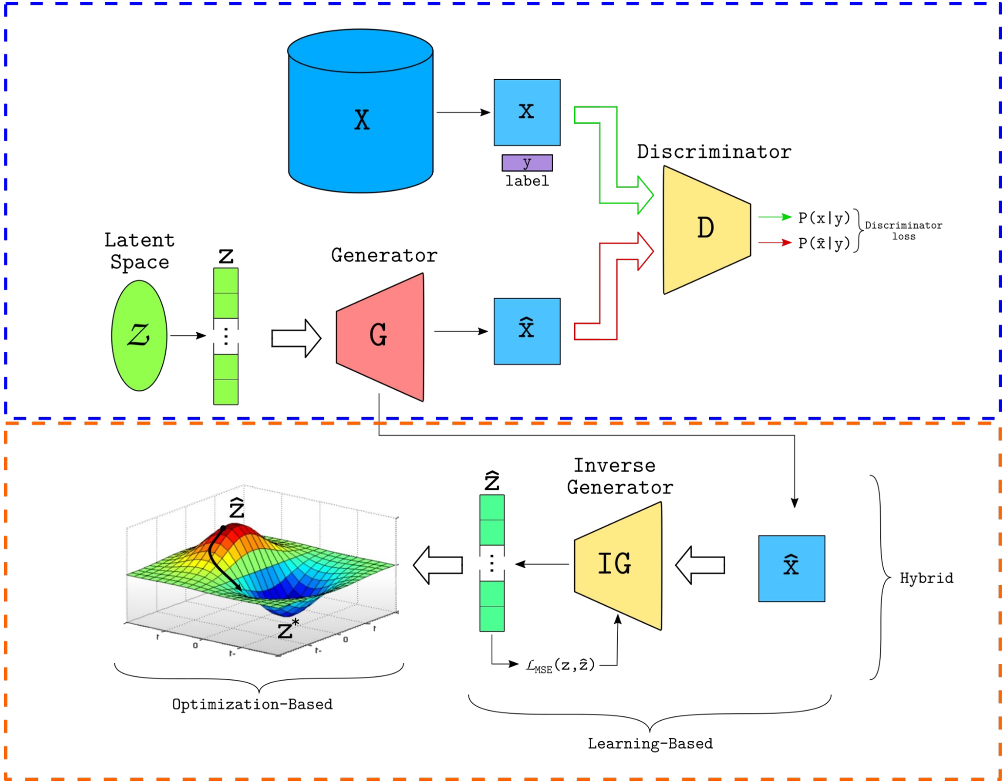
Embedding: transformation from a data representation **A** to a data representation **B**



- Data security: encryption
- Data manipulation: context related representations
- Data transfer } compression
- Data store

GAN

Inverse GAN



- Introduction
- **Latent codes**
- Datasets
- Training and optimization
- Analysis and results
- Conclusion

Pseudorandom functions

Università degli Studi di Padova

A function $F:S\rightarrow Y$, where S is the seed space and Y the output space, is a pseudorandom function(PRF) if:

- *Efficiently computable*: there is an polynomial time algorithm that given a seed $s \in S$ can compute $F(s)$,
- *Pseudorandom*: for any polynomial-time adversary, it is computationally infeasible to distinguish between the function F and a random function $f:X\rightarrow Y$.

Extra requirement

- *Invertibility*: exist an inverse function $F^{-1}:Y\rightarrow S$ that given an output $y\in Y$ return the seed $s\in S$ that generated it.

Chaotic map

Hénon map

$$x_{n+1} = 1 - ax_n^2 + bx_{n-1}$$

Logistic map

$$x_{n+1} = rx_n(1 - x_n)$$

Tent map

$$x_{n+1} = \mu \, min(x_n, 1 - x_n)$$

Linear Congruential Generator(LCG)

$$x_{n+1} = (ax_n + c) \mod m$$

Sequences correlation

- *Cosine similarity*: measures the cosine of the angle between two non-zero vectors, quantifying how similar they are regardless of magnitude.
- *Pearson correlation coefficient(PCC)*: measures the linear relationship between two variables, reflecting how changes in one predict changes in the other.

Randomness

- *Kolmogorov-Smirnov test (KS)*: compares the cumulative distributions of the sequences with a reference distribution to assess whether they differ significantly.
- *Wald-Wolfowitz runs test (run test)*: evaluates the randomness of a sequence by analyzing the arrangement of runs (consecutive similar elements) in the data.

Latent codes

Distribution/PRF	**Cosine similarity**	**PCC**	**KS Test**	**Run Test**
Uniform distribution	-2.09e-07	-2.13e-07	0.505868	-0.019535
Gaussian distribution	5.28e-03	1.43e-06	0.185694	0.011459
Hénon map	1.27e-03	7.32e-05	0.364905	0.484668
Logistic map	7.97e-03	2.52e-05	0.198464	4.326132
Tent map	8.43e-03	1.30e-05	0.144591	2.495715
LCG	-1.46e-05	-1.38e-05	0.504770	-0.004743

- Introduction
- Latent codes
- **Datasets**
- Training and optimization
- Analysis and results
- Conclusion

Datasets

MNIST dataset

- Handwritten digits
- 28x28 pixels images
- 70.000 images
- 10 classes

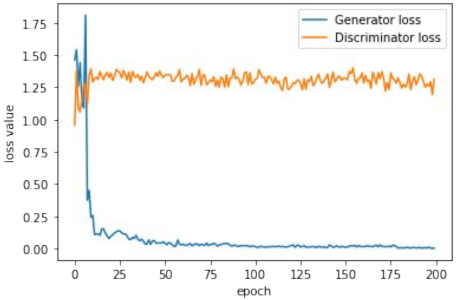
Anime faces dataset

- Anime character faces
- 64x64 pixels images
- 63.632 images
- No classes

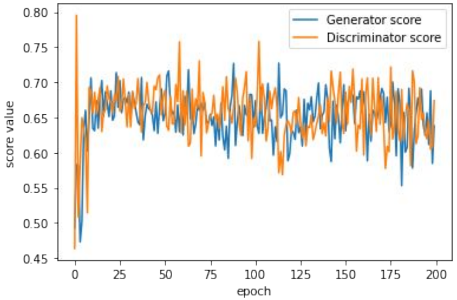
- Introduction
- Latent codes
- Datasets
- **Training and optimization**
- Analysis and results
- Conclusion

MNIST training process

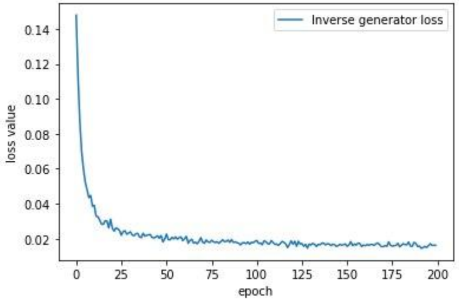
GAN losses



GAN scores



Inverse generator loss

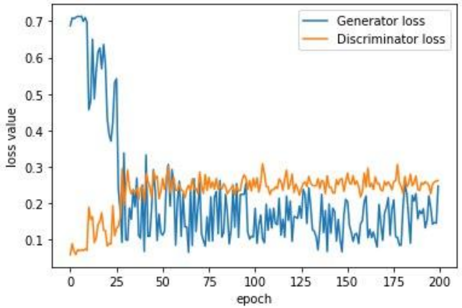


- 200 epochs for GAN and 200 epochs for inverse Generator
- sequences of 25 symbols

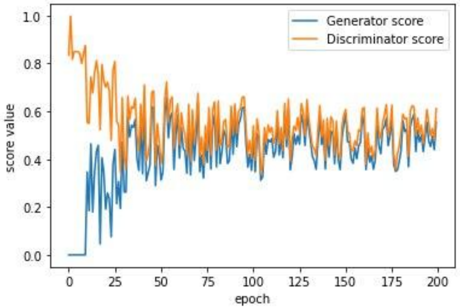
Generalization

- *GAN 2 steps*: each epoch has two steps, one with sequences from a uniform distribution and one with LCG sequences.
- *GAN LCG + noise*: addition of sequences from a uniform distribution to the LCG sequences.

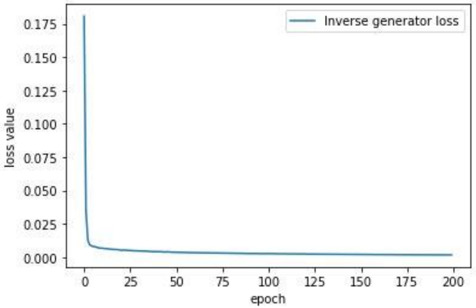
GAN(5101) losses



GAN(5101) scores



Inverse generator(5101) loss



- 200 epochs for GAN and 200 epochs for inverse Generator
- Sequences of 100 symbols

Parameters exploration

- *GAN(5101)*: a=1733, c=1627 and m=5101
- *GAN(29282)*: a=1255, c=6173 and m=29282
- *GAN(524287)*: a=283471, c=35899, m=524287

Seed search

Università degli Studi di Padova

Input: $z \in \mathbb{Z}^{n \times d}$ {Sequences generated by the IG}
Parameters: ϵ {Size of the nearby}
Output: $seeds = [s_1, ..., s_n]$ s.t. $\forall s_i \in s, s_i \in [0, m]_{\mathbb{N}}$ {Seeds of the sequences}

Initialize $seeds \in \mathbb{Z}^{n \times 1}$ with zeros

Latent denormalization

$denorm\_z \leftarrow Denormalize(z)$ {return z from range[-1,1] to [0,m]}
$denorm\_seeds \leftarrow denorm\_z[:0]$ {get first element of each de-normalized sequence}

for $i = 0$ to $len(denorm\_seeds) - 1$ **do**

Neighborhood search

  $error \leftarrow \infty$
  for $t = s[i] - \epsilon$ to $s[i] + \epsilon$ **do**
    if $t < 0 \vee t > m$ **then**
      Continue
    end if
    $gen\_seq \leftarrow LCG\_sequence(t)$
    $error_i \leftarrow \ell_{MSE}(z[i], gen\_seq)$
    if $error_i < error$ **then**
      $seed_i = t$
    end if
  end for

Optimal seeds collection

  $seeds[i] \leftarrow seed_i$
end for
return $seeds$

Noise composition

Latent code (LCG or Uniform) as linear composition of LCG:

$$\dot{z} = w_1 LCG(s_1) + w_2 LCG(s_2) + \ldots + w_n LCG(s_n)$$

Iterative compute vectors *s* and *w* as follow:

$$w_i = \min\left(z_{i-1} \otimes LCG_{map}\right) ,$$
$$s_i = \underset{s}{argmin}\left(z_{i-1} \otimes LCG(s)\right) ,$$
$$z_i = z_{i-1} - w_i LCG(s_i) .$$

Three composition levels taken into account: 5, 10 and 20.

- Introduction
- Latent codes
- Datasets
- Training and optimization
- **Analysis and results**
- Conclusion

Metrics

- *Mean Square Error (MSE)*: allow us to understand how much of the original data has been lost in the embedding operation.

- *Structural Similarity Index Metrics (SSIM)*: to capture the reconstruction quality, focusing on preserving structural details and perceived sharpness in the generated images.

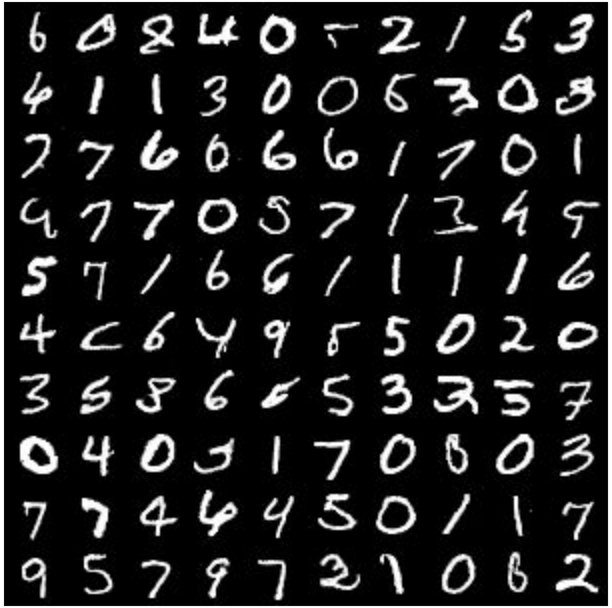
- *Fréchet Inception Distance (FID)*: to assess the similarity in distribution between generated images and real images, focusing on capturing both high-level feature alignment and overall image quality.

Synthetic dataset and Test dataset

Synthetic Dataset

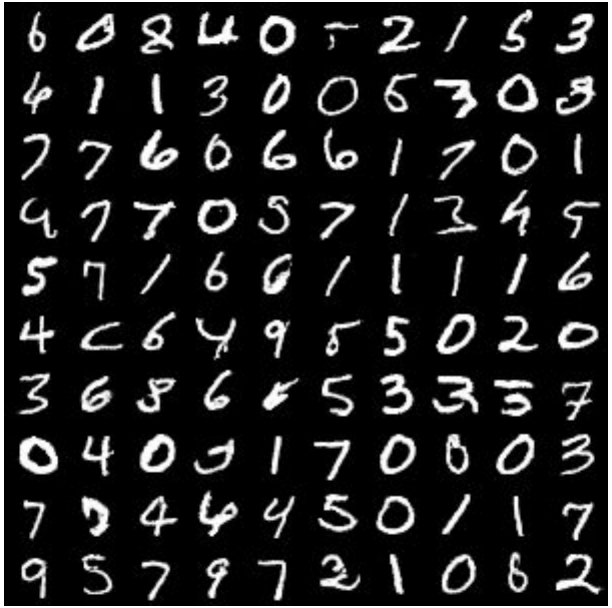
Training type	MSE (latent)	MSE (img)	SSIM	FID
Inverse GAN	0.012189	0.041274	0.776580	32.721333
Inverse GAN LCG+noise	0.014229	0.033165	0.808409	30.506159
Inverse GAN 2 steps	0.015392	0.038858	0.800476	31.874275

Synthetic Dataset reconstruction samples



Original samples

inversion+seed search



GAN 2 steps (reconstructed)

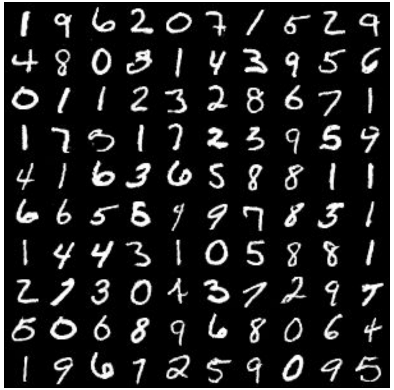
Test Dataset

Training type	MSE (img)	SSIM	FID
Inverse GAN	0.280662	0.547575	24.431034
Inverse GAN LCG+noise	0.264190	0.558244	24.653860
Inverse GAN 2 steps	0.274251	0.551294	23.280581
Autoencoder	0.193280	0.549797	135.027039

Test Dataset reconstruction samples

Original samples

inversion+seed search



GAN 2 steps (reconstructed)

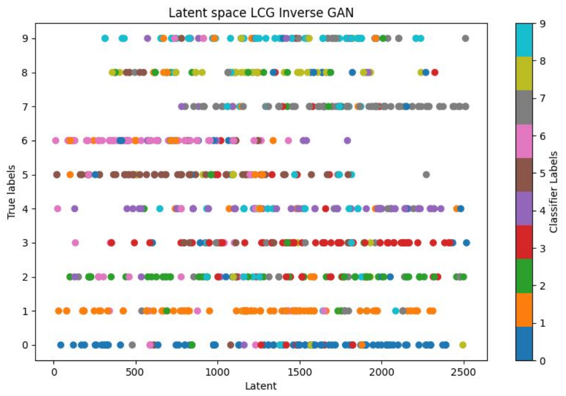
Encoding+decoding

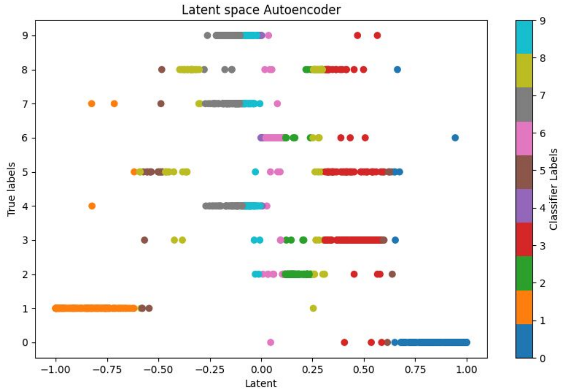
Autoencoder (reconstructed)

Classification test

Cross-entropy: measures the difference between the predicted probability distribution and the true label distribution.

Method	CE metric
Inverse GAN	4.022841
Inverse GAN LCG+Noise	3.766780
Inverse GAN 2 steps	3.624026
Autoencoder	3.646455

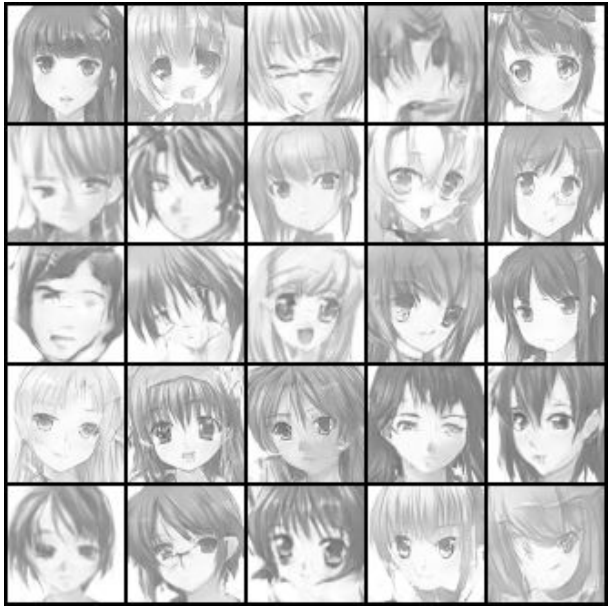




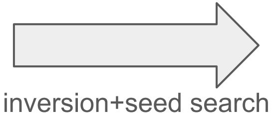
Synthetic Dataset

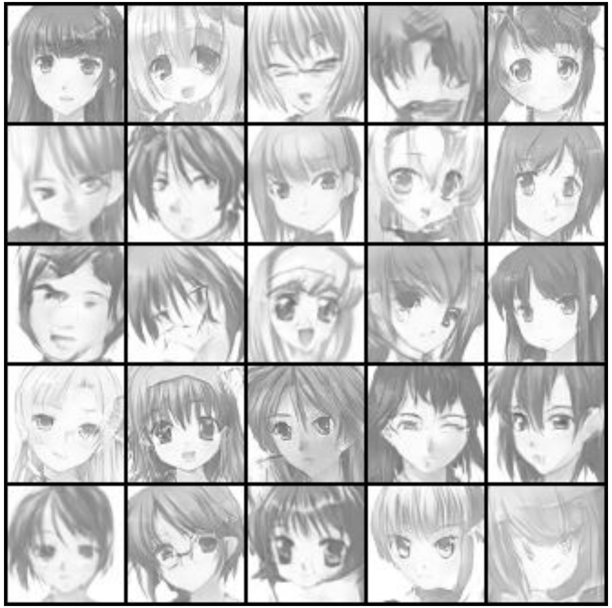
Training type	MSE (latent)	MSE (img)	SSIM	FID
Inverse GAN(5101)	0.001819	0.001403	0.9681	17.2624
Inverse GAN(29282)	0.011398	0.006256	0.8759	29.8182
Inverse GAN(524287)	0.056614	0.010546	0.8153	34.5797

Synthetic Dataset reconstruction samples



Original samples



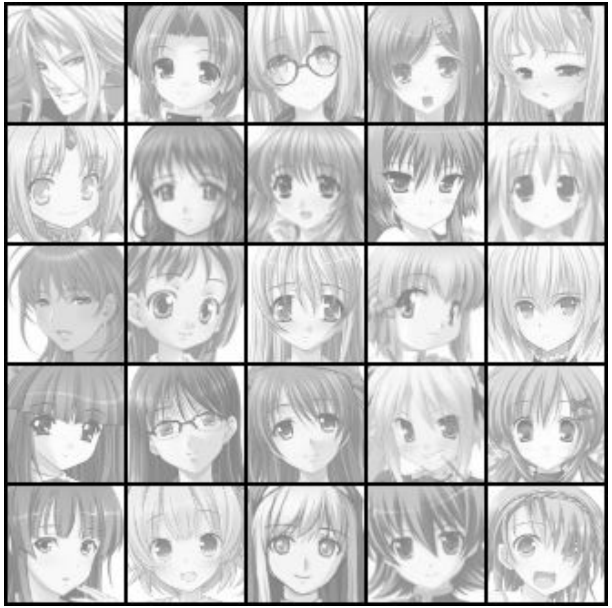


GAN 5101 (reconstructed)

Test Dataset

Training type	MSE (img)	SSIM	FID
Inverse GAN(5101)	0.091134	0.146485	131.553131
Inverse GAN(29282)	0.096304	0.144632	108.923164
Inverse GAN(524287)	0.0855831	0.162683	114.904045
Autoencoder	0.043306	0.258801	324.929474

Test Dataset reconstruction samples

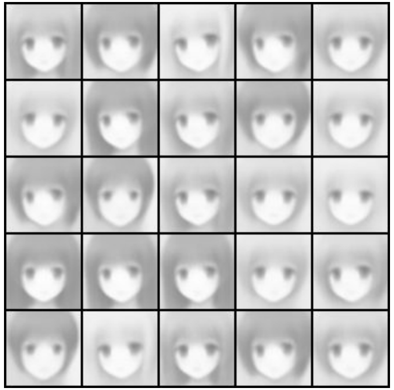


Original samples

inversion+seed search

GAN 524287 (reconstructed)

Encoding+decoding



Autoencoder (reconstructed)

Noise composition test

Synthetic dataset

Compression	MSE (img)	SSIM	FID
Uniform 5	0.050898	0.383645	40.323863
Uniform 10	0.032227	0.536680	39.302006
Uniform 20	0.015579	0.710165	33.472889
LCG 5	0.065070	0.264013	27.386955
LCG 10	0.045012	0.404454	26.318813
LCG 20	0.021814	0.647998	23.057238

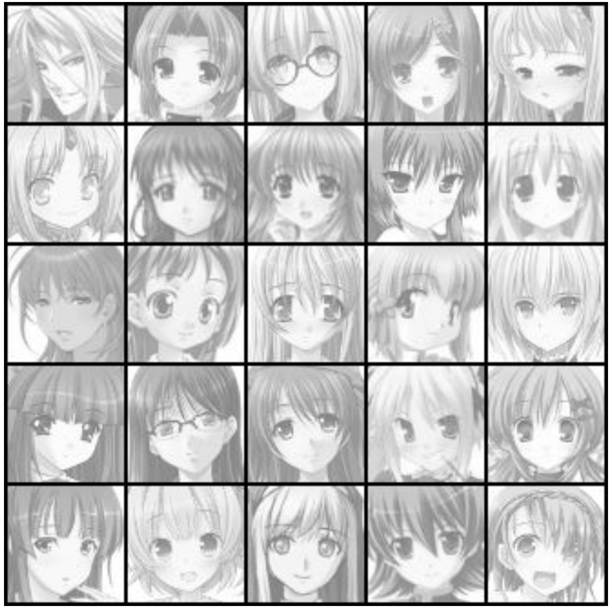
Test dataset

Compression	MSE (img)	SSIM	FID
Uniform 5	0.056232	0.246840	115.374329
Uniform 10	0.043571	0.300170	114.668144
Uniform 20	0.035309	0.354097	113.415405
LCG 5	0.069271	0.195917	163.398407
LCG 10	0.124997	0.106405	163.898788
LCG 20	0.124159	0.105760	163.612579

Autoencoders comparison

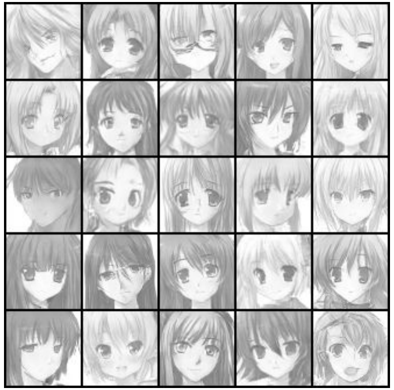
Architectures	**MSE (img)**	**SSIM**	**FID**
AE(7)	0.033821	0.364243	311.198334
AE(14)	0.028392	0.435497	285.047241
AE(28)	0.025243	0.494204	249.512772
Uniform 20	0.035309	0.354097	113.415405

Test Dataset reconstruction samples



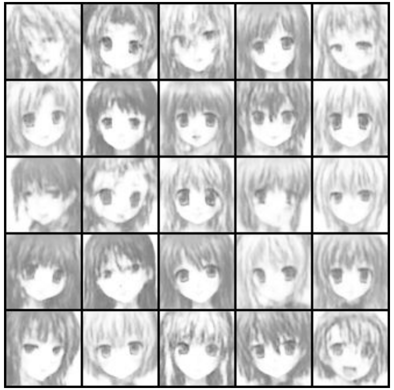
Original samples

inversion+noise composition



Uniform 20 (reconstructed)

Encoding+decoding



Autoencoder 28 (reconstructed)

- Introduction
- Latent codes
- Datasets
- Training and optimization
- Analysis and results
- **Conclusion**

Conclusion

Recap

- Novel approach in the image embedding context
- LCG latent codes for high compression ratio
- Inverse GAN architecture to control the latent
- Optimization algorithms to find optimal embedding
- Low bit-rate with generated datasets or high prior datasets
- Poor results with heterogeneous datasets

Further work

- Seed search and noise composition optimization
- Style GAN architecture
- Different Pseudorandom functions

Thank you for the attention

questions?