Quantized and Regularized Optimization for Coding Images Using Steered Mixtures-of-Experts

Rolf Jongebloed, Erik Bochinski, Lieven Lange, and Thomas Sikora

Communication Systems Group
Technische Universität Berlin
Einsteinufer 17
10587 Berlin, Germany
{jongebloed,bochinski,lange,sikora}@nue.tu-berlin.de

Abstract

Compression algorithms that employ Mixtures-of-Experts depart drastically from standard hybrid block-based transform domain approaches as in JPEG and MPEG coders. In previous works we introduced the concept of *Steered Mixtures-of-Experts* (SMoEs) to arrive at sparse representations of signals. SMoEs are gating networks trained in a machine learning approach that allow individual experts to explain and harvest directional long-range correlation in the N-dimensional signal space. Previous results showed excellent potential for compression of images and videos but the reconstruction quality was mainly limited to low and medium image quality. In this paper we provide evidence that SMoEs can compete with JPEG2000 at mid- and high-range bit-rates. To this end we introduce a SMoE approach for compression of color images with specialized gates and steering experts. A novel machine learning approach is introduced that optimizes RD-performance of quantized SMoEs towards SSIM using fake quantization. We drastically improve our previous results and outperform JPEG by up to 42%.

1 Introduction

In recent years a lot of effort has been made to investigate the potential of the novel and well-defined *Steered Mixture-of-Experts* (SMoE) framework for coding image [1, 2], video [3] and even higher dimensional image modalities such as light field [4] and light field video [5]. This compression approach drastically departs from conventional block-based frequency domain transform coding techniques, such as JPEG and JPEG2000, as SMoE models explain the data in the spatial rather than in the transform domain. As a derivative of the so-called *Mixture-of-Experts* (MoE) approach it follows the divide-and-conquer principle [6]. Each expert acts as a regressor weighted by a gating function. This arrives at a soft partitioning of the input space to determine in which regions the experts are trustworthy. The set of arbitrarily-shaped gating functions are modeled by a mixture of steered *Gaussian* kernels indicating the spatial relevance of the respective expert. In general experts and gates can extend over the entire image plane and thus harvest long-range correlation. In some instances very simple single experts can be responsible for the reconstruction of arbitrarily-shaped regions with thousands of pixels, as we will show in this paper.
In previous works ([1, 3–5]) *Gaussian Mixture Models* (GMM) were used to jointly

This work was supported by a Google Faculty Research Award 2016 in Machine Perception.

represent the experts and associated gates trained by the well-known *Expectation-Maximization* (EM) algorithm to find the representing model parameters. However, the EM algorithm is not necessarily optimal for regression tasks as it maximizes the joint likelihood function of the input and output space rather than minimizing the Mean Squared Error (MSE). Therefore, an approach minimizing the MSE using Gradient Descent (GD) has been presented in [2] yielding promising results in very low bit-rate cases. Nevertheless, the recent works have in common that the parameters are estimated first - either by the EM algorithm or by GD - and then quantized for coding. Additionally, these approaches still suffer from the problem of initialization of the parameters, such that the maximum potential of SMoE models is not fully exploited.
In [7] this problem has been tackled by starting the optimization with a very large number of components initialized on an evenly distributed grid. Additional objectives are introduced to the loss function to establish a trade-off between the number of parameters and the regression error to enforce sparsity of the image representation while achieving high reconstruction qualities.
This work extends the framework presented in [7] in multiple ways: First, an extension to model color information is made. Instead of minimizing the MSE, this framework is able to maximize the structural similarity (SSIM) index referencing the perceived reconstruction quality [8]. This results both visually and quantitatively in drastically improved results compared to our previous works. Because the reconstruction quality of the model degrades quickly when performing optimization after modelling, the quantization step is incorporated into the optimization process of the SMoE model. Evaluations show that our approach outperforms JPEG by up to 42.48% and is competitive to JPEG2000 in mid-range to high-range bit-rates. Furthermore, our approach can be easily extended to model and encode video, light field/light field video, or other types of higher dimensional data using the same modelling and coding pipeline.

2 Steered Mixture-of-Experts

In our Steered Mixture of Experts (SMoE) framework the underlying prediction function of an amplitude of a pixel (luminance output) given its position (spatial input) is formulated as a weighted sum of K experts:

$$y_p(\boldsymbol{x}) = \sum_{k=1}^{K} m_k(\boldsymbol{x}) w_k(\boldsymbol{x}). \tag{1}$$

The experts in our approach can be (hyper-)planes in the pixel domain or input space acting as regressors

$$m_k(\boldsymbol{x}) = \boldsymbol{m}_k^T \boldsymbol{x} + m_{0,k} \tag{2}$$

described by $\boldsymbol{m}_k$ containing the slopes in each dimension of the pixel domain and an offset $m_{0,k}$. The weighting function, also called gating, in Eq. 1 is a weighted soft

max function

$$w_k(\boldsymbol{x}) = \frac{\pi_k \cdot \mathcal{K}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{A}_k)}{\sum_{j=1}^{K} \pi_j \cdot \mathcal{K}(\boldsymbol{x}; \boldsymbol{\mu}_j, \boldsymbol{A}_j)} \tag{3}$$

with mixing coefficients π_k. We employ Gaussian kernels

$$\mathcal{K}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{A}) = \exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{A}\boldsymbol{A}^T(\boldsymbol{x}-\boldsymbol{\mu})\right] \tag{4}$$

defined by their center positions $\boldsymbol{\mu} \in \mathbb{R}^d$ and steering parameters

$$\boldsymbol{A}_k := \begin{pmatrix} a_{11,k} & & \\ \vdots & \ddots & 0 \\ a_{d1,k} & \dots & a_{dd,k} \end{pmatrix} \tag{5}$$

as the inverse cholesky decomposition of a covariance matrix where only the lower triangular part is nonzero and d is the dimension of the input space.
To represent also the chrominance channels, the experts are extended as multidimensional regressors, for each channel a (hyper-)plane is fitted into the corresponding domain. Thus, having experts with three dimensional output allows Eq. 2 to be rewritten as

$$\boldsymbol{m}_k(\boldsymbol{x}) = \boldsymbol{M}_k \boldsymbol{x} + \boldsymbol{m}_{0,k} \tag{6}$$

with

$$\boldsymbol{M}_k = \left[\boldsymbol{m}_{\mathrm{Y},k}^T\ \boldsymbol{m}_{\mathrm{U},k}^T\ \boldsymbol{m}_{\mathrm{V},k}^T\right]^T \tag{7}$$

where $\boldsymbol{m}_{\mathrm{Y},k}$, $\boldsymbol{m}_{\mathrm{U},k}$, and $\boldsymbol{m}_{\mathrm{V},k}$ contain the slopes in the Y,U and V domains, respectively, and $\boldsymbol{m}_{0,k}$ the offsets for each channel. Consequently, the prediction function from Eq. 1 has to be reformulated to:

$$\boldsymbol{y}_p(\boldsymbol{x}) = \sum_{k=1}^{K} \boldsymbol{m}_k(\boldsymbol{x}) w_k(\boldsymbol{x}). \tag{8}$$

Parameter Optimization Similar to the recent work in [7], a multi-task optimization technique is used to train the SMoE model. Instead of optimizing the MSE, the main optimization criterion in this work is to maximize towards the SSIM. The loss function formulated as a minimization problem is defined as follows:

$$\mathcal{L}^{\mathrm{SSIM}} := (1 - \mathrm{SSIM}(I_{\mathrm{Target}}, I_{\mathrm{Rec}})) \tag{9}$$

where I_{Target} and I_{Rec} are the original and reconstructed images, respectively, using default parameters as defined in [8]. To ensure a sparse representation of the underlying image while providing image reconstructions of high qualities, an additional sparsity promoting regularization loss

$$\mathcal{L}^{\mathrm{S}} := \lambda_{\mathrm{S}} \cdot \sum_{k=1}^{K} \pi_k \tag{10}$$

is used as in [7] where its influence can be adjusted by λ_{S}. Therefore, the mixing coefficients π_k are gradually decreased until values ≤ 0 are reached, stating that the corresponding kernel has no influence to the regression function, and thus, it can be removed from the model. As such, the final loss function is composed as

$$\mathcal{L} := \mathcal{L}^{\mathrm{SSIM}} + \mathcal{L}^{\mathrm{S}}. \tag{11}$$

which needs to be minimized by finding a set of parameters $\boldsymbol{A}_k$, $\boldsymbol{M}_k$, $\boldsymbol{m}_{0,k}$, $\boldsymbol{\mu}_k$ and π_k:

$$\underset{\boldsymbol{A},\boldsymbol{M},\boldsymbol{m},\boldsymbol{\mu},\boldsymbol{\pi}}{\arg\min} \{\mathcal{L}\} \tag{12}$$

following the negative gradient $-\nabla\mathcal{L}$ using Gradient Descent.

Parameter Quantization SMoE models define a highly complex dependence between the model parameters and regression function. Therefore, they are extremely sensitive towards quantization of parameters regarding the remaining reconstruction quality. It is crucial to optimize the model with already quantized parameters rather than to quantize the parameters after training. Therefore, the interaction of the parameters and the impact of quantization are considered jointly during optimization. The uniform quantization function is as follows:

$$\bar{v} := \min(\max(v, a), b) \tag{13}$$

$$\Delta := \frac{a-b}{2^n - 1} \tag{14}$$

$$\hat{v} := \mathrm{round}\left(\frac{\bar{v} - a}{\Delta}\right) \cdot \Delta + a \tag{15}$$

where $\bar{v}$ is the clipped value of the continuous parameter v within the range $[a, b]$, Δ is the step size depending on the number of bits n, and $\hat{v}$ is the resulting quantized parameter. During optimization, v is trained with full precision but the loss is computed with respect to its quantization $\hat{v}$.

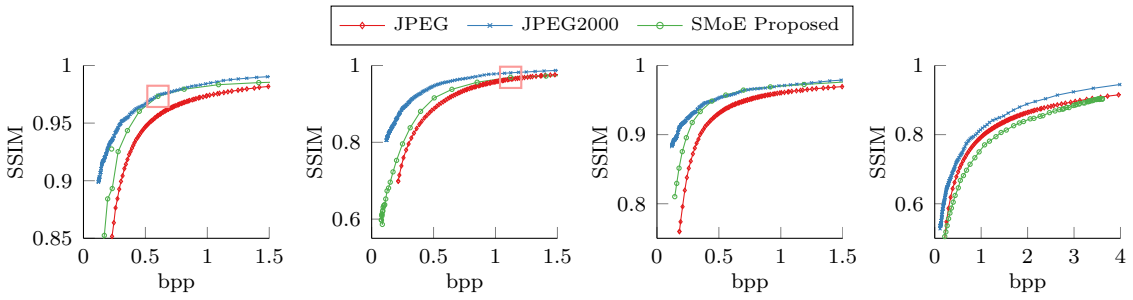
3 Experiments and Results

In this section we evaluate our coding approach for colored still images and compare the results to commonly used image compression standards, such as JPEG and JPEG2000. The implementation for optimization is done using the Tensorflow framework based on the recent work in [7].
As the gradients of the parameters to be optimized are of different magnitudes, different learning rates are required to exploit the maximum potential of the SMoE model. The learning rates for the Adam optimizer [9] are 1, 10^{-4} for $\boldsymbol{A}$, $\boldsymbol{\pi}$, respectively, and 10^{-3} for $\boldsymbol{\mu}$, $\boldsymbol{m}$ if not stated otherwise. The remaining parameters of the optimizer are set to the default values $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$ for all experiments. In our experiments we use a so-called *Constant Experts - Steered Gates* model for sparse

SMoE representation. As such the parameters $\boldsymbol{M}$ are not trained in any experiment, remain zero and need not be transmitted. This results in simple zero-order regressors and the more-elaborate steering gates explore directional long-range correlation. Taking all parameters $\boldsymbol{M}$ into account would increase the number of data drastically, while at the same time the quality barely improves as the structural information lies mostly in the gating shown in Fig. 3. Thus, spending too many bits on the expert's slopes is avoided for better compression performance.
The image modeling and coding process is divided into four steps: First, the model is initialized and pre-trained without regularization for sparsification. After convergence, regularization to sparsify the model is employed. Afterwards, fine-tuning to maximize the reconstruction quality without regularization is performed. Kernels with respective mixing coefficients reaching $\pi_k < 0$ during training in all steps are removed from the model. In the final step, the resulting model is coded using an entropy arithmetic coding approach.
During all steps of optimization, the quantization of the model parameters is employed using the Tensorflow's *fake quantization* function allowing for gradient computations and parameter updates with full precision [10]. All parameters are uniformly quantized while the minimum/maximum range of the quantizers for each kind of variable are also optimized at the same time, with the exception of the mixing coefficients π_k. As the ratios of the mixing coefficients to each other is highly relevant whereas their absolute values are not, the quantization range is fixed to $[0, 2]$. Additionally, the main diagonal entries of $\boldsymbol{A}$ have their own quantization range as they are always > 0, but the lower triangular entries are arbitrary.

Initialization and Pre-Training The kernels of the model are initialized with $\boldsymbol{\mu}$ distributed on regular grid of $k_1 \times k_2$ with $k_{1,2} = \lfloor \frac{\mathrm{Res}_{w,h}}{4} \rfloor$ where $\mathrm{Res}_{w,h}$ represents the number of pixels in the x and y dimension of the corresponding test image, respectively. All mixing coefficients are set to $\pi_k = 1$, i.e. to the middle of their defined range. $\boldsymbol{A}$ is initialized in a way that the distance between the centers of two kernels equals two standard deviations 2σ. As aforementioned, the slopes of the experts are not trained and remain zero in all cases, the offsets $\boldsymbol{m}_{0,k}$ are initialized with the mean of the respective channel (Y,U,V) of the training image where the gating of the corresponding kernel has maximum influence. Within the pre-training step the model is trained for 10k iterations, no sparsify regularization is applied ($\lambda_{\mathrm{S}} = 0$).

Regularization After pre-training, the regularization phase is employed. Best results are achieved by slowly introducing the regularization term and exponentially increasing the coefficient λ_{S}. Therefore, the following schedule is applied: $\lambda_{\mathrm{S}} = \frac{x^2}{k_1 \cdot k_2}$ with x evenly distributed in $[0.1, 15]$ with 50 steps. The fixed number of initial kernels $k_1 \cdot k_2$ is included to account for different sensitivity levels depending on the initialization of the kernels. In this phase the learning rate for $\boldsymbol{\pi}$ is decreased to $2 \cdot 10^{-5}$ to avoid kernels being dropped out too quickly and ensuring neighboring kernels can assimilate.



(a) *Lena* (b) *Barbara* (c) *Peppers* (d) *Baboon*

Figure 1: Rate-Distortion curves

Fine-tuning In the last step of optimization, the model is trained for an additional 500 training iterations without sparsify regularization to compensate for the prior trade-off between the SSIM $\mathcal{L}^{\text{SSIM}}$ and the regularization loss $\mathcal{L}^{\text{S}}$ to fine-tune the model towards a maximum SSIM value. Vanilla Gradient Descent is used with same learning rates as in the step before.

Arithmetic Coding Finally, the coding step is employed to transform the quantized parameters into a bit stream. An arithmetic coding approach is used to bring the bit-rate into line with the underlying entropy of the parameters. Each parameter is coded under the assumption that it is independent identically distributed by fitting a continuous distribution to the relative frequency of symbol occurrences. Three probability density functions (pdf) are fitted by likelihood: Normal, Laplacian, and Maxwell. For each, an estimation of the model parameters describing the underlying pdf is determined on the signal. Finally, the best fitting pdf is chosen by MSE comparison. In the final bit stream an index of the chosen pdf, the model parameters, and a binary arithmetic coded version of the zero-one binary stream are transmitted right before the arithmetic coded parameter data.
In general, each kind of parameter can be fitted to one of the three aforementioned density functions quite well except the center $\boldsymbol{\mu}$ as they are distributed over the entire image and consequently each symbol occurs only once. Therefore, we only train and code the different center position regarding their start positions as we initialize the center position on an evenly distributed grid, which is known at the decoder side. Hence, an additional bit stream only containing flags for signaling which kernels are still active is needed to reconstruct the center assignments as some kernels are removed due to the sparsification promoting regularization.

Results The whole compression process including modeling and coding has been tested on 512×512 pixel seized color images. These images have been filtered by the *Block-matching and 3D filtering* (BM3D) algorithm [11] to reduce the noise corruption before optimization. Noise removal is a common technique to save bits in image and video compression [12–14]. In our SMoE framework we have not incorporated yet sophisticated experts to model noise-like signals. Because the loss function is mainly

Figure 2: Visual Comparison at approx. same bit-rates for *Lena* ($\sim$0.6 bpp) and *Barbara* ($\sim$1.12 bpp). The depicted results of our proposed method are highlighted with □ in the respective curve in Fig. 1. (best viewed in color and zoomed-in)

defined by the SSIM function, the optimization would focus too much on modeling the local variance of the image due to the underlying noise rather than the structural elements of the images. The standard deviation parameter of the BM3D algorithm is set $\sigma = 4.7$ for all images.

The parameters $\boldsymbol{A}, \boldsymbol{m}, \boldsymbol{\mu}, \boldsymbol{\pi}$ are quantized with 9, 7, 10 and 4 bits, respectively. As the luminance channel is the most important regarding the human visual system compared to the chrominance channels the SSIM function in Eq. 9 is evaluated for each channel independently and added up with weights Y:U:V $\leftrightarrow$ 6:1:1 to give the luminance channel more relevance within the optimization process. The same weights are used to determine the SSIM value for validation.

Fig. 1 depicts rate-distortion curves for *Lena*, *Barbara*, *Peppers*, and *Baboon*. To make fair comparisons we consider the filtered images as ground truth and use them also for the JPEG and JPEG2000 reference methods. Without *quantization aware training* (quantized after finishing the training) results drastically deteriorate. It can be seen that our approach outperforms JPEG in terms of SSIM for all test images except *Baboon*. As *Baboon* contains predominantly very high frequency content, a very high number of components is necessary to represent it. Surprisingly, at very high bit-rates our approach is visually comparable to JPEG and JPEG2000. Very promising results are achieved from mid to high bit-rates for *Lena* and *Peppers* - our method is even competitive to JPEG2000. The results for *Barbara* are also noteworthy. Although a lot of high frequency structures are included in this image it outperforms

Figure 3: Gating for *Lena* ($K = 2717$) and *Barbara* ($K = 5496$), respectively, showing color coded the area of maximum influence of a respective kernel. A closer look reveals that fine-grained details are modelled by placing small kernels above larger ones belonging to the background.

JPEG at any bit-rate. Even more convincing are visual comparisons depicted in Fig. 2 showing results of JPEG, JPEG2000, and our approach, respectively, at approximately the same bit-rates for *Lena* and *Barbara*. JPEG and JPEG2000 suffer from block and ringing artifacts, especially in areas of edges which can be seen for *Lena* in the hat structure and wisp of hair on her shoulder. Our SMoEs approach on the other hand is not block-based and the experts can steer along edges and represent them perfectly. It is worth noting how accurately the fine structures are represented in our method, even though our SMoE experts do not include any kind of advanced texture models yet. The same artifacts for JPEG and JPEG2000 are visible for *Barbara* on her arms. It can be noted that the SSIM value at this bit-rate for *Barbara* is better for JPEG2000 than for SMoE. However, our results are more visually appealing. This indicates that the SSIM metric may not necessarily be an appropriate tool to evaluate image qualities regarding the human visual perception for such structures.
Fig. 3 illustrates the corresponding gatings of our results for *Lena* and *Barbara* shown in Fig. 2. As expected, the proposed regularization promoting optimization results in a sparse separation of the image. In textured areas we achieve a dense concentration of kernels (i.e. in the feather for *Lena* and table cloth for *Barbara*) while less kernels in less structured areas. It is also clear that major information about the topology of the modeled image is embedded within the gating. As previously pointed out the steered kernels explore directional correlation properties in the image. While some kernels cater for small numbers of pixels some kernel spread out over the entire input space providing global support for thousands of pixels. As an example the green segment

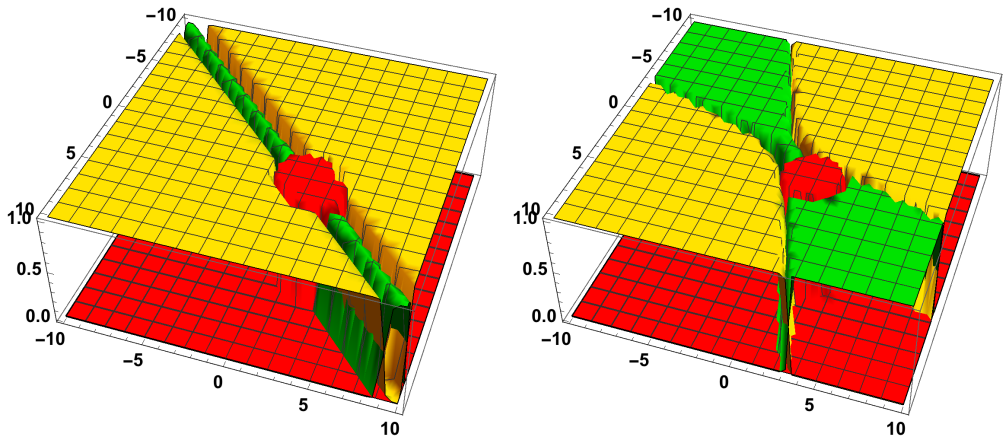


Figure 4: Examples of resulting gatings of merely three kernels. Their center positions are identical. Only a single parameter is changed between left and right.

highlighted in Fig. 3 is generated by one kernel and its associated steered gate covers an area almost from top to bottom of the image, while being overlapped and interrupted in the maximum influence by smaller kernels. Fig. 4 is intended to illustrate how gates can interact and provide complex steered and overlapping representations. Here two non-steered kernels and one steered kernel provide three separate gating functions. In Fig. 4 (left) the yellow gating function is separated by the green steered gate. Even though separated in the image plane, the yellow gate is, nevertheless, generated by only one kernel. The same holds for the steered green gate generated by its single kernel, separated by the red one. In Fig. 4 (right) the gates were generated from kernels with same parameters used in Fig. 4 (left), except that the bandwidth of the green kernels is made larger. A complex novel soft-gating pattern with curves results. It is the ability of SMoEs to model large sparse areas with few kernels that makes the approach very efficient. On the other hand complex patterns especially at edges and in textured areas can be reconstructed without edge-boundary artifacts.

4 Conclusion and Future Work

In this paper we proposed a Gradient Descent based optimization approach incorporating quantization of the model parameters for efficient coding within the SMoE framework. By maximizing the SSIM instead of minimizing the MSE the objective and perceived image quality drastically improves compared to our previous works as well as compared to JPEG and JPEG2000. The simultaneous consideration of optimizing and quantizing the model parameters allows for efficient bit allocation. Evaluations on color images show that our approach outperforms JPEG with bit-rate savings up to 42.48%. For mid- and high-range bit-rates it is also competitive to JPEG2000 in terms of SSIM while being visually more appealing. As SMoE is a novel approach to coding of images large potential for improvement is possible. This includes the incorporation of more complex experts being capable of modeling high frequency content and noise corruption as well as more advanced optimization strategies.

References

[1] R. Verhack, T. Sikora, L. Lange, G. V. Wallendael, and P. Lambert, "A universal image coding approach using sparse steered Mixture-of-Experts regression," in *2016 IEEE International Conference on Image Processing (ICIP)*, Sept 2016, pp. 2142–2146.

[2] M. Tok, R. Jongebloed, L. Lange, E. Bochinski, and T. Sikora, "An MSE Approach For Training And Coding Steered Mixtures Of Experts," in *2018 Picture Coding Symposium (PCS)*, June 2018, pp. 273–277.

[3] L. Lange, R. Verhack, and T. Sikora, "Video representation and coding using a sparse steered mixture-of-experts network," in *2016 Picture Coding Symposium (PCS)*, Dec 2016, pp. 1–5.

[4] R. Verhack, T. Sikora, L. Lange, R. Jongebloed, G. V. Wallendael, and P. Lambert, "Steered mixture-of-experts for light field coding, depth estimation, and processing," in *2017 IEEE International Conference on Multimedia and Expo (ICME)*, July 2017, pp. 1183–1188.

[5] V. Avramelos, I. Saenen, R. Verhack, G. V. Wallendael, P. Lambert, and T. Sikora, "Steered mixture-of-experts for light field video coding," pp. 10 752 – 10 752 – 12, 2018. [Online]. Available: https://doi.org/10.1117/12.2320563

[6] S. E. Yuksel, J. N. Wilson, and P. D. Gader, "Twenty Years of Mixture of Experts," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 8, pp. 1177–1193, 2012.

[7] E. Bochinski, R. Jongebloed, M. Tok, and T. Sikora, "Regularized Gradient Descent Training of Steered Mixture of Experts for Sparse Image Representation," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Oct 2018, pp. 3873–3877.

[8] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, April 2004.

[9] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference for Learning Representations*, 2015.

[10] M. Abadi, A. Agarwal *et al.*, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," 2015, software available from tensorflow.org. [Online]. Available: http://tensorflow.org/

[11] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering," *Trans. Img. Proc.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007. [Online]. Available: http://dx.doi.org/10.1109/TIP.2007.901238

[12] S. Yea and W. A. Pearlman, "Critical encoding rate in combined denoising and compression," in *IEEE International Conference on Image Processing 2005*, vol. 3, Sept 2005, pp. III–341.

[13] A. Norkin and N. Birkbeck, "Film Grain Synthesis for AV1 Video Codec," in *2018 Data Compression Conference*, March 2018, pp. 3–12.

[14] I. Hwang, J. Jeong, J. Choi, and Y. Choe, "Enhanced Film Grain Noise Removal for High Fidelity Video Coding," in *2013 International Conference on Information Science and Cloud Computing Companion*, Dec 2013, pp. 668–674.