A Study on the Effects of Latent Space Compression on a Pretrained Classifier for Prostate Biopsies

Guilherme Junqueira Perticarari

December 2023

Abstract

The goal of this study is to understand the trade-off between size of the latent space representation of a patch from a whole-slide image from a prostate biopsy and the quality of a pretrained classifier on its reconstruction. This is done with a special kind of autoencoder, wherein the output of every convolution block can be treated as its own latent space. Using this autoencoder, I find that reconstructions from encodings of up to 12.5% the size of the original image yield a drop of only 3.33% in the classifier accuracy. The study also shows that the precision of the model increases with smaller encodings, while the recall presents a sharp decline, which indicates that for poorer reconstructions, the classifier loses the ability to discriminate between more nuanced images.

1 Introduction

Modern medical images can present enourmous sizes, with whole-slide images (WSI) taking gigabites of space for one ultra-resolution picture [Litjens et al., 2017]. For biopsies, images like this provide a lot of information about the tissues depicted in them, which present an opportunity for deep learning algorithms to learn to automate complex tasks [Litjens et al., 2017] [Çelik and Karabatak, 2021]. Nevertheless, such a big size becomes a challenge for the hardware needed to store and process these images, which can drive up the costs for the entire process quite significantly [Çelik and Karabatak, 2021]. A possible solution to this problem would be to use autoencoders to encode the original image to a much smaller size, which could then be used to reconstruct the image for other kinds of deep learning applications, like segmentation or classification [Çelik and Karabatak, 2021].

The goal of this study is to understand the relationship between the size of an autoencoder's latent space representation of an image and the predictive quality of a classifier trained on the original image. For efficiency purposes, this will be done with a special kind of autoencoder that can accommodate latent spaces of different sizes.

2 Data and Methodology

2.1 Data

The dataset used in this study consists of $256 \times 256 \times 3$ RGB patches of WSIs of prostate biopsies collected from patients at the Skåne University Hospital, Malmö (Sweden) between 2014 and 2018. This is the same data used in [Marginean et al., 2021], and some images are shown in Figure 1.

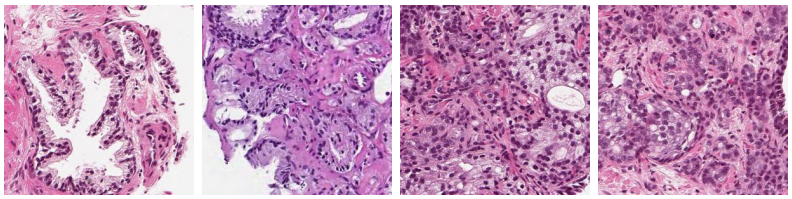


Figure 1: Examples of slide images from the dataset.

These slides are accompanied by their labels, which contain expert information annotated by Drs. Felicia Marginean and Athanasios Simoulis. These labels are either a Gleason Grading Score, from 3 to 5, or benign, if there are no signs of malignant cancer. In this study, a classifier will be trained to predict if the label is benign or not. These images have already been split in two cohorts: a training cohort with 327,999 images and a test cohort with 12,772 images upon the dataset creation. This exact split will be maintained for the training and validation of both the autoencoder and the model that will be described in subsequent sections.

2.2 Autoencoder

Autoencoders are artificial neural networks that are built to reconstruct their inputs after passing through a narrow bottleneck layer, whose values are referred to as 'latent space representations' of the inputs.

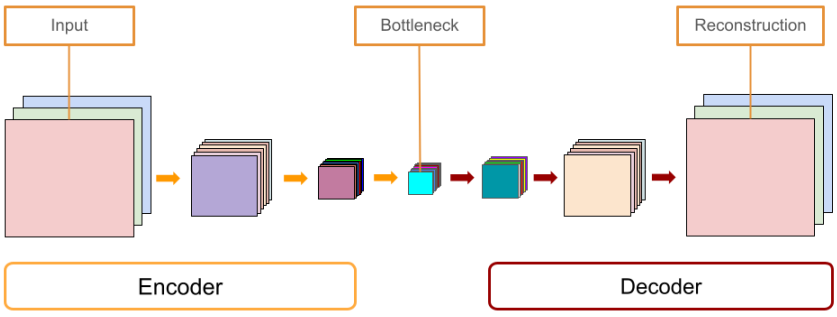


Figure 2: Example of an autoencoder.

This structure, shown in Figure 2, can be broken down in two parts: an encoder, which compresses the input to this bottleneck layer, and a decoder, which reconstructs the original input from this bottleneck layer. The loss function is some similarity metric between the original input and its reconstruction.

2.2.1 Architecture for Adjustable Latent Spaces

In this study, an autoencoder will be used to reconstruct the RGB patches of prostate biopsies WSIs described in section 2.1. However, unlike the configuration shown in Figure 2, this autoencoder will not display the usual single-encoder single-decoder structure. It was built so that each layer after a convolution block could be interpreted as its own latent space representation of the image. The number of pixels in these representations decrease exponentially with a factor of two, as will be explained below. With that, the user can choose the size of the representation they want to decode.

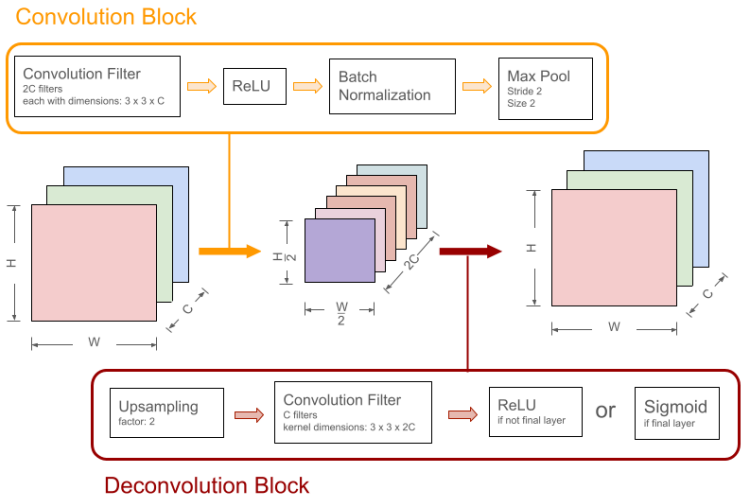


Figure 3: Convolution and Deconvolution blocks used for the encoder and the decoder, respectively.

The autoencoder uses convolution blocks (Figure 3) that double the number of channels and halve the height and width of the incoming layer. So, they effectively halve the number of pixels from the incoming layer. If its input has dimensions $H \times W \times C$, the block will consist of:

- $2C$ convolution filters with kernel size $3 \times 3 \times C$ with same padding
- ReLU activation function
- Batch Normalization
- Max Pooling layer with size 2 and stride 2

The deconvolution blocks (Figure 3) double the number of channels and halve the height and width of the incoming layer. So, they effectively double the number of pixels from the incoming layer. If its input has dimensions $H \times W \times C$, the block will consist of:

- Upsampling layer with factor 2
- $\frac{C}{2}$ convolution filters with kernel size and same padding
- Sigmoid activation function for the final layer and ReLU otherwise. This forces the reconstruction's pixel values to fall between 0 and 1.

The number of convolution blocks define the size of this autoencoder. For an autoencoder with size equal to N, there will be N convolution blocks, and for each block j, with $j = 1, ..., N$, there will be a unique associated decoder with exactly j deconvolution blocks. Figures 4, 5 and 6 depict this architecture.

2.2.2 Training

Training the network described above is done sequentially for each convolution block and its associated decoder, with previously trained convolution blocks of the encoders being kept frozen. This is done because, after they have been sufficiently optimized, these blocks should be able to extract a useful lower-dimensional representation of the image. Thus, they provide some efficiency gains relative to training additional autoencoders from scratch. The examples below will describe this process with a network of size 3. The autoencoder used in this study had size 4.

For the first encoding, the encoder comprises the first convolution block and its decoder comprises a single deconvolution block, as shown in Figure 4.

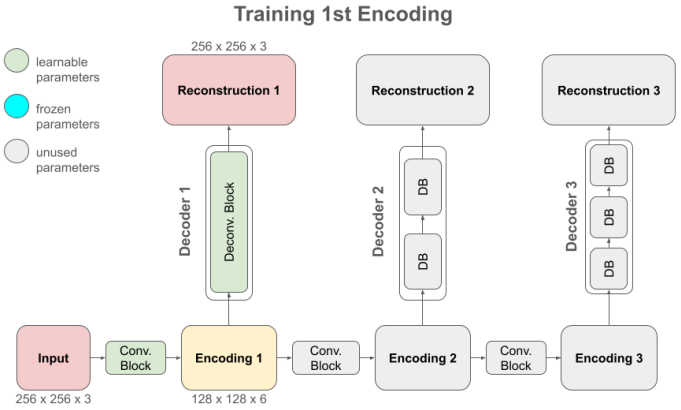


Figure 4: When training the first encoding, only the first convolution block and its decoder are optimized.

For the second encoding, the encoder comprises the first two convolution blocks and its decoder comprises two deconvolution blocks, as shown in Figure 5. The parameters of the first convolution block are used in the calculation, but its values are kept unchanged, since they have already been optimized.

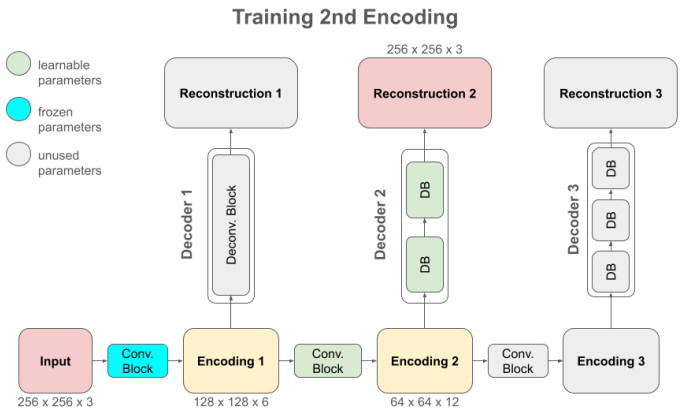


Figure 5: When training the second encoding, only the second convolution block and its decoder are optimized.

For the third encoding, the encoder comprises the first three convolution blocks and its decoder comprises three deconvolution blocks, as shown in Figure 6. The parameters of the first two convolution blocks are used in the calculation, but their values are kept unchanged, since they have already been optimized.

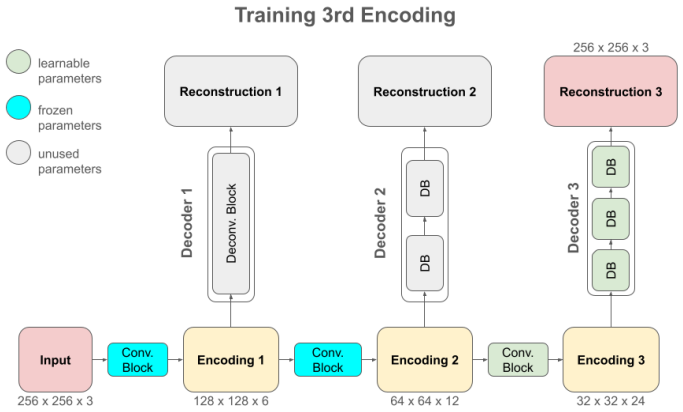


Figure 6: When training the third encoding, only the third convolution block and its decoder are optimized.

The loss function is defined in Equation 1, with MSE and MAE being the Mean-Squared Error and the Mean-Absolute Error, respectively, for all pixels. SSIMLoss is simply 1 - SSIM, where SSIM is the Structure Similarity Index Measure. This choice was motivated by [Roy et al., 2021].

$$loss = 0.5 \times MSE + 0.5 \times MAE + 0.5 \times SSIMLoss \tag{1}$$

2.2.3 Usage

After training the whole network, it can be used by simply stating how much latent space compression is required for the image's reconstruction. This argument will dictate how far down the convolution blocks to go for a latent space representation before using the last convolution block's associated decoder for the reconstruction.

2.3 Classifier

For the benign v. malignant classifier, a simple neural network with convolution blocks and dense layers was built with a sigmoid layer for the output layer and binary cross entropy for a loss function. The exact architecture is shown below on Figure 7.

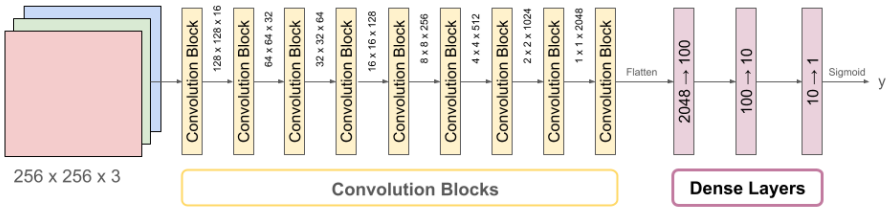


Figure 7: Classifier structure.

All convolution blocks halve the height and width of its incoming layer. The number of channels, on the other hand, initially goes from 3 (RGB) to 16 and, after that, is doubled until it reaches 2048 at the last convolution block. This is done with 3x3 kernels in the convolution layer. Between the convolution layer and the Max Pool layer, there is a ReLU layer followed by a Batch Normalization layer, following the same sequence of operations of the convolution blocks from the autoencoder, described in Figure 3.

The model was trained on the "train" cohort and validated on the "test" cohort, as explained in section 2.1. Moreover, the training was done on the original images of the train cohort only, but the validation was done first on the original test images, and then on the reconstructions of each encoding, as shown in Figure 10.

3 Results

The autoencoder's ability to reconstruct the test set images was evaluated for each possible encoding. The performance was measured in terms of loss, as defined in Equation 1, and Structural Similarity Index Measure (SSIM). The results (Figure 8) clearly show a drop in similarity with reconstructions from encodings of higher compression factors. The drop in performance can also be noticed by comparing a sample with its reconstructions in Figure 9.

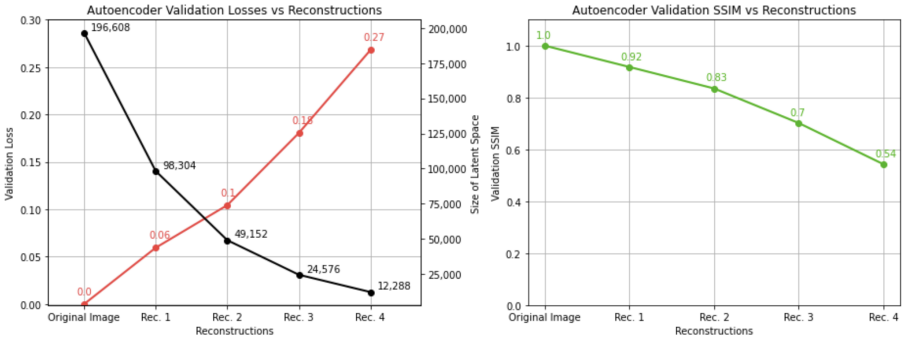
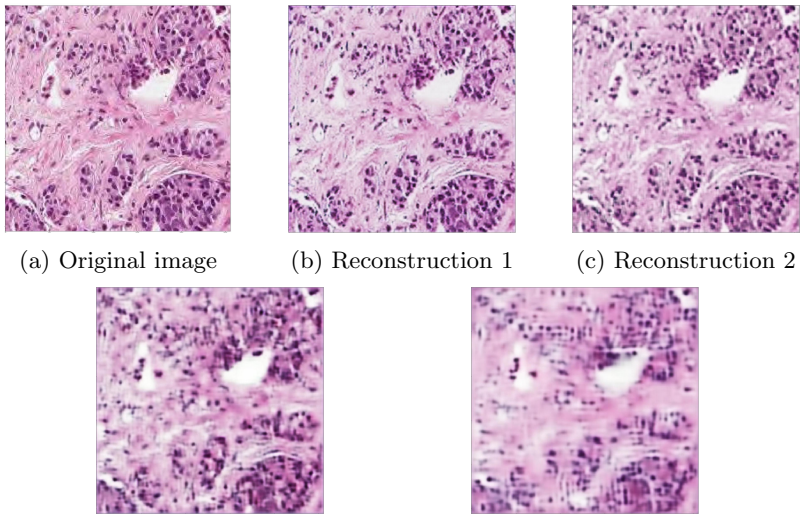


Figure 8: The autoencoder's validation performance for the reconstructions. The black line shows the size of the latent space representation, the red line shows the autoencoder's loss and the green line shows the SSIM between each reconstruction and the original image.



(a) Original image (b) Reconstruction 1 (c) Reconstruction 2

(d) Reconstruction 3 (e) Reconstruction 4

Figure 9: Example of an original test image alongside its reconstructions

The main result, though, was the comparison between the classifier's performance on the original test images and on the images' reconstruction using the autoencoder. A depiction of how this will be done is shown in Figure 10. This comparison, as measured by Binary Cross-Entropy loss, the accuracy, the precision and the recall, is shown in Figure 11, alongside the size of the latent space for each reconstruction.

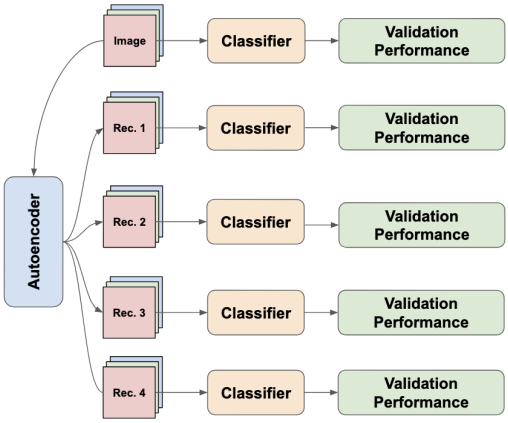


Figure 10: Depiction of the validation process of the classifier on the test dataset.

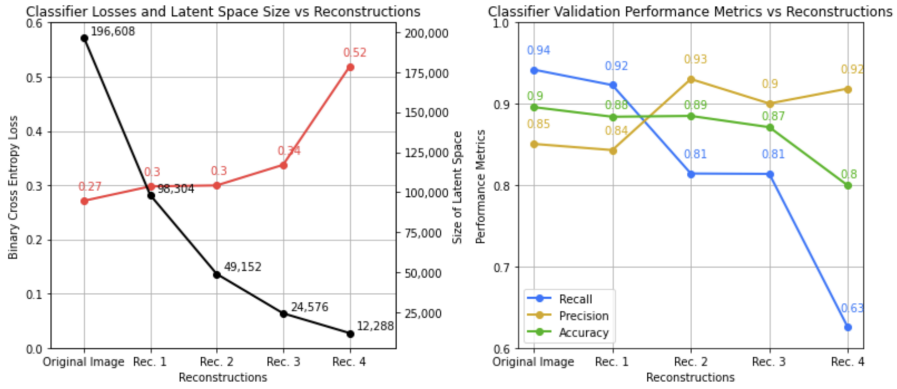


Figure 11: Classifier performance metrics and latent space size relative to each reconstruction. The black line shows the latent space size, the red line shows the classifier loss. In the following plot, the classifier's accuracy, precision and recall are shown in green, yellow and blue, respectively. The threshold for the accuracy, recall and precision are taken to be 50%.

The graphs in Figure 11 quantify the trade-off between the autoencoder's compression and the quality of a pretrained classifier, which was the goal of this study. Some of the key takeaways from this result are:

1. the autoencoder loss, as well as the SSIM, degrade almost linearly for every reconstruction. This decrease is much more dramatic than the one we see for the classifier.

2. the classifier's accuracy degrades slightly for the first 3 reconstructions, going from 90% in the original image to 87% (3.33% decrease). At the same time, the size of the latent space falls from 196,608 (original image - no compression) to 24,576 (reconstruction 3), an 87.5% reduction.

3. the precision and recall curves seem to indicate that as the latent space shrinks, only the most obvious images are classified correctly, which leads to a high precision (confidence in the model's output) and low recall (true positive rate).

4 Conclusion and Discussion

This study showed that there is a clear trade-off between the latent space dimension and a classifier's predictive quality when applied to the image's reconstruction from the autoencoder, as described in section 2 for the task of benign v. malignant prediction from patches of prostate biopsy WSIs. However, the classifier's loss seem to increase much less than the efficiency gain in shrinking the encoding of an image with the autoencoder reconstructions 1, 2 and 3 - with compressions of 50%, 25% and 12.5% respectively. Therefore, as long as the task could do with a slightly worse performance, a user could use an autoencoder to save an encoding of a small enough size and have a more efficient operation. These gains could be impressive when dealing with WSIs, which typically present $100,000 \times 100,000$ pixels [Wang et al., 2012].

Although this study successfully answers the questions raised at its conception, it left many interesting aspects unexplored. For instance, one could test different autoencoder structures, allowing a smoother decrease in the size of the latent spaces. The sharp decline in classifier performance from reconstruction 3 to reconstruction 4 (Figure 11) indicates that perhaps an exponential decay could be good for the first convolution blocks, but it may become too sharp for the later ones. Moreover, one could explore training classifiers using the latent spaces themselves as inputs, so that the decoders could be left out of future use.

References

[Litjens et al., 2017] Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A., van Ginneken, B., and Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88.

[Marginean et al., 2021] Marginean, F., Arvidsson, I., Simoulis, A., Christian Overgaard, N., Åström, K., Heyden, A., Bjartell, A., and Krzyzanowska, A. (2021). An artificial intelligence–based support tool for automation and standardisation of gleason grading in prostate biopsies. *European Urology Focus*, 7(5):995–1001.

[Roy et al., 2021] Roy, M., Kong, J., Kashyap, S., Pastore, V. P., Wang, F., Wong, K. C. L., and Mukherjee, V. (2021). Convolutional autoencoder based model histocae for segmentation of viable tumor regions in liver whole-slide images. *Scientific Reports*, 11(1):139.

[Wang et al., 2012] Wang, F., Oh, T. W., Vergara-Niedermayr, C., Kurc, T., and Saltz, J. (2012). Managing and querying whole slide images. *Proc SPIE Int Soc Opt Eng*, 8319.

[Çelik and Karabatak, 2021] Çelik, Y. and Karabatak, M. (2021). Extracting low dimensional representations from large size whole slide images using deep convolutional autoencoders. *Expert Systems*, 40.