Electrical and Electronics Engineering
Semester Project
Master Semester 3 - Winter 2020

# Interpretation of Latent Space in Deep-Learning Based Image Compression


*Student:* Zengyu Yan

*Supervised by:* Michela Testolina
Dr. Evgeniy Upenik
Prof. Dr. Touradj Ebrahimi


January 14, 2021


Multimedia Signal Processing Group
EPFL


**Abstract**

Images and videos have become the main resource for people to receive information in the age of internet. This type of data requires a lot of storage space on hard drives as well as resources of communication channels. Therefore, it is necessary to compress images into lower bits to ensure efficient storage and transmission. In recent years, a lot of deep learning-based image compression codecs have been developed, which uses a completely different architecture than the conventional DCT block-based and wavelet image coding. Those leaning-based techniques have been shown to achieve state-of-art performance in many areas. In this project, we will review, characterize and summarize some of the most promising end-to-end image compression schemes with neural networks. Experiments will also be performed to compare the performance of selected codecs in terms of objective assessment matrices. In addition, we will specify one deep-learning based image codec and extract latent space vector from it to see if any modifications on this vector will cause meaningful changes in the output.
**Keywords:** image compression, deep learning, quality assessment, latent space

# Contents

# 1 Introduction

Image compression algorithms are aimed at exploiting the spatial redundancy and perceptual irrelevance of an image by exploring the characteristics of human visual system, thus create a lower dimensional representation of the original image/video. Traditional image/video compression techniques such as JPEG are based on blockwise-transform. Their underlying algorithm can be summarized as first dividing the image into multiple blocks, and then applying orthogonal transform such as DCT to those blocks. In this way, the spatial frequency information is extracted from the spatial amplitude samples. Since human visual system is less sensitive to high frequency components, the low frequencies will therefore be maintained and quantized with a pre-defined quantization table. However, this kind of compression technique usually results in severe visual artifacts such as blockiness, blurring and ringing distortions due to coarse quantization, and do not achieve optimal performance especially under low bit-rate condition.

In the last few years, several image compression algorithms based on deep-learning have been proposed. These algorithms use a completely different framework from those used in conventional coding, and explore the problem from an entirely different perspective. Basically, a learning-based image compression algorithm is based on a neural network architecture called autoencoder. An autoencoder typically consists of two parts: an encoder that transforms the input image into a lower dimensional space, which is referred to as latent representation, and a decoder that reconstructs the image from this latent representation. These learning-based compression schemes have shown promising performance in both visual quality and compression efficiency compared to traditional transform-based coding solutions.

The main task of this project is to characterize and evaluate some of the most promising learning-based image compression schemes that are trained end-to-end with a large amount of image dataset. Moreover, we will test the performance of some of the recent learning-based image compression algorithms, and analyze the performance by plotting the rate-distortion plots. Finally, we extract the latent space from one deep-learning based image compression method and explore how changes to particular elements in the latent space vector can affect the perceptual visual quality of the reconstructed images.

# 2 Review of Recent Work

In recent years, a number of learning-based image compression schemes have been developed and achieved excellent performance compared to conventional codecs. This section will review some selected end-to-end image/video compression algorithms and give a summary of each solution.

## 2.1 Lossy Image Compression with Compressive Autoencoders[11]:

Neural networks have already achieved state-of-art results in lossless image compression. However, it has yet to surpass existing codecs in lossy image compression due to the inherent non-differentiability of the compression loss. This paper proposes a simple but effective approach for dealing with the non-differentiability of rounding-based quantization, and for approximating the non-differentiable cost of coding the generated coefficients.

It aims at directly optimizing the rate-distortion tradeoff produced by an autoencoder. The idea is to replace the derivative of rounding function in the backward pass of back-propagation with the derivative of a smooth approximation, and uses a continuous and differentiable approximation to estimate entropy rate. This proposed method achieves better performance than JPEG 2000 in terms of SSIM and MOS scores.

## 2.2 End-To-End Optimized Image Compression[3]

This paper describes an image compression method consisting of three parts: a nonlinear analysis transformation, a uniform quantizer, and a nonlinear synthesis transformation. This method uses a more flexible transforms built from cascades of linear convolutions and nonlinearities to optimize MSE. It also uses a proxy loss function based on a continuous relaxation of the probability model, replacing the quantization step with additive uniform noise. In addition, it implements an entropy code and report performance using actual bit rates rather than reporting differential or discrete entropy estimates, thus demonstrating the feasibility of our solution as a complete lossy compression method. This proposed method is shown to have improvements in rate–distortion performance over JPEG and JPEG 2000 for most images and bit rates. In addition, the perceptual quality of this method (as estimated with the MS-SSIM index) exhibits substantial improvement across all test images and bit rates.

## 2.3 Variational Image Compression with A Scale Hyperprior[4]:

One fact with entropy coding in image compression is that it tries to approximate the margin distribution of the latent vector representation. This paper proposed an end-to-end variational image compression model that utilizes the side information, which is usually used in traditional codecs, as a prior on the parameters of the entropy model. This prior can be viewed as hyperpriors of the latent representation. The model therefore incorporates this hyperprior to effectively capture spatial dependencies in the latent representation. Unlike existing autoencoder compression methods, this model trains a complex prior jointly with the underlying autoencoder, which means it essentially learns a latent representation of the entropy model, in the same way that the underlying compression model learns a representation of the image. The experiment results show that the proposed method leads to state-of-the-art image compression when measuring visual quality using the popular MS-SSIM index, and yields rate–distortion performance surpassing published ANN-based methods when evaluated using a more traditional metric based on squared error (PSNR).

## 2.4 Joint Autoregressive and Hierarchical Priors for Learned Image Compression[8]:

The GSM-based model proposed by Ballé uses a noise-based relaxation to be able to apply gradient descent methods to the loss function and introduces a hierarchical prior to improve the entropy model. Ballé uses a Gaussian scale mixture (GSM) where the scale parameters are conditioned on a hyperprior. This paper extends this GSM-based model in two ways. It generalizes the hierarchical GSM model to a Gaussian mixture model, and adds an autoregressive component, which is complementary to the hyperprior. Compared to Ballé's model, the proposed model provides a net benefit and has a better performance

in terms of rate–distortion. In addition, the complementary property allows the hyperprior to learn to store information needed to reduce the uncertainty in the autoregressive model while avoiding information that can be accurately predicted from context.

## 2.5 Generative Adversarial Networks for Extreme Learned Image Compression[1]:

This paper proposed a GAN-based framework for learned generative compression and use it to build an extreme image compression. It presents two main models: generative compression (GC) and selective generative compression (SC). GC preserves the overall image content while generating structure of different scales such as leaves of trees or windows in the facade of buildings. SC, however, completely generates parts of the image from a semantic label map while preserving user-defined regions with a high degree of detail. The GAN-based framework gives dramatic bitrate savings for low bitrates when evaluated in terms of visual quality in a user study (since traditional evaluation matrices are not suitable for low bitrates compression). Moreover, with available semantic label maps, storage savings can also be achieved when constraining the application domain.

## 2.6 High-Fidelity Generative Image Compression[7]:

This paper combines Generative Adversarial Networks and learned compression to achieve high quality reconstructions that are very close to the input. Specifically, it studies the components of the proposed architecture, including normalization layers, generator and discriminator architectures, training strategies, as well as the loss with respect to perceptual metrics and stability. The user study conducted in this paper shows that this approach is visually preferred to previous methods even when these approaches use more than double the bits. In addition, the proposed method is also evaluated by a set of perceptual metrices such as PSNR and MS-SSIM. The results of rate-distortion plots shows that no existing metric can perfectly predict the full raking of the result in user study, but the metrices FID and KID are useful in exploring architectures and other design choices.

## 2.7 Efficient Nonlinear Transforms for Lossy Image Compression[2]:

The Sadam and GDN techniques have been successfully implemented in state-of-the-art image compression methods. In this paper, the author assesses the performance of these two techniques, and provide a detailed comparison of their performance with other popular training algorithms and nonlinearities. In the experiment the author uses an assessment metric PSNR to compare the performance of Samdam with Adam, and GDN with other nonlinearities such as ReLU, softplus and so on. The results show that Sadam can stabilize the training process for nonlinear image transforms, and GDN can increase the approximation capacity and efficiency of the transforms compared to other popular nonlinearities. Future work can be done by exploring whether the efficiency of GDN in terms of number of filters can also bring benefit to reducing computational cost. These algorithms have all shown promising performance when evaluated by objective assessment matrices. We will select four algorithms from above and compare their performance with one of the most used codecs JPEG in the next section.

# 3 Rate Distortion Plots

We select some learning-based image compression models that have achieved promising performance. We first introduce the models to be test and show the experiment results.

## 3.1 Selected Models

- F. Mentzer, G. Toderici, M. Tschannen, E. Agustsson: "High-Fidelity Generative Image Compression"
  - $HiFiC^{Lo}$
  - $HiFiC^{Mi}$
  - $HiFiC^{Hi}$

  These are the GAN-based models that were used for the paper. The target rate (a hyperparameter introduced by the paper) of these models are 0.14 bpp for "lo", 0.3 bpp for "mi", and 0.45 bpp for "hi".

- D. Minnen, J. Ballé, G.D. Toderici: "Joint Autoregressive and Hierarchical Priors for Learned Image Compression"
  - mbt2018-mean-mse-[1-8]
  - mbt2018-mean-msssim-[1-8]

  These are hyperprior models with non-zero mean Gaussian conditionals (without autoregression), optimized for MSE (mean squared error) and MS-SSIM (multiscale SSIM), respectively. The number 1-8 at the end indicates the quality level (1: lowest, 8: highest).

  While generalizing the hyperprior models to non-zero mean distributions translates into some compression gains, the gains are not as high as for a combined autoregressive and hierarchical prior. However, the runtime for autoregressive priors is generally quite high.

- J. Ballé, D. Minnen, S. Singh, S.J. Hwang, N. Johnston: "Variational Image Compression with a Scale Hyperprior"
  - bmshj2018-factorized-mse-[1-8]
  - bmshj2018-factorized-msssim-[1-8]
  - bmshj2018-hyperprior-mse-[1-8]
  - bmshj2018-hyperprior-msssim-[1-8]

  These are the factorized prior and hyperprior models optimized for MSE (mean squared error) and MS-SSIM (multiscale SSIM), respectively. The number 1-8 at the end indicates the quality level (1: lowest, 8: highest).

  These models demonstrate the bit rate savings achieved by a hierarchical vs. a factorized prior (entropy model).

- J. Ballé: "Efficient Nonlinear Transforms for Lossy Image Compression"

- b2018-leaky-relu-128-[1-4]
- b2018-leaky-relu-192-[1-4]
- b2018-gdn-128-[1-4]
- b2018-gdn-192-[1-4]

These are nonlinear transform coders with factorized priors (entropy models) optimized for MSE (mean squared error), with either leaky ReLU and GDN activation functions, and 128 or 192 filters per layer. The number 1-4 at the end indicates the quality level (1: lowest, 4: highest).

These models demonstrate the higher representational efficiency of GDN vs. scalar activation functions such as leaky ReLU.

## 3.2 Experiment Results

We test the models mentioned above using four images from Kodak dateset. The first three images are of size 768x512; the last image is of size 512x768. The original images are shown below:


Figure 1: Original Images

The performance of the models will be evaluated on four objective assessment matrices: PSNR, SSIM, MS-SIM and VIFP. PSNR is the peak signal-to-noise ratio which expresses the ration between the maximum squared value and the MSE of the image. SSIM (Structural Similarity Index Measure) takes into account three measurements, i.e. the luminance, contrast and structure of the image. MS-SSIM (Multi Scale - Structural Similarity Index Measure) extends SSIM by computing it at multiple scales, and aims at simulating the differences in the perceived quality. VIF (Visual Information Fidelity) is based on natural scene statistics and the notion of image information extracted by the human visual system.

Each model is evaluated over different quality levels. In addition, we also add the result of JPEG compression technique to futher compare the learning-based algorithms with the traditional codec. The resulting rate-distortion plot for the four images are shown below:

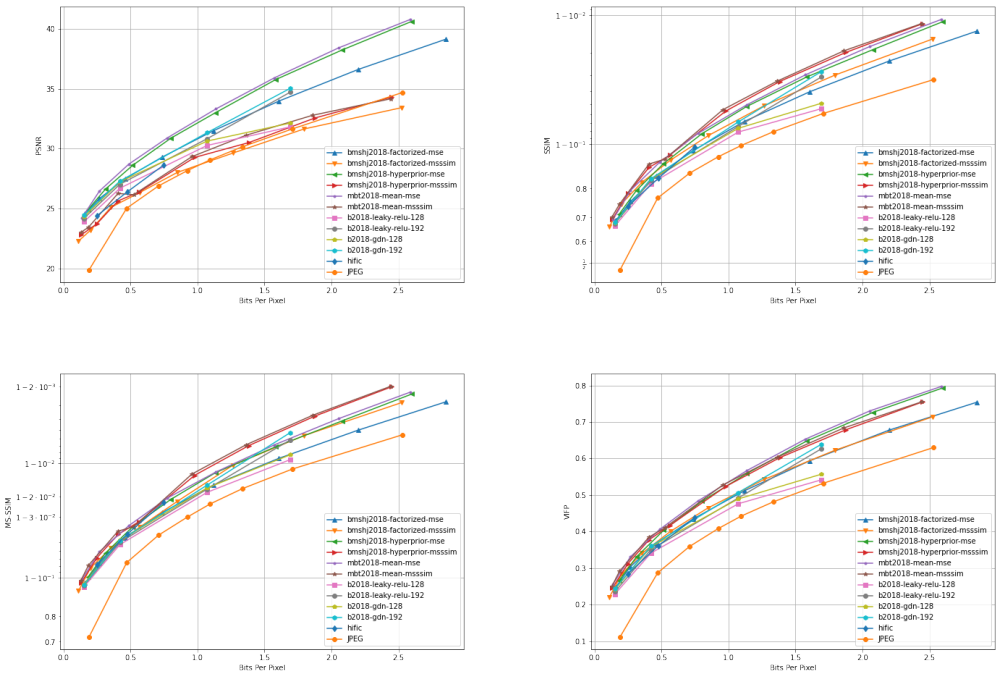


Figure 2: Results of Kodim01

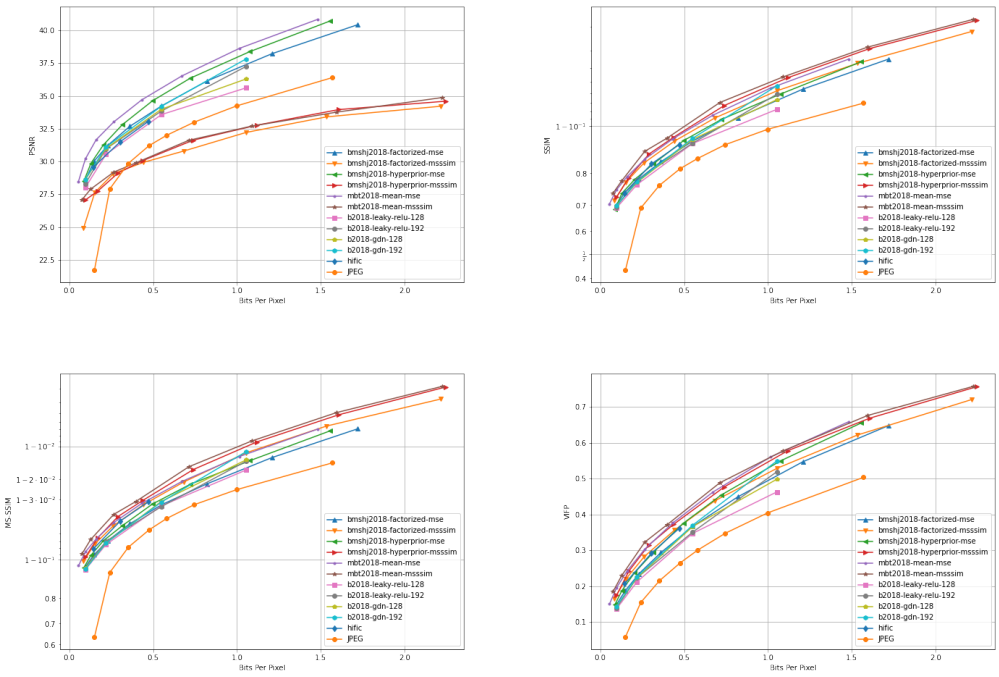


Figure 3: Results of Kodim02

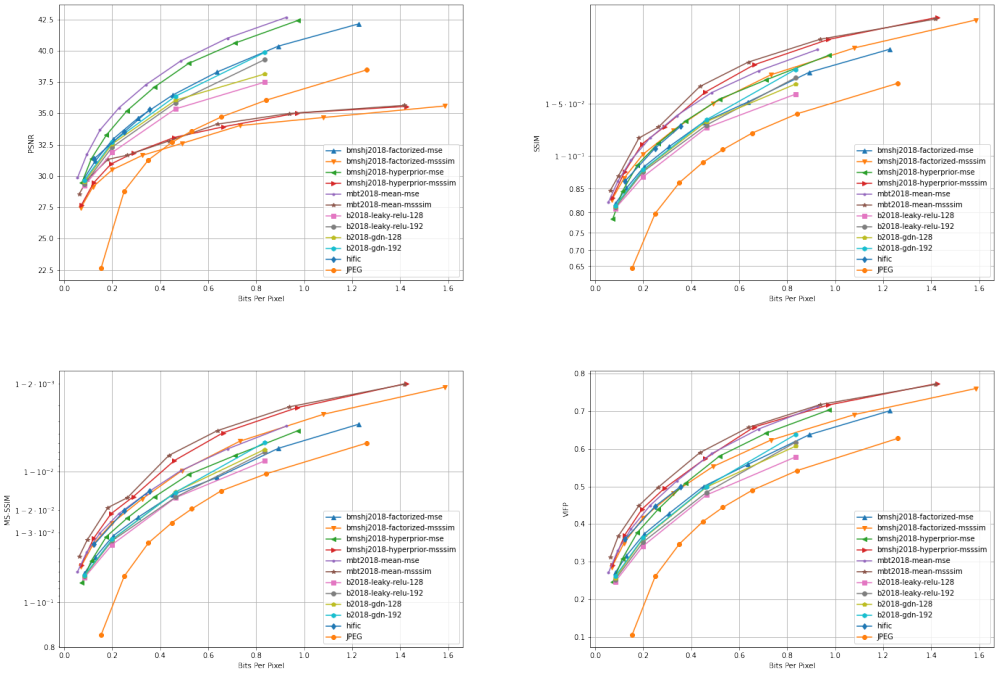


Figure 4: Results of Kodim03

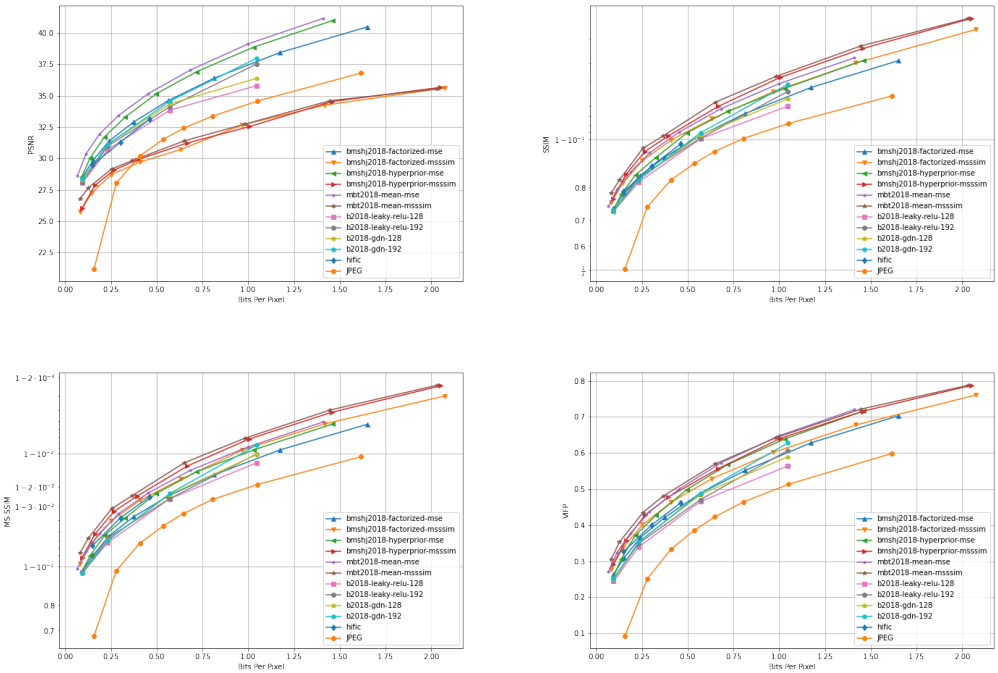


Figure 5: Results of Kodim04

### 3.3 Results Analysis

From the above experiment results we have the following observations. Firstly, it is not surprising that the results highly depends on which distortion metric is used to optimize during training process. For PSNR distortion measurement, all three models that are optimized for MS-SSIM behave poorly, and even surpassed by JPEG in higher bit compression. However, models optimized for MSE show an excellent performance when measured on PSNR, where the result is the inverse when measured on SSIM and MS-SSIM. In addition, the improved model proposed in paper "Variational Image Compression with A Scale Hyperprior" with a hyperprior achieve better performance when compared with the model with a factorized prior. In terms of the use of different nonlinearities, the performance of GDN shows a superiority when compared with ReLU function, as presented in the paper "Efficient Nonlinear Transforms for Lossy Image Compression". If we observe the performance of the model HiFiC, it does not show any superiority in any metrics, which evidence the statement in the paper "High-Fidelity GenerativeImage Compression" that those metrices are not consistent with the result of user study. Finally, in general, the learning-based algorithms outperform the traditional codec JPEG to some extend in terms of all four metrics.

## 4 Latent Space Interpretation

In this section we introduce the concept of latent space. Image compression can be viewed as the process of encoding image using fewer bits than the original representation. The learning-based algorithm is based on the autoencoder architecture, which first converts the image from a high dimensional input to a bottleneck layer, where the number of neurons is the smallest, and then converts it back to the original input shape. The latent space is the space in which the data lies in the bottleneck layer.

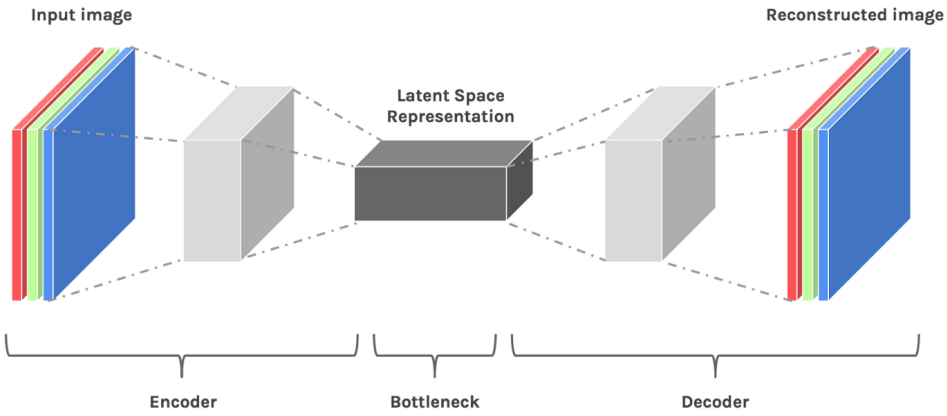


Figure 6: Encoder-Decoder Architecture

Although the latent space is hidden from most, there are certain tasks in which understanding the latent space is not only helpful, but necessary. So far, the latent space representation has been explored in the field of image generation, since the modification of latent vector can bring significant change in the output. This has been further studied in the Generative adversarial networks (GAN), which aims at mapping points in the latent space to generated images.

The latent space itself has no meaning. It is usually a 100-dimensional vector with each variable drawn from a Gaussian distribution with a mean of zero and a standard deviation of one. In the training process, the generator maps points into the latent space, and tries to generate target images. At each step, this mapping would be modified as the model being trained. The latent space therefore has meaningful structure that can be explored when interpreted by the generator model. For example, by interpolating between points and performing vector arithmetic between points in latent space, one can obtain meaningful and targeted effects on the generated images.

Below is an illustration of how changes in the latent space bring different output images using GAN. Typically, new images are generated using random points in the latent space. A series of points can be created on a linear path between two points in the latent space, such as two generated images. These points can be used to generate a series of images that show a transition between the two generated images.

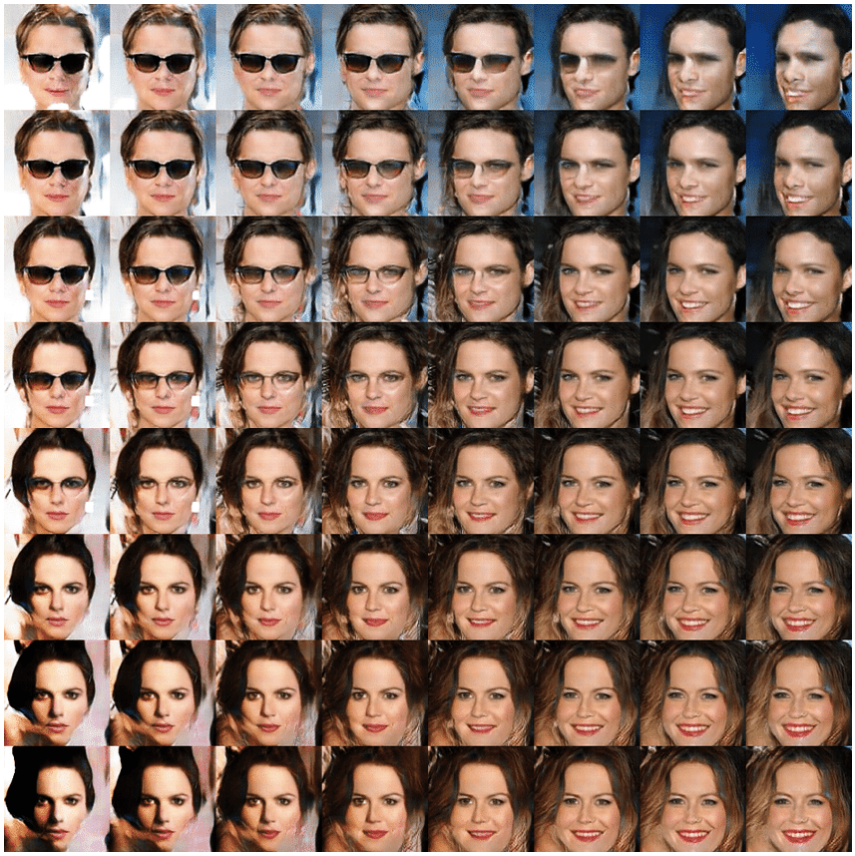

Figure 7: Bilinear interpolation on latent space for random noise vectors

We can observe that the generative models have the ability to perform interpolations for real samples along any arbitrary axis to generate new images. To this aspect, one can for example use deep generative models to manipulate images of human faces along axes like age, gender, hair color and etc. The interpolation works by performing simple linear algebra in the latent space learned by the generative model. More specifically, one can first find an axis in the latent space to interpolate along with, which can be the color of the hair. The interpolation vector for the hair color can then be simply computed as the vector pointing from the centroid of one color to the centroid of the other color in the latent space. Figure 8 shows a real implementation of this kind of technique, where one

can freely change different features of the human face along different axis.

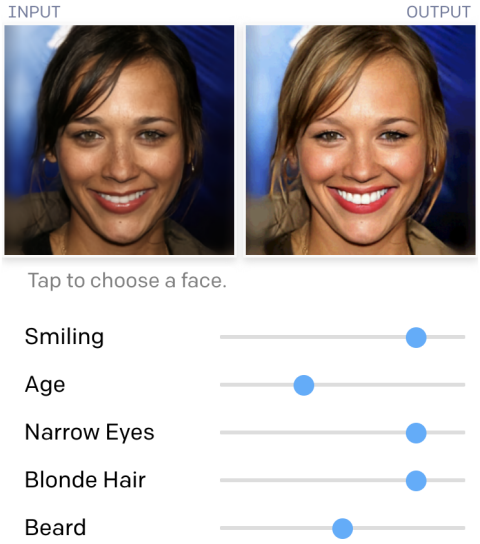


Figure 8: Manipulation of attributes of human face

The study of latent space is therefore meaningful for generative models to produce imaginary images. However, it has not been widely studied in image compression since the situation changes significantly when we bring the notation of latent space to compression field. The most important difference is that in image generation, each dimension in latent space carried specific meaning, with which one can modify and generate meaningful output. However, this is not the case for image compression. The neural network would in general take the input image as a whole and compress it in a way that might not be easily interpretable (which can be called as a black box).

To practically explore the characteristics of latent space in image compression, we extract a latent space representation from one model and perform modifications on it. The model choosed here is from the paper "Variational Image Compression with a Scale Hyperprior", with a hyperprior and optimized for MSE. We set the bitrate level to 6 and compare the results by observing the changes in PSNR, MS-SSIM, bitrate as well as visual perceptual quality. The test image is downloaded from the internet, with size of 768x512, and is shown below:

Figure 9: Test image - Stmalo fracape

We extract the latent representation from the model, which is a 192 by 32 by 48 array. Below is a visualization of part of the dimentions of the latent space:

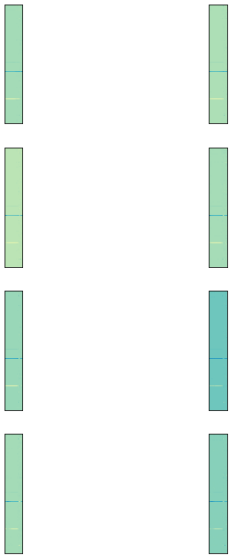

Figure 10: Latent space vector

We then do modifications on the latent space. Figure 11 shows the results of the original implementation. By performing multiple experiments, we observe that the second and third dimension are directly related to the size of the image, i.e. they are proportional to the image size. To show this, we set half of the latent vector to zero and obtain the following result:


Figure 11: Result of the original implementation


Figure 12: Result of setting half of the values in second dimension to zero

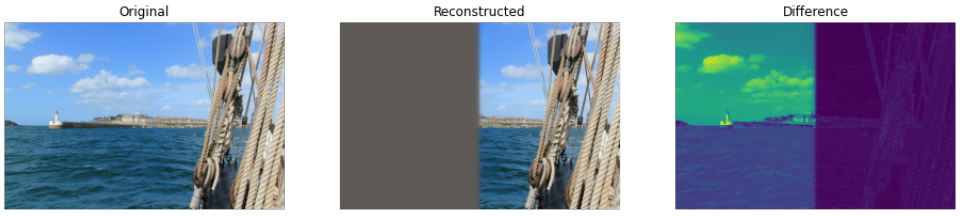


Figure 13: Result of setting half of the values in third dimension to zero

Next, we modify the first dimension. Since we barely know anything about the first dimension, we modify it randomly, by setting the first half, the middle half, and the second half of the values in the first dimension to zero, as well as multiplying the whole latent vector by a factor of 100. The results are shown in figure 14 to figure 17 respectively.


Figure 14: Result of setting the first half of the values in second dimension to zero


Figure 15: Result of setting the middle half of the values in second dimension to zero

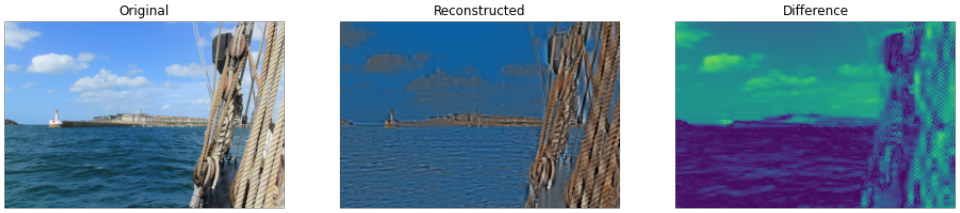


Figure 16: Result of setting the second half of the values in second dimension to zero

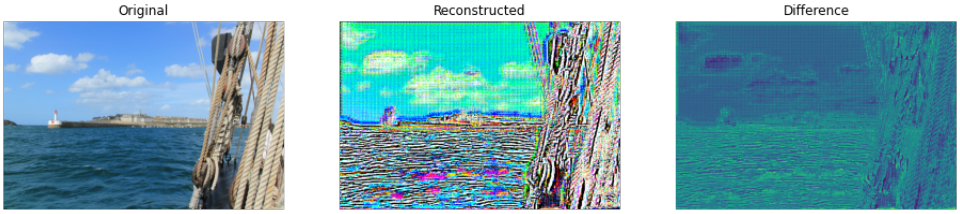


Figure 17: Result of multiplying the latent vector by a factor of 100

The table below shows compares PSNR, MS-SSIM and BPP of all the implementations:

| Figure | PSNR | MS-SSIM | BPP |
|---|---|---|---|
| 11 | 31.42 | 0.9803 | 0.498 |
| 12 | 0.498 | 0.7024 | 0.342 |
| 13 | 12.83 | 0.7145 | 0.366 |
| 14 | 15.89 | 0.7301 | 0.414 |
| 15 | 0.414 | 0.8827 | 0.449 |
| 16 | 10.99 | 0.6327 | 0.419 |
| 17 | 6.61 | 0.2760 | 5.231 |

Table 1: Comparison of all the implementations

After doing multiple modifications, we observe that the PSNR and MS-SSIM decrease as we change the vector values, and the bitrate instead increases as compared to the result generated from original latent representation. Besides, unlike image generation that based on generative models, where each vector can represent a certain feature, the latent vector in image compression field do not represent any specific component of the image. The algorithm however takes the input image as a whole and compress it. Modifying the latent space do not bring the same type of changing or improvement to the output as in image generation. Future work can be done by implementing linear or nonlinear transformation to the latent space, or even using machine learning algorithms to explore the insight of the latent space.

# 5 Conclusion

In this project, we summarize the state-of-art learning-based image compression technique, and analyzed their performance by conducting experiments and generating rate-distortion plots using four objective assessment metrics: PSNR, SSIM, MS-SSIM and VIFP. In addition, we study the state of art of the latent space interpretation in deep learning, in particular in generative models. The modifications on the latent space of different axis can change the attributes of the output, and thus generate new images. We bring this notion to image compression field and extract the latent vector from one selected model and do modifications on it. The resulting images differ hugely from the original implementation, regardless of visual perception or objective assessment metric. Future work can be conducted by using different transformations or algorithms to explore meaningful changes in the latent space.

# References


[1] E. Agustsson, M. Tschannen, F. Mentzer, R. Timofte, and L. Gool. Generative adversarial networks for extreme learned image compression. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 221–231, 2019.

[2] J. Ballé. Efficient nonlinear transforms for lossy image compression. *2018 Picture Coding Symposium (PCS)*, pages 248–252, 2018.

[3] J. Ballé, V. Laparra, and E. P. Simoncelli. End-to-end optimized image compression. *ArXiv*, abs/1611.01704, 2017.

[4] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston. Variational image compression with a scale hyperprior. *ArXiv*, abs/1802.01436, 2018.

[5] F. P. T. E. Joao Ascenso, Pinar Akyazi. Learning-based image coding: early solutions reviewing and subjective quality evaluation. *SPIE Photonics Europe*, 2020.

[6] D. P. Kingma and P. Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *ArXiv*, abs/1807.03039, 2018.

[7] F. Mentzer, G. Toderici, M. Tschannen, and E. Agustsson. High-fidelity generative image compression. *ArXiv*, abs/2006.09965, 2020.

[8] D. Minnen, J. Ballé, and G. Toderici. Joint autoregressive and hierarchical priors for learned image compression. *ArXiv*, abs/1809.02736, 2018.

[9] M. Pieters and M. Wiering. Comparing generative adversarial network techniques for image creation and modification. *ArXiv*, abs/1803.09093, 2018.

[10] W. Tao, F. Jiang, S. Zhang, J. Ren, W. Shi, W. Zuo, X. Guo, and D. Zhao. An end-to-end compression framework based on convolutional neural networks. In *2017 Data Compression Conference (DCC)*, pages 463–463, 2017.

[11] L. Theis, W. Shi, A. Cunningham, and F. Huszár. Lossy image compression with compressive autoencoders. *ArXiv*, abs/1703.00395, 2017.

[12] G. Toderici, D. Vincent, N. Johnston, S. J. Hwang, D. Minnen, J. Shor, and M. Covell. Full resolution image compression with recurrent neural networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5435–5443, 2017.