On combining denoising with learning-based image decoding

Léo Larigauderie, Michela Testolina, and Touradj Ebrahimi

Multimedia Signal Processing Group (MMSPG)
Ecole Polytechnique Fédérale de Lausanne (EPFL)
CH-1015 Lausanne, Switzerland
leo.larigauderie@epfl.ch, michela.testolina@epfl.ch, touradj.ebrahimi@epfl.ch

ABSTRACT

Noise is an intrinsic part of any sensor and is present, in various degrees, in any content that has been captured in real life environments. In imaging applications, several pre- and post-processing solutions have been proposed to cope with noise in captured images. More recently, learning-based solutions have shown impressive results in image enhancement in general, and in image denoising in particular. In this paper, we review multiple novel solutions for image denoising in the compressed domain, by integrating denoising operations into the decoder of a learning-based compression method. The paper starts by explaining the advantages of such an approach from different points of view. We then describe the proposed solutions, including both blind and non-blind methods, comparing them to state of the art methods. Finally, conclusions are drawn from the obtained results, summarizing the advantages and drawbacks of each method.

Keywords: Image denoising, learning-based compression, latent space, image processing, deep-learning

1. INTRODUCTION

Capturing images through digital devices such as smartphones, tablets or cameras has recently become a common practice, leading to a growing demand for storage of trillions of pictures per year [*]. The vast amount of stored data motivates the research toward novel and more efficient compression methods, which could allow reducing the enormous needs for storage space. While a number of conventional standards have been proposed in the past, recent research efforts are mostly devoted to learning-based compression methods.[1] As an example, the JPEG Committee has recently organized an activity with the goal of standardizing a novel learning-based compression algorithm, also known as JPEG AI. The group first reported the leading performance of learning-based methods over conventional image compression during a Call for Evidence,[2] and successively compared different emerging technologies in a Call for Proposals.[3]

Recent trends reveal that images are nowadays not only intended for human consumption, but also for computer vision applications. Therefore, compressed contents should not only maximize the perceptual similarity with its original version, but also guarantee good performance for computer vision and image processing tasks. In this context, JPEG AI proposed, in the "Use Cases and Requirements for JPEG AI" document,[4] a framework that allows image processing and computer vision tasks applied directly in the latent space of learning-based image compression, therefore without the need for standard decoding. In particular, the compressed stream should not only allow reconstruction with a standard decoder that specifically targets the human vision, but should also allow computer vision tasks applied to the compressed domain or non-normative decoders for image processing operations like denoising or super-resolution. This framework has the advantage that it does not require preliminary information about the target application, and that the non-normative decoders or computer vision networks can be updated to the most recent technology without the need to transcode or re-capture the content.

Noise is a common disturbance factor in images, which impacts the visual quality of images and the performance of multiple computer vision methods, including face detection and recognition.[5] Noise is often caused by both intrinsic factors, like the camera sensor, and extrinsic factors, like the ambient light, and may be impossible to avoid in many situations. This makes image denoising necessary and desirable, and a classical and

[*] https://blog.mylio.com/how-many-photos-will-be-taken-in-2021-stats/

well-studied problem in the state of the art. Generally, the goal of image denoising is to reconstruct an image $\hat{x}$ from its noisy observation $y = x + n$. The noise n is often approximated in the literature as additive white Gaussian noise (AWGN), which is signal-independent with zero mean and standard deviation σ. Real noise can be more realistically approximated with the Gaussian-Poissonian model,[6] where the noise is approximated by a Poissonian signal-dependent component η_p and a Gaussian signal-independent component η_g.

In this paper, we propose and assess different non-normative decoders able to jointly reconstruct and denoise a compressed stream generated by a learned encoder. Notably, different blind and non-blind solutions are implemented and compared, and the results are assessed using a number of objective quality metrics. Moreover, the benefit of including extra information, e.g. the standard deviation of the noise σ, is discussed. All the proposed solutions allow for improved performance when compared to the anchor methods (including compression and denoising in cascade) in terms of perceptual visual quality and computational complexity.

The remaining of this paper is structured as follows: Section 2 summarizes the state of the art in learning-based image compression, image denoising, and computer vision and image processing methods applied directly to the latent space of image compression. Section 3 reviews the different proposed methods for combined compression and denoising. Results are reported and discussed in Section 4, while conclusions are drawn in Section 5.

2. RELATED WORK

Following the constant growth in the total number of images taken by and stored on digital devices, new and more efficient solutions to image compression are consistently being researched. Recently, a number of image compression solutions based on autoencoders have been investigated,[7–12] reporting high performance in terms of compression efficiency and perceived visual quality.[13] In particular, Ballè et al. firstly proposed an autoencoder solution using nonlinear transforms in cascade to linear convolutions,[7] which was then extended by introducing side information in the form of a hyperprior that captures the spatial dependencies in the latent representation,[8] and includes an autoregressive model to reduce the amount of side information.[9] More recently, generative models have been proposed, synthesizing details of the image to improve the performance at the lowest bitrates first,[11] and successively maximizing perceptual similarity metrics to generate images with improved visual quality.[12]

In conventional scenarios, image compression is followed by either pre- or post-processing operations, with the goal of limiting the distortions introduced by capture, compression and other factors. In this context, image denoising is used as both pre- and post- processing operations. Multiple conventional denoising methods have been proposed in the state of the art. As an example, Wavelet thresholding[14] relies on the wavelet transform to denoise images. More recently, denoising methods based on neural networks[15,16] were able to achieve better performance at the cost of an additional computational cost. Notably, Zhang et al. proposed a denoising solution based on a deep convolutional neural network (CNN), known as DnCNN, trained to estimate the residual noise from a noisy observation,[15] and successively improved the method by integrating a uniform noise level map as input to the network[16] in FFDNet. This additional information enables the network to handle a wide range of noise levels and to compromise between noise reduction and detail preservation. Recently, Guo et al.[17] proposed a learning-based approach combining a noise level estimation network with a non-blind denoising network into a unified blind method known as CBDNet, trained on realistic noise and with emphasis on mitigating noise level under-estimation. Finally, Yue et al.[18] proposed an innovative deep-learning-based bayesian framework for blind image denoising and noise modeling, based on variational inference.

In recent years, due to the large amount of images that are intended for machine consumption, researchers in image compression try to design compression methods able to encode images that are not only visually pleasing after the reconstruction with a conventional decoder, but that also optimize computer vision and image processing tasks.[4,19] A limited number of methods attempted to apply computer vision and image processing methods directly in the latent space of image compression. Early results have been presented by Torfason et al.,[20] which proposed to apply image classification and semantic segmentation in the latent representation of a learning-based image compression method, showing improvements in run-time, memory usage, robustness and synergy, and by compromising only the performance at the lowest bitrates. More recently, super-resolution algorithms have been applied to the latent space of image compression,[21] showing promising results in terms of visual quality. Preliminary work in the compressed domain image denoising field proposed a non-normative decoder

solution able to combine decoding and denoising operations, while reducing the computational complexity of the pipeline.[22] A different approach for latent-space denoising was proposed by Alvar et al.,[23] where a joint compression and denoising network based on a scalable latent space allowed to achieve BD-rate savings and improve the quality of images simultaneously. A joint compression and denoising method designed for satellite images was proposed, by training both the encoder and the decoder of a learning-based compression algorithm with an alternative loss function.[24] Finally, Cheng et al[25] recently proposed a pipeline for joint compression and denoising, with the goal of reducing the storage space by minimizing the allocated bits used to store the noise information. While these last methods have demonstrated improved performance, they are only suitable for a limited number of applications but not all; for instance, reconstructing the original image without denoising is desirable for preserving artistic intent. Focusing the denoising operations at the decoder side allows for a more flexible choice of the desired decoder, without the need of storing multiple versions of the same content.

Regardless, the research on learning-based computer vision and image processing techniques applied to the latent space of image compression is still at an early stage, and more efforts are needed to design robust coding methods which are suitable for both machine and human vision. Notably, the impact of different architectures on the performance of denoising methods applied in the latent space of image compression has not been fully investigated yet.

3. COMBINED COMPRESSION AND DENOISING

In this section, different pipelines for combined decoding and denoising are proposed. Notably, six different methods are presented, including both blind and non-blind methods.For the experiments, the variational autoencoder with a scale hyperprior model,[8] and specifically the CompressAI implementation[26] pre-trained for mean squared error (MSE), is used as a baseline, and non-normative decoders are implemented by using novel training strategies and loss functions. For all the experiments, the encoder is frozen to allow the fine-tuning of the decoder only. The proposed training strategies can be applied to almost any learning-based compression method, and therefore, in the future, it can be tested on upcoming learning-based image compression standards, e.g.the JPEG AI compatible coding. For all the experiments, synthetic Gaussian-Poissonian noise[6] is considered and applied to the images using the practical noise generator designed for the JPEG AI CfP.[27]

3.1 Blind combined decoding and denoising

The first proposed blind solution is able to blindly reconstruct and denoise images simultaneously, i.e. without using the information of the standard deviation of the noise σ, or of the a and b parameters used to generate the Gaussian-Poissonian noise.[6] The loss function is edited by removing the rate computation and optimizing only the distortion, since freezing the encoder allows for the rate to be fixed. The distortion D is computed between the reconstructed and original noise-free images, allowing the decoder to learn denoising in parallel to decoding. Consequently, the proposed loss is the following:

$$\mathcal{L}(x, \widehat{x}_n) = D(x, \widehat{x}_n) \tag{1}$$

Where x is the original noise-free image, and $\widehat{x}_n$ is the reconstructed and denoised image. In this case, both the original and noisy images are available to the network, as synthetic random Poissonian-Gaussian noise is applied to each batch during the training. The distortion metric D is that used in the original compression model, i.e. MSE, in this case. Therefore, the decoder is fine-tuned using the loss function 1, or by computing the distortion metric between the reconstructed and the original noise-free image. The pipeline of the proposed blind combined decoding and denoising method is presented in Figure 1. We denote this model as *blind*.

3.2 Non-blind combined denoising and decoding

Two additional solutions for non-blind combined denoising and decoding strategies are proposed. Notably, the decoder architecture is extended to take as input a noise map concatenated to the image latent, and particularly n input channels and corresponding filters are added to the first convolutional layer of the learning-based decoder, being the noise map composed of n channels. The architecture of the other layers of the decoder remains unchanged. The noise map used as input to the decoder is a lower resolution version of the true point-wise noise

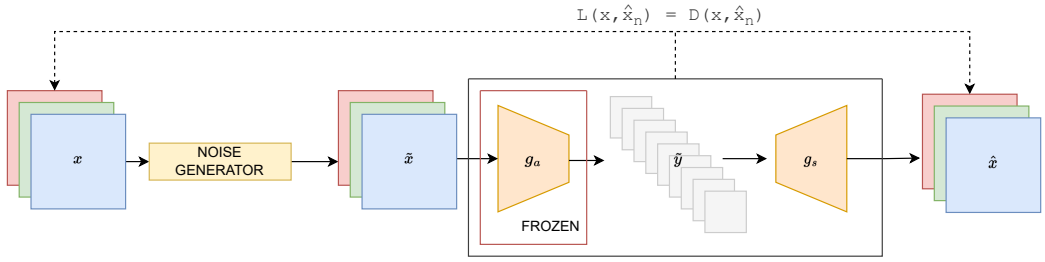


Figure 1: Training pipeline of the proposed *blind* combined decoding and denoising method. Here, x represents the original noise-free image, $\tilde{x}$ the noisy input image, g_a the encoder, g_s the decoder, $\tilde{y}$ the latent presentation, and $\hat{x}$ the reconstructed noise-free image.

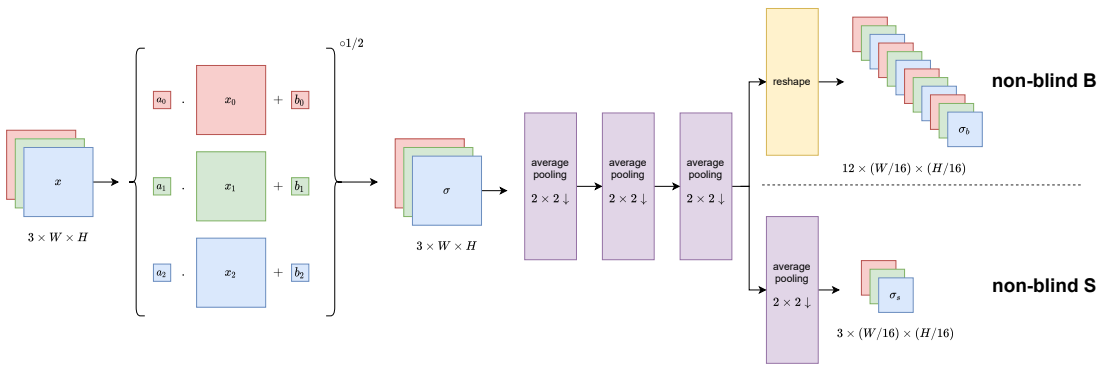


Figure 2: Noise map computation process. The noise map with 12 channels σ_b is used in the *non-blind B* method, the noise map having 3 channels σ_s is used in the *non-blind S* method.

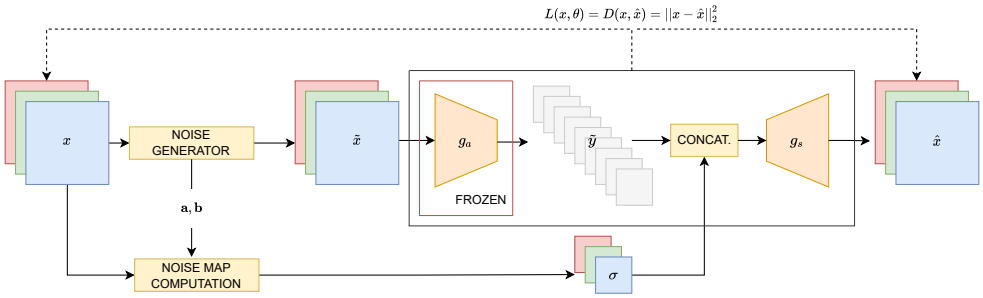


Figure 3: Training pipeline of the proposed non-blind denoising and decoding methods. Here σ refers to the low resolution noise map with either 3 channels σ_s for *non-blind S* or 12 channels σ_b for *non-blind B*.

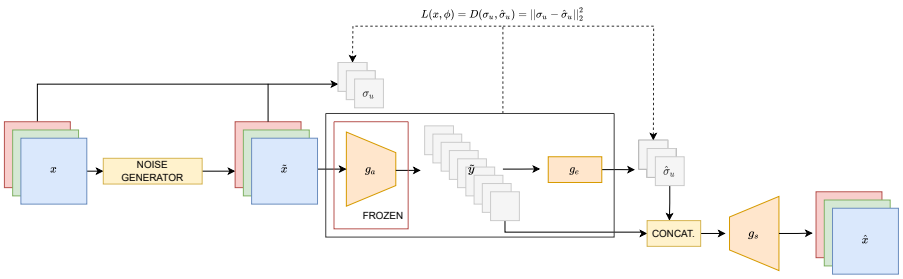


Figure 4: Training pipeline of the proposed *blind E* denoising and decoding method. The architecture is built upon the non-blind method, with g_s referring to the non-blind denoising decoder. g_e refers to the noise level estimation network. The ground truth uniform noise level map σ_u contains the overall noise level of the noisy image $\tilde{x}$, i.e., an estimation of the standard deviation of $\tilde{x} - x$ over all pixels

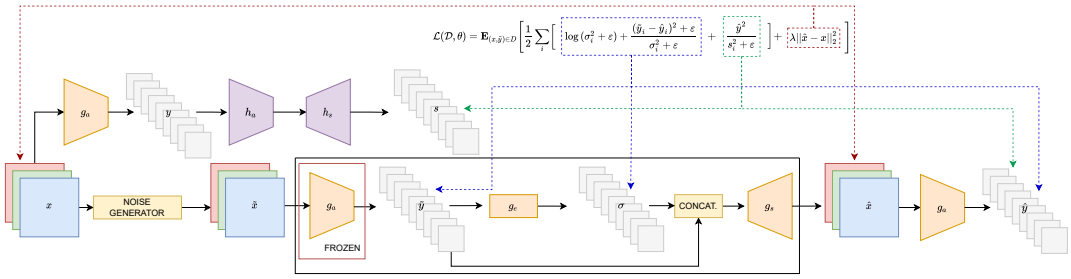


Figure 5: Training pipeline of the *blind L* method.

level map in the original RGB space, given by the parameters of the Poisson-Gaussian model and by the original image. For an original RGB image $\mathbf{x}$, in channel i, at a 2D-position p, the local noise level (standard deviation) $\sigma_i(p)$ is given by:

$$\sigma_i(p) = \sqrt{a_i x_i(p) + b_i} \tag{2}$$

Where a_i and b_i are the parameters of the Poissonian-Gaussian noise[6,27] in channel i.

Equation 2 is used to compute the ground-truth point-wise noise map in the original RGB space. A lower resolution noise map is obtained by passing the point-wise noise map through successive 2x2 average pooling stages, which results in a weighted average where local noise level changes have a higher contribution, while simultaneously increasing the receptive field when compared to a 16x16 average pooling or 16x16 downsampling. It is then reshaped to have the same height and width as the image latent, in order to be concatenated.

Two variants of the model are proposed, by reviewing different resolution noise maps. In particular, different channel sizes of the input noise map are explored, either using 3 or 12 channels for the noise map, and corresponding to 4 and 3 average pooling stages respectively. We denote the model which uses a noise map with 3 channels as *non-blind S*, and the model with 12 channels as *non-blind B*. The pipeline to obtain both noise maps is presented in Figure 2.

During the training, the MSE is used as a distortion metric, between the original noise-free and the decoded image. This is equivalent to considering only the distortion term from the rate-distortion trade-off in the baseline compression model, with no perceptual transform applied to images. The non-blind architecture is shown in Figure 3.

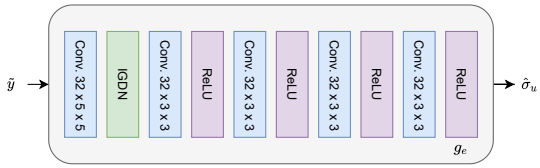


Figure 6: g_e architecture used in *blind E* to estimate the uniform noise level map σ_u, based on the architecture of CNN_E from CBDNet.[17]

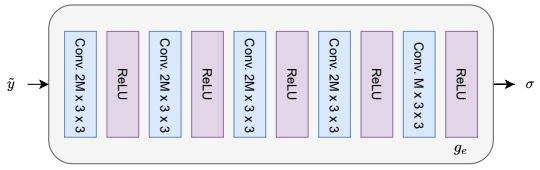


Figure 7: g_e architecture used in *blind L* to estimate the point-wise latent noise level map $\boldsymbol{\sigma}$, based on the architecture of CNN_E from CBDNet .[17] M is the number of channels of the noisy latent $\tilde{\boldsymbol{y}}$.

As obtaining the ground truth spatially variant noise level map may not be always feasible in practice, a relaxation of the *non-blind S* model is proposed. During inference only, a uniform noise map $\boldsymbol{\sigma}_u$ containing the empirical noise level of the image is used instead of the ground truth spatially variant noise map. We refer to this pipeline as *non-blind U*.

3.3 Blind combined decoding and denoising with noise map estimation

An additional blind solution is proposed, denoted as *blind E*, taking advantage of an additional learned model to estimate the noise map. Similarly to CBDNet,[17] the pipeline is composed of a subnetwork (denoted here as g_e) to estimate the noise level, and of a non-blind denoising subnetwork. The estimation network is trained separately from the decoder, using Mean Square Error between a uniform ground truth noise level map and the output noise map as the objective. The employed denoising network is the non-blind denoising decoder g_s presented in Section 3.2. More specifically, the decoder is chosen depending on the dimensionality of the latent space in the baseline compression network. The image latent is composed of either 192 or 384 channels, for which the *non-blind S* and the *non-blind B* decoders are used respectively.

The pipeline of the *blind E* method is represented in Figure 4.

3.4 Blind combined decoding and denoising with noise modeling in the latent space

Instead of estimating the noise level information in the original RGB space, an additional method that aims at inferring the point-wise *latent noise* level directly from the latent space, here denoted as *blind L*, is presented. Notably, we define the *latent noise* as the difference between the quantized latent $\tilde{\boldsymbol{y}}$ of the noisy image and the inversely quantized latent $\boldsymbol{y}$ of the clean image. The relationship between the latent noise and the noise applied to the original image is unknown, as the latent noise is influenced by the encoding and quantization operations. We approximate this noise in the latent image as zero-mean, point-wise independent Gaussian noise applied to the clean latent, as such is typically done in the literature for noise with unknown properties applied to RGB images.[17,18] The network is trained for inference of the latent noise level map $\boldsymbol{\sigma}$ and for combined denoising and decoding of the noisy latent $\tilde{\boldsymbol{y}}$ simultaneously, using the following loss function :

$$L(\mathcal{D};\boldsymbol{\theta}) = \boldsymbol{E}_{(\boldsymbol{x},\tilde{\boldsymbol{y}})\sim\mathcal{D}}\left[\frac{1}{2}\sum_i\left[\log\left(\sigma_i^2+\varepsilon\right)+\frac{(\tilde{y}_i-\hat{y}_i)^2+\varepsilon}{\sigma_i^2+\varepsilon}+\frac{\hat{y}^2}{s_i^2+\varepsilon}\right]+\lambda||\hat{\boldsymbol{x}}-\boldsymbol{x}||_2^2\right] \tag{3}$$

Where $\boldsymbol{\theta}$ refers to the learned parameters of g_s and g_e. $\mathcal{D}$ is the set of clean-image and noisy-latent pairs $(\boldsymbol{x},\tilde{\boldsymbol{y}})$ that compose the training dataset. $\lambda > 0, \varepsilon > 0$ are hyperparameters. The detailed derivation of the loss

function can be found in Appendix A. In our implementation, we choose $\varepsilon = 1e-3$ and use the same λ value as in the rate-distortion trade-off from the baseline compression model. Analogously to the non-blind methods presented in Section 3.2, the number of output channels in the first convolutional layer of the decoder is doubled and $\boldsymbol{\sigma}$ is concatenated to $\tilde{\boldsymbol{y}}$ before decoding.

4. EXPERIMENTAL RESULTS AND DISCUSSION

Results of the proposed pipelines are computed on images from the JPEG AI noisy test set.[28] Notably, the dataset includes both noisy images and the relative noise-free original versions, presenting three different noise levels (i.e. low, medium and high), randomly generated using the JPEG AI noise generator.[27] The results of the proposed methods are compared to two anchor pipelines, namely:

1. **Original anchor**: the learning-based anchor denoising method, i.e. FFDNet,[16] is used to denoise the images in the JPEG AI noisy test dataset. The denoising is applied before any compression, thus avoiding any compression artifact.

2. **Decoded anchor**: the learning-based anchor denoising method, i.e. FFDNet,[16] is applied in the pixel domain after encoding and decoding the noisy test images with the variational autoencoder with a scale hyperprior model at multiple bitrates.[8]

The results are reported both in the form of rate-distortion plots and through visual examples. In this paper, only the results for images *'00001'* and *'00016'* of the JPEG AI datasets[28] are reported. Notably, the first image was chosen as it presents a wide smooth area, i.e. a white background; instead, the second image presents high-frequency patterns, corresponding to the feathers of a bird. Therefore, the performance of the proposed methods are assessed on a variety of conditions.

Figure 8 and Figure 9 present the objective results for image *'00001'* and image *'00016'* respectively. The results are presented in the form of rate-distortion plots for a number of metrics, namely PSNR Y (i.e. computed on the luminance component), MS-SSIM Y, VIFp Y, FSIM, and VMAF.[28] The objective metrics have been computed using the objective quality framework provided by JPEG AI [†].

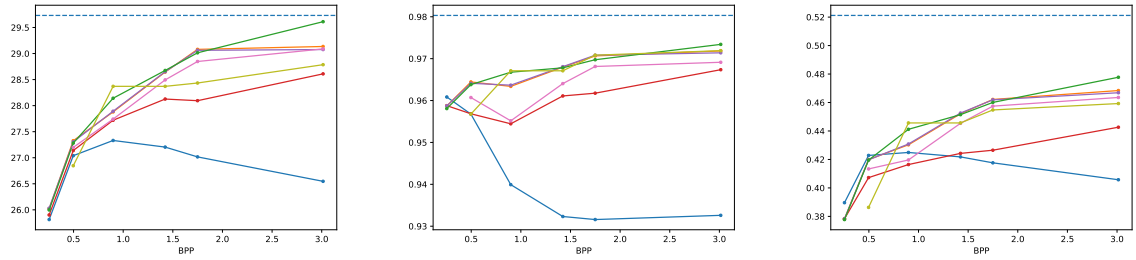
Figure 10 and Figure 11, on the other hand, present some visual examples of details from images decoded and denoised with the proposed methods. Notably, the results for the highest rate (i.e. approximately 1.5bpp) are presented, as the effects of compression are milder at such rates and therefore the visual difference between the methods is more prominent.

4.1 Discussion

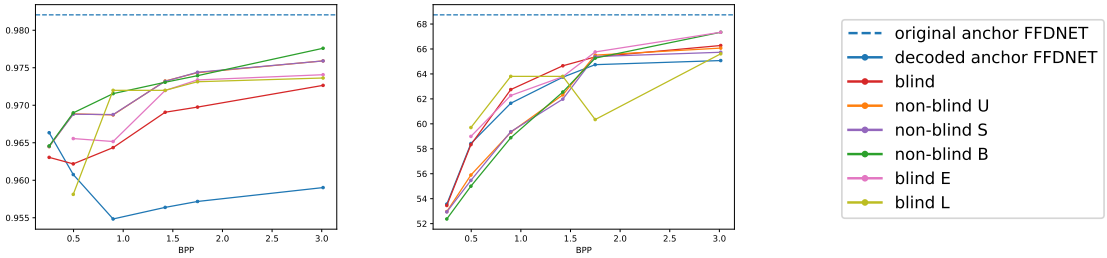
The objective quality and visual results presented above highlight that all the proposed methods are able to improve the performance of the decoded anchor, but generally not the performance of the original anchor. This can be explained by the fact that FFDNet was trained only on noisy uncompressed images, therefore the performance on the decoded images is expected to be lower and could be improved by including examples of encoded noisy images during the training of the network. In addition, the following observations can be drawn from the rate-distortion plots:

- the proposed *blind* method, being the simplest and least complex method, shows lower performance than the non-blind methods. This indicates that the decoder benefits from the added information about the noise, generating better results both in terms of subjective visual quality and according to the objective quality metrics. Regardless, the *blind* method has the advantage of not requiring any prior information about the noise level, or any additional network which estimates information about the noise, making it suitable for applications with low-latency constraints.

[†] https://gitlab.com/wg1/jpeg-ai/jpeg-ai-qaf

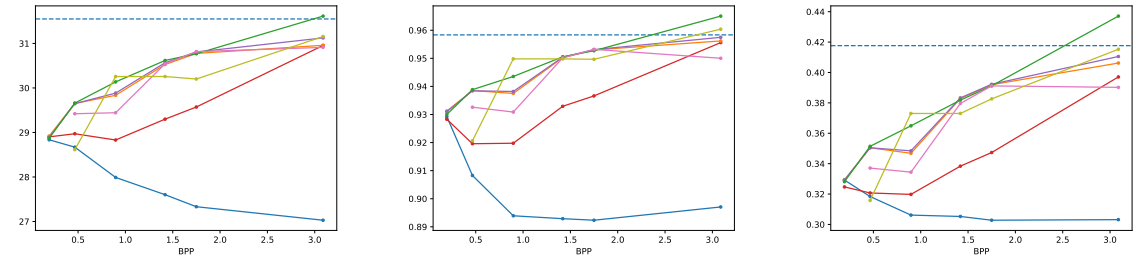


(a) PSNR Y (b) MS-SSIM (c) VIFp

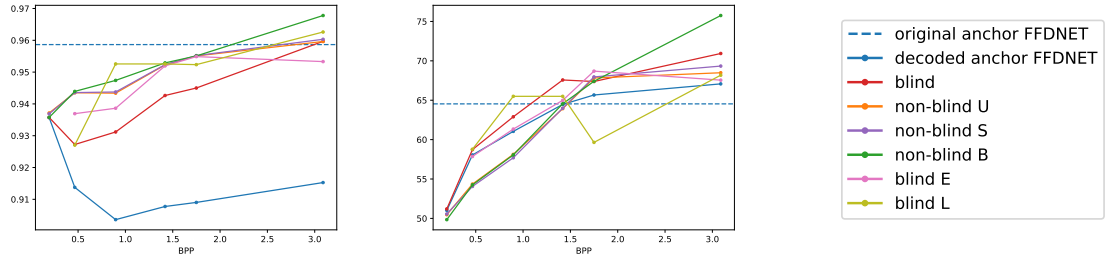


(d) FSIM (e) VMAF

Figure 8: Rate-distortion results for image *'00001'* of the JPEG AI noisy test set. The results regard only the images with the highest noise level.



(a) PSNR Y (b) MS-SSIM (c) VIFp



(d) FSIM (e) VMAF

Figure 9: Rate-distortion results for image *'00016'* of the JPEG AI noisy test set. The results regard only the images with the highest noise level.

(a) original

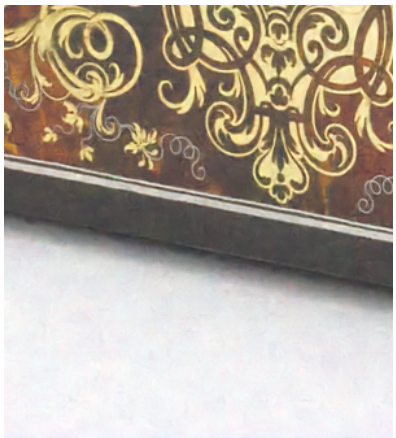
(b) noisy

(c) original anchor

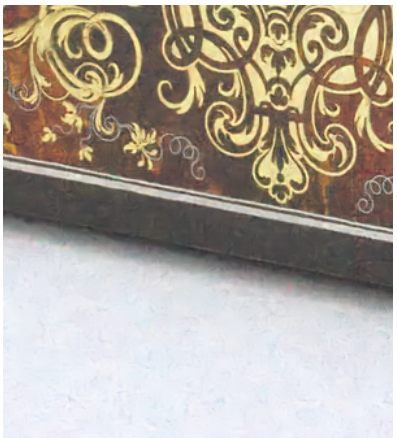
(d) decoded anchor

(e) blind

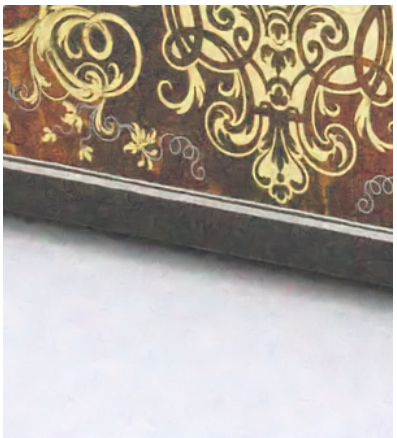
(f) non-blind U



(g) non-blind B



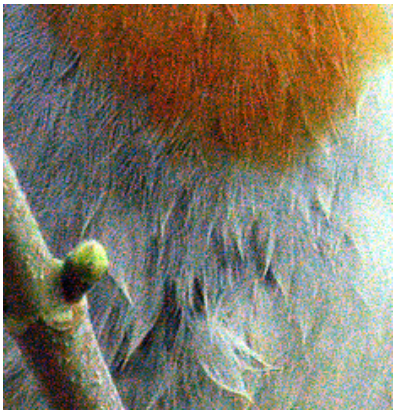
(h) blind E



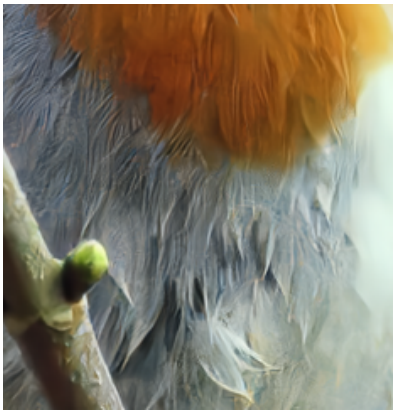
(i) blind L

Figure 10: Visual results for image *'00001'* of the JPEG AI noisy test set. The results regard only the images with the highest noise level, encoded at the highest bitrate.

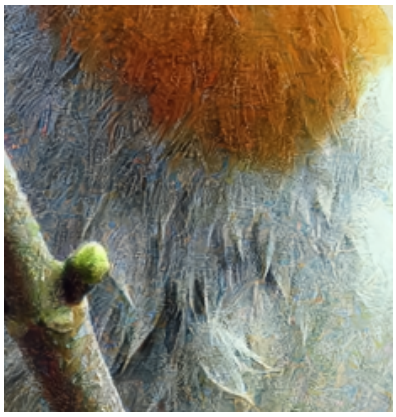
(a) original



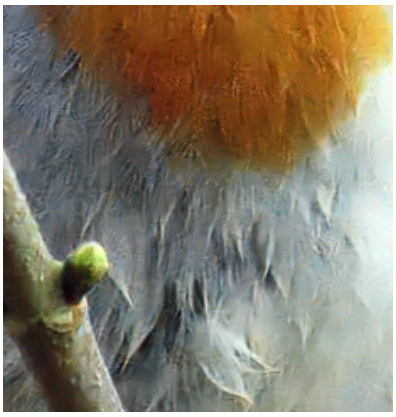
(b) noisy



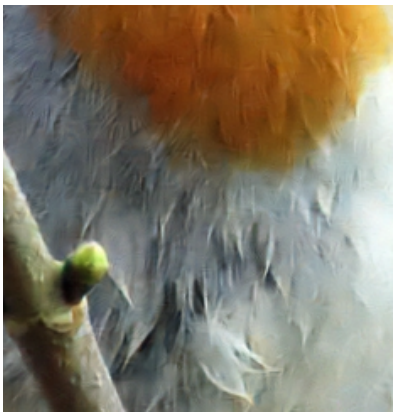
(c) original anchor



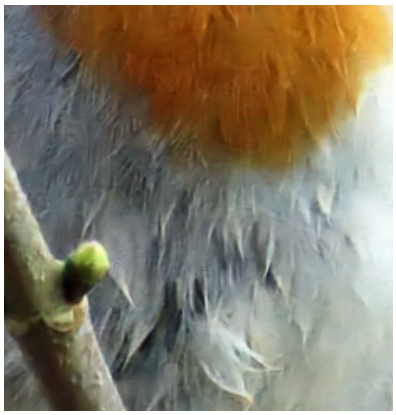
(d) decoded anchor



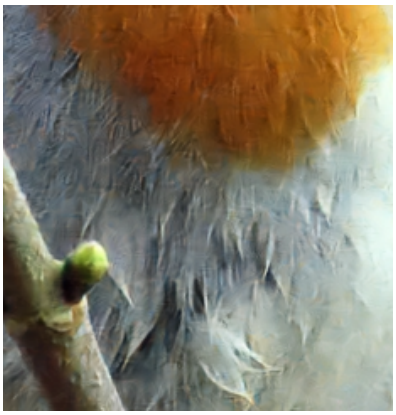
(e) blind



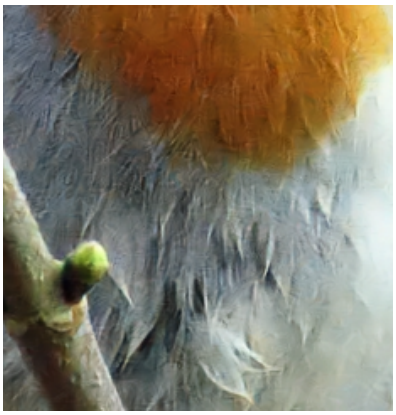
(f) non-blind U



(g) non-blind B



(h) blind E



(i) blind L

Figure 11: Visual results for image *'00016'* of the JPEG AI noisy test set. The results regard only the images with the highest noise level, encoded at the highest bitrate.

- the proposed non-blind methods, namely *non-blind U*, *non-blind S* and *non-blind B*, are all able to improve the performance of the blind method. Notably, the *non-blind U* and the *non-blind S* methods present similar performance, while the *non-blind B* method presents improved objective performance. Yet, the difference in terms of visual quality between these methods is limited.
- while the *blind E* method presents higher performance than the *blind* method, benefiting from the noise level estimation, the objective metrics reveal lower performance than the non-blind methods. This shows that the estimating network is not able to accurately estimate the noise map, and revealing the need for further research in this direction.
- the *blind L* method presents improved performance, especially at the lower bitrates, where the distribution of the noise is highly distorted by the encoding and quantization operations. Visually, this method is able to preserve high-frequency details better than the other proposed methods.

5. CONCLUSIONS

In this paper, different methods for integrating denoising operations into the decoder of a learning-based compression framework are proposed. Notably, both blind and non-blind solutions have been explored. Experimental results reveal that additional information about the noise distribution benefits the combined methods, which achieve higher performance both objectively and subjectively when compared to an anchor performing decoding and denoising in cascade. While in this paper the proposed strategies are only applied to a single framework, they are flexible enough to be adapted to a wide variety of other learning-based compression methods, e.g. in the future it can be applied to the upcoming JPEG AI learning-based codec. In this work, only the distortion metric used in by original compression model (i.e. MSE) is used. As future work, a trade-off between two objective metrics (e.g. MSE and SSIM) or a metric specific to noise reduction performance assessment might be used to further improve the perceptual visual quality of the decoded and denoised images. Additionally, more advanced approaches to estimate properties of *latent noise* might be explored.

ACKNOWLEDGMENTS

The authors would like to acknowledge support from the Swiss National Scientific Research project entitled "Advanced Visual Representation and Coding in Augmented and Virtual Reality" under grant number 200021_178854.

REFERENCES

[1] Testolina, M., Upenik, E., and Ebrahimi, T., "Comprehensive assessment of image compression algorithms," in [*Applications of Digital Image Processing XLIII*], **11510**, 469–485, SPIE (2020).

[2] ISO/IEC JTC 1/SC29/WG1 N89022, "Report on the JPEG AI Call for Evidence Results." 89th JPEG Meeting, Online, October 2020.

[3] ISO/IEC JTC 1/SC29/WG1 N100250, "Report on the JPEG AI Call for Proposals Results." 96th JPEG Meeting, Online, July 2022.

[4] ISO/IEC JTC 1/SC29/WG1 N100094, "Use Cases and Requirements for JPEG AI." 94th JPEG Meeting, Online, January 2022.

[5] Lu, Y., Barras, L., and Ebrahimi, T., "A novel framework for assessment of deep face recognition systems in realistic conditions," in [*10th European Workshop on Visual Information Processing (EUVIP)*], IEEE (2022).

[6] Foi, A., Trimeche, M., Katkovnik, V., and Egiazarian, K., "Practical poissonian-gaussian noise modeling and fitting for single-image raw-data," *IEEE Transactions on Image Processing* **17**(10), 1737–1754 (2008).

[7] Ballé, J., Laparra, V., and Simoncelli, E. P., "End-to-end optimized image compression," in [*5th International Conference on Learning Representations, ICLR 2017*], (2017).

[8] Ballé, J., Minnen, D., Singh, S., Hwang, S. J., and Johnston, N., "Variational image compression with a scale hyperprior," in [*International Conference on Learning Representations*], (2018).

[9] Minnen, D., Ballé, J., and Toderici, G. D., "Joint autoregressive and hierarchical priors for learned image compression," *Advances in neural information processing systems* **31** (2018).

[10] Cheng, Z., Sun, H., Takeuchi, M., and Katto, J., "Learned image compression with discretized gaussian mixture likelihoods and attention modules," in [*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*], 7939–7948 (2020).

[11] Agustsson, E., Tschannen, M., Mentzer, F., Timofte, R., and Gool, L. V., "Generative adversarial networks for extreme learned image compression," in [*Proceedings of the IEEE/CVF International Conference on Computer Vision*], 221–231 (2019).

[12] Mentzer, F., Toderici, G. D., Tschannen, M., and Agustsson, E., "High-fidelity generative image compression," *Advances in Neural Information Processing Systems* **33**, 11913–11924 (2020).

[13] Ascenso, J., Akyazi, P., Pereira, F., and Ebrahimi, T., "Learning-based image coding: early solutions reviewing and subjective quality evaluation," in [*Optics, Photonics and Digital Technologies for Imaging Applications VI*], **11353**, 164–176, SPIE (2020).

[14] Chang, S. G., Yu, B., and Vetterli, M., "Adaptive wavelet thresholding for image denoising and compression," *IEEE transactions on image processing* **9**(9), 1532–1546 (2000).

[15] Zhang, K., Zuo, W., Chen, Y., Meng, D., and Zhang, L., "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Transactions on Image Processing* **26**(7), 3142–3155 (2017).

[16] Zhang, K., Zuo, W., and Zhang, L., "Ffdnet: Toward a fast and flexible solution for cnn-based image denoising," *IEEE Transactions on Image Processing* **27**(9), 4608–4622 (2018).

[17] Guo, S., Yan, Z., Zhang, K., Zuo, W., and Zhang, L., "Toward convolutional blind denoising of real photographs," in [*Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*], 1712–1722 (2019).

[18] Yue, Z., Yong, H., Zhao, Q., Meng, D., and Zhang, L., "Variational denoising network: Toward blind noise modeling and removal," *Advances in neural information processing systems* **32** (2019).

[19] Choi, H. and Bajić, I. V., "Scalable image coding for humans and machines," *IEEE Transactions on Image Processing* **31**, 2739–2754 (2022).

[20] Torfason, R., Mentzer, F., Agustsson, E., Tschannen, M., Timofte, R., and Van Gool, L., "Towards image understanding from deep compression without decoding," in [*International Conference on Learning Representations*], (2018).

[21] Upenik, E., Testolina, M., and Ebrahimi, T., "Towards super resolution in the compressed domain of learning-based image codecs," in [*Applications of Digital Image Processing XLIV*], **11842**, 531–541, SPIE (2021).

[22] Testolina, M., Upenik, E., and Ebrahimi, T., "Towards image denoising in the latent space of learning-based compression," in [*Applications of Digital Image Processing XLIV*], **11842**, 412–422, SPIE (2021).

[23] Alvar, S. R., Ulhaq, M., Choi, H., and Bajić, I. V., "Joint image compression and denoising via latent-space scalability," *arXiv preprint arXiv:2205.01874* (2022).

[24] de Oliveira, V. A., Chabert, M., Oberlin, T., Poulliat, C., Bruno, M., Latry, C., Carlavan, M., Henrot, S., Falzon, F., and Camarero, R., "Satellite image compression and denoising with neural networks," *IEEE Geoscience and Remote Sensing Letters* **19**, 1–5 (2022).

[25] Cheng, K. L., Xie, Y., and Chen, Q., "Optimizing image compression via joint learning with denoising," *arXiv preprint arXiv:2207.10869* (2022).

[26] Bégaint, J., Racapé, F., Feltman, S., and Pushparaja, A., "Compressai: a pytorch library and evaluation platform for end-to-end compression research," *arXiv preprint arXiv:2011.03029* (2020).

[27] Alvar, S. R. and Bajić, I. V., "Practical noise simulation for rgb images," *arXiv preprint arXiv:2201.12773* (2022).

[28] ISO/IEC JTC1/SC29/WG1 N100106, "JPEG AI Common Training and Testing Conditions." 94th Meeting, Online, January 2022.

APPENDIX A. LOSS FUNCTION DERIVATION OF THE *BLIND L* METHOD

Notation :
$\boldsymbol{x}$: original noise-free image
$\boldsymbol{y}$: (unquantized) latent representation of the noise-free image, $\boldsymbol{y} = g_a(\boldsymbol{x})$
$\boldsymbol{s}$: noise-free latent scale hyperprior $\boldsymbol{s} = h_s(h_a(Q\{\boldsymbol{y}\}))$
$\tilde{\boldsymbol{y}}$: noisy latent
$\hat{\boldsymbol{x}}$: reconstructed image, $\hat{\boldsymbol{x}} = g_s(\tilde{\boldsymbol{y}}, \boldsymbol{\sigma})$
$\hat{\boldsymbol{y}}$: reconstructed latent, $\hat{\boldsymbol{y}} = g_a(\hat{\boldsymbol{x}}) = g_a(g_s(\tilde{\boldsymbol{y}}, \boldsymbol{\sigma}))$
$\boldsymbol{\sigma}$: noise level map of the noisy latent, $\boldsymbol{\sigma} = g_e(\tilde{\boldsymbol{y}})$
$\boldsymbol{z}$: unobserved noise-free latent

The combined denoising and decoding problem is first posed as the modeling of our data, original noise-free image/noisy-latent pairs $(\boldsymbol{x}, \tilde{\boldsymbol{y}})$. The objective is to find the network parameter values that maximize the expected log-likelihood of the joint distribution $p(\boldsymbol{x}, \tilde{\boldsymbol{y}})$, over the dataset $\mathcal{D}$ of original noise-free images/noisy-latent pairs.

$$\boldsymbol{E}_{(\boldsymbol{x},\tilde{\boldsymbol{y}})\in\mathcal{D}}[\log p(\boldsymbol{x}, \tilde{\boldsymbol{y}})] = \boldsymbol{E}_{(\boldsymbol{x},\tilde{\boldsymbol{y}})\in\mathcal{D}}[\log p(\tilde{\boldsymbol{y}})] + \boldsymbol{E}_{(\boldsymbol{x},\tilde{\boldsymbol{y}})\in\mathcal{D}}[\log p(\boldsymbol{x}|\tilde{\boldsymbol{y}})] \tag{4}$$

Under the same assumptions as in the compression framework,[8] where λ is the hyper-parameter of the rate-distortion trade-off :

$$(\boldsymbol{x}|\tilde{\boldsymbol{y}}) \sim \mathcal{N}(\hat{\boldsymbol{x}}, (2\lambda)^{-1}\boldsymbol{I}) \tag{5}$$

$$\log p(\boldsymbol{x}|\tilde{\boldsymbol{y}}) = -\lambda||\boldsymbol{x} - \hat{\boldsymbol{x}}||_2^2 + cst. \tag{6}$$

The above allows for interpretation of the objective in parallel to that of a compression model :

$$\begin{aligned}\boldsymbol{E}_{(\boldsymbol{x},\tilde{\boldsymbol{y}})\in\mathcal{D}}[-\log p(\boldsymbol{x}, \tilde{\boldsymbol{y}})] &= E_{(\boldsymbol{x},\tilde{\boldsymbol{y}})\in\mathcal{D}}[-\log p(\tilde{\boldsymbol{y}})] + \lambda||\boldsymbol{x} - \hat{\boldsymbol{x}}||_2^2 \\ &= R + \lambda D\end{aligned}$$

Where the term R corresponds to the rate of latent $\tilde{\boldsymbol{y}}$ and D to the distortion in a classic rate distortion trade-off, where before quantization, the latent is perturbed by a more complex noise source. Note that unlike learned compression, our scope here is not to find a latent representation that minimizes rate, but to minimize the rate given the fixed noisy latent, by infering parameters $\boldsymbol{\sigma}$ and $\hat{\boldsymbol{y}}$ on the distribution of the latent.

As the evidence $\log \tilde{\boldsymbol{y}}$ is untractable, we thus consider instead its evidence lowerbound, using an approximation $q(\boldsymbol{z})$ of the true distribution $p(\boldsymbol{z}|\tilde{\boldsymbol{y}})$, similarly to what is proposed for VDNet.[18] Note that unlike in the framework presented by Yue et al.,[18] only the noise-free latent $\boldsymbol{z}$ is an unobserved variable and not the noise level map $\boldsymbol{\sigma}$.

$$\log p(\tilde{\boldsymbol{y}}) = ELBO(q) + KL(q(\boldsymbol{z})||p(\boldsymbol{z}|\tilde{\boldsymbol{y}})) \geq ELBO(q) \tag{7}$$

$$ELBO(q) = \boldsymbol{E}_{\boldsymbol{z}\sim q}[log(p(\tilde{\boldsymbol{y}}|\boldsymbol{z}))] - KL(q(\boldsymbol{z})||p(\boldsymbol{z})) \tag{8}$$

$$= \boldsymbol{E}_{\boldsymbol{z}\sim q}[log(p(\tilde{\boldsymbol{y}}|\boldsymbol{z}))] + \boldsymbol{E}_{\boldsymbol{z}\sim q}[\log p(\boldsymbol{z})] - \boldsymbol{E}_{\boldsymbol{z}\sim q}[\log q(\boldsymbol{z})] \tag{9}$$

Similarly to the approach taken by VDNet framework[18] for denoising in the RGB space but here in the latent space, a true distribution is imposed on $\boldsymbol{z}$, where ε is a hyperparameter. The distribution $q(\boldsymbol{z})$ which approximates $p(\boldsymbol{z}|\tilde{\boldsymbol{y}})$ is also defined:

$$\boldsymbol{z} \sim \mathcal{N}(0, S + \varepsilon I) \tag{10}$$

$$S_{ij} = \begin{cases} s_i^2 & \text{if i=j} \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

$$\boldsymbol{z} \overset{q}{\sim} \mathcal{N}(\hat{\boldsymbol{y}}, \varepsilon \boldsymbol{I}) \tag{12}$$

Finally, based on our point-wise independent gaussian *latent noise* assumption, the distribution of $\tilde{\boldsymbol{y}}|\boldsymbol{z}$ is given by :

$$(\tilde{\boldsymbol{y}}|\boldsymbol{z}) \sim \mathcal{N}(\boldsymbol{z}, \boldsymbol{\Sigma} + \varepsilon I) \tag{13}$$

$$\Sigma_{ij} = \begin{cases} \sigma_i^2 & \text{if i=j} \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

The 3 terms of the evidence lower bound can be computed as :

$$\boldsymbol{E}_{\boldsymbol{z}\sim q}[\log q(\boldsymbol{z})] = -\frac{1}{2}\sum_i [\log 2\pi + \log \varepsilon + 1]$$

$$\boldsymbol{E}_{\boldsymbol{z}\sim q}[\log p(\boldsymbol{z})] = -\frac{1}{2}\sum_i \left[\log 2\pi + \log\left(s_i^2 + \varepsilon\right) + \frac{\varepsilon}{s_i^2 + \varepsilon} + \frac{\hat{y}_i^2}{s_i^2 + \varepsilon}\right]$$

$$\boldsymbol{E}_{\boldsymbol{z}\sim q}[log(p(\tilde{\boldsymbol{y}}|\boldsymbol{z}))] = -\frac{1}{2}\sum_i \left[\log 2\pi + \log\left(\sigma_i^2 + \varepsilon\right) + \frac{\varepsilon}{\sigma_i^2 + \varepsilon} + \frac{(\tilde{y}_i - \hat{y}_i)^2}{\sigma_i^2 + \varepsilon}\right]$$

From which the evidence lower bound is obtained :

$$ELBO(q) = -\frac{1}{2}\sum_i \left[\log\left(\sigma_i^2 + \varepsilon\right) + \frac{(\tilde{y}_i - \hat{y}_i)^2 + \varepsilon}{\sigma_i^2 + \varepsilon} + \frac{\hat{y}_i^2}{s_i^2 + \varepsilon}\right] + cst.$$

Which gives the following loss function to minimize as a function of the learned parameter $\boldsymbol{\theta}$ of g_s and g_e :

$$\mathcal{L}(\mathcal{D}; \boldsymbol{\theta}) = \boldsymbol{E}_{(\boldsymbol{x},\tilde{\boldsymbol{y}})\sim\mathcal{D}}\left[\frac{1}{2}\sum_i \left[\log\left(\sigma_i^2 + \varepsilon\right) + \frac{(\tilde{y}_i - \hat{y}_i)^2 + \varepsilon}{\sigma_i^2 + \varepsilon} + \frac{\hat{y}^2}{s_i^2 + \varepsilon}\right] + \lambda||\hat{\boldsymbol{x}} - \boldsymbol{x}||_2^2\right] \tag{15}$$