

Fully Convolutional Model for Variable Bit Length and Lossy High Density Compression of Mammograms

Aupendu Kar, Sri Phani Krishna Karri, Nirmalya Ghosh, Debdoot Sheet*
 Department of Electrical Engineering, Indian Institute of Technology Kharagpur
 Kharagpur, West Bengal, India
 debdoot@ee.iitkgp.ac.in

Ramanathan Sethuraman
 Intel Technology India Pvt. Ltd.
 Bangalore, Karnataka, India
 ramanathan.sethuraman@intel.com

Abstract

Early works on medical image compression date to the 1980’s with the impetus on deployment of teleradiology systems for high-resolution digital X-ray detectors. Commercially deployed systems during the period could compress $4,096 \times 4,096$ sized images at 12 bpp to 2 bpp using lossless arithmetic coding, and over the years JPEG and JPEG2000 were imbibed reaching upto 0.1 bpp. Inspired by the reprise of deep learning based compression for natural images over the last two years, we propose a fully convolutional autoencoder for diagnostically relevant feature preserving lossy compression. This is followed by leveraging arithmetic coding for encapsulating high redundancy of features for further high-density code packing leading to variable bit length. We demonstrate performance on two different publicly available digital mammography datasets using peak signal-to-noise ratio (pSNR), structural similarity (SSIM) index and domain adaptability tests between datasets. At high density compression factors of $>300 \times$ (0.04 bpp), our approach rivals JPEG and JPEG2000 as evaluated through a Radiologist’s visual Turing test.

1. Introduction

While image and video compression for consumer grade cameras have been in use since early 1970’s, it was more than two decades later, with the advent of television and teleconferencing, that medical images started being compressed. Availability of digital X-ray detector, development of full-scale digital imaging systems, deployment of

*Work is supported under the Intel India Grand Challenge 2016 grant for Project MIRIAD.

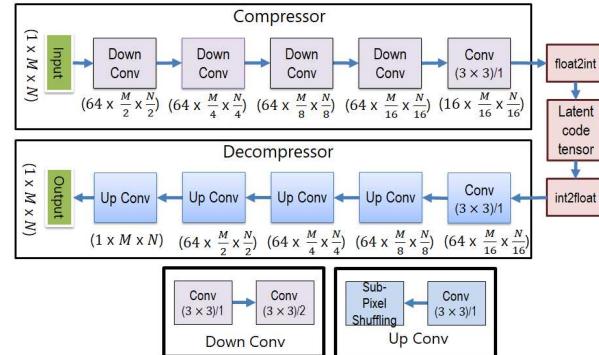


Figure 1. Overview of the convolutional autoencoder (CAE) with adaptive arithmetic encoding of the latent code tensor for variable bit length mammogram compression.

teleradiology networks for screening, life long archival of medical images for pathology modeling and personalized medicine were few of the factors that inspired early research on medical image compression in 1980’s. These were deployed a decade later [5] for addressing the challenges faced by clinical establishments. Hospitals typically accumulate about 4TB of imaging data per year, and with the growing trend towards personalized medicine necessitating life long archival, it inspires development of high-density medical image compression at factors $> 300 \times$ achieving < 0.05 bpp with no loss of visual features relevant for diagnosis. Our method in Fig. 1 is inspired by recent developments in deep learning based compression techniques for camera images [12, 2, 11, 6] and experimentally rivals JPEG [13] as well as the reigning standard JPEG2000 [9] at such demands of high compression factors, validated with experiments performed using X-ray mammograms.

The rest of the paper is organized as follows. Prior art

is discussed in Sec. 2. Sec. 3 describes the fully convolutional autoencoder (CAE) based compression engine for mammograms. The experiments performed for evaluation and benchmarking of performance of our approach vs. JPEG and JPEG2000 are described in Sec. 4 and characteristic aspects along with clinical usability study is discussed in Sec. 5, finally concluding the work in Sec. 6.

2. Related work

Information compression in digital media including images can typically be grouped into lossless or lossy. Entropy based methods, also known as arithmetic encoding techniques are popular for lossless compression [5] and on the other hand transformed domain compression using discrete cosine transform (DCT) based JPEG [13] and discrete wavelet transform (DWT) based JPEG2000 [9] are the reigning popular standards for lossy image compression [7]. Radiological image compression since 1980's has been lossless at its infancy [16] to avoid loss of any potential features of diagnostic relevance [5]. With gain in pixel-density of sensors and availability of high-resolution full-scale radiological scanners, JPEG and JPEG2000 have seen entry for radiological image compression, especially mammograms [3]. In view of the distortions typical to JPEG based compression [1], limiting its use for high-density image compression, recent developments in learning based approaches have been proposed to ensure ability to learn to preserve representative structures in images. Recent efforts for color image compression use recurrent convolutional neural network [12, 2]. Subsequently fully convolutional architectures without any recurrence [11] have also outperformed models with recurrence. Recent developments also include adversarial learning [6] to achieve visually smooth decompression of color images. We address the aspect of learning based medical image compression, especially in mammograms, to develop fully convolutional neural network based methods to overcome limitations of related prior-art [10].

3. Methodology

Our model for fully convolutional image compression consists of (a) compressor-decompressor blocks trained as a convolutional autoencoder, (b) adaptive arithmetic encoding for further lossless compression of the bit-length.

Compressor-Decompressor: The compressor is learned as the encoder unit and the decompressor as the decoder unit of a fully convolutional autoencoder. Fig. 1 shows the architecture of the autoencoder used here. The *compressor* consists of down-convolution (down-conv) blocks to extract key features and reduce the bits allocated for storage. The symmetrically shape matched up-convolution (up-conv) blocks in the *decompressor*

reconstruct the image from the compressed bitstream.

Compressor: Each down-conv block consists of a convolution layer with 3×3 kernel and stride of 1 followed by ReLU() activation, and subsequently a second convolution layer with 3×3 kernel and stride of 2 followed by ReLU() activation. Mammograms typically have bit-depths ranging 12/16 bpp and for our purpose they are range normalized in $[0, 1]$ represented in floating point tensors. The input image is processed through 4 stages of down-conv, followed by a convolution layer with 3×3 kernel and stride of 1, with ClippedReLU() activation function with clipping at 1.

Latent code tensor: This is generated using the *float2int()* operation defined as $g(x, y, c) = [(2^n - 1)i(x, y, c)]$ where $i(x, y, c) \in [0, 1]$ is the floating point value obtained from the compressor, $[.]$ is the integer rounding off operator, n is the bit-length of each element $g(x, y, c)$ in the latent code tensor, and x, y, c correspond to the spatial and channel index specifying location of the scalar in the compressed code tensor. The *int2float()* operator converts the integer valued latent code tensor to floating point value $j(x, y, c) = (2^n - 1)^{-1}g(x, y, c)$. These being non-differentiable $\nabla \text{float2int}()$ and $\nabla \text{int2float}()$ are approximated as 1.

Decompressor: The first convolution layer consists of 3×3 sized kernels with stride of 1 and ReLU() activation function, followed by 4 units of up-conv blocks. Each up-conv block consists of 3×3 sized convolution kernels with stride of 1 and ReLU() activation followed by sub-pixel shuffling [8]. The last up-conv block uses a ClippedReLU() with clipping at 1 as activation function instead of ReLU().

Adaptive arithmetic encoding: This phase comes into action only during deployment and not during training. Here the integer value latent code tensor representing the compressed version of the image is linearized into a 1-D array following either row-major or column-major representation, such that each element is a n-bit long integer. Hence a tensor of size $k \times m \times c$ would be a $kmcn$ bit-long representation. Subsequent to this, 8-bit long bit streams are extracted to have $kmcn/8$ codes that are compressed losslessly following entropy based adaptive arithmetic encoding [15]. This stage further compresses the bit-stream, on account of the high amount of spatial redundancy in mammograms to obtain high-density compression.

4. Experiments and Results

Dataset description: *CBIS-DDSM*¹ and *Dream*² are the two publicly available databases of digital mammogram images that are used for training and performance validation of the compression engines. Mammograms in *Dream* are encoded at 12 bpp and the ones in *CBIS-DDSM* are encoded

¹<https://wiki.cancerimagingarchive.net/display/Public/CBIS-DDSM/>

²<https://www.synapse.org/#/Synapse:syn4224222>

either as 16 bpp or 8 bpp. In experiments with *CBIS-DDSM* only 16 bpp are used since 8 bpp is typically not employed for digital mammography during acquisition. Out of the 3,102 mammograms in 16 bpp, a subset of 102 randomly selected ones are used for testing. In experiments with *Dream* 480 mammograms are used for training the model and 20 for testing.

Training: While training the *compressor-decompressor* jointly, 256×256 sized randomly located patches from the range normalized mammogram are used. A patch is included in training set only when $> 50\%$ pixels are non-zero valued, the mean intensity of the patch is not 0 or 1, and variance > 0 . In *Dream* we use 3,840 patches and in *CBIS-DDSM* we use 3,000 patches.

Training parameters: The Adam optimizer [4] with $\epsilon = 10^{-8}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ is used. All models are trained over 1,000 epochs with a learning rate of 10^{-4} and the batch size of 16. Mean squared error (MSE) is used as the loss function for back propagating error during training.

Baselines: We compare the performance of our approach against conventional methods like JPEG, JPEG2000. We also compare against a fully-connected autoencoder (FCAE) model inspired from [10] and optimized for high-density compression.

FCAE is trained using 16×16 sized patches collected following the inclusion principle used for CAE training. The *compressor* with $N = 16^2$ connects $N \rightarrow 8N \rightarrow 4N \rightarrow 2N$ hidden neuron with $\tanh()$ activation, and to N binary neurons representing the *latent code tensor*. The *decompressor* consists of $N \rightarrow 2N \rightarrow 4N \rightarrow 8N$ hidden neuron with $\tanh()$ activation, and to N neurons with ClippedReLU() activation function with clipping at 1 to obtain the decompressed patch sized 16×16 . During compression mammograms are divided into non-overlapping sequential blocks, and decompressor reorders it for retrieval.

Implementation: Both CAE and FCAE models are implemented with PyTorch on Python 2.7, accelerated with CUDA 9.0 on Nvidia Quadro P6000 with 24 GB DDR5 RAM on a PC with Intel Core i5 CPU and 28 GB of system RAM running Ubuntu 16.04 LTS. Average training time is ~ 30 sec / epoch. The compressor has 268,368 learnable parameters and decompressor has 454,724.

Results: Fig. 2 presents qualitative results while Fig. 3 shows the pSNR and SSIM [14] plots at different bpp corresponding to different compression factors on the two datasets used for different compression engines. While Figs. 3(a), 3(b) show the results of testing it on *CBIS-DDSM*, Figs. 3(c), 3(d) show tests on *Dream*. While the focus of high-density image compression is to be domain specific, we specifically evaluate performance of the method on *Dream* when trained with *CBIS-DDSM* and vice-versa as well, to observe interesting trends on their cross-dataset generalizability in the same domain.

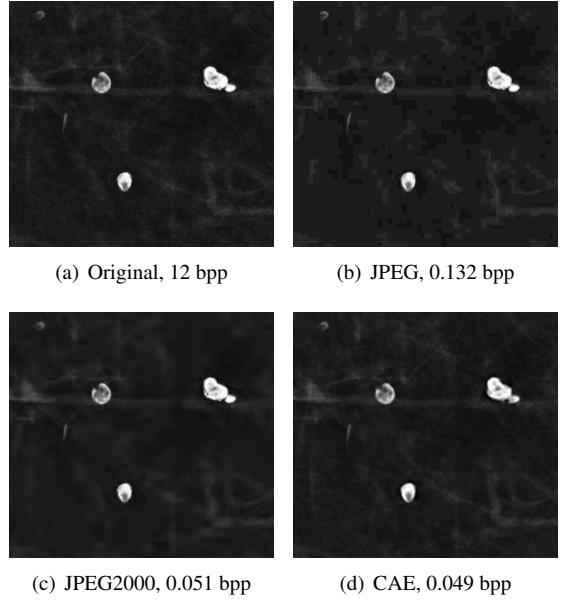


Figure 2. Calcified disc on a sample from *Dream*.

5. Discussions

Quantitative evaluation of the CAE based compression engine outperforms JPEG and JPEG2000 at higher compression factors yielding < 0.1 bpp in terms of both pSNR and SSIM. The fact that CAE has a relatively flat value of image quality over a wide dynamic range of bpp is noteworthy. The variation in bpp is brought in by varying n in the *latent code tensor*. It was observed that the values of $g(x, y, c)$ in the *latent code tensor* typically do not densely range over the complete range of $[0, 2^n - 1]$ but only over a smaller set of values occupying some $k << 2^n$ number of possible values; thus exhibiting a relatively similar value of entropy (H) until a value of n lower than H , e.g. $H = 2.95, 2.51, 2.44, 1.98$ at $n = 14, 10, 6, 2$ respectively. This corroborates with a relatively flat response in pSNR and SSIM for CAE, while those for FCAE, JPEG and JPEG2000 steadily decrease with lowering of bpp. Observing in Fig. 2 it is evident that a given SSIM mark of 0.80 can be achieved with FCAE at 0.101 bpp, CAE at 0.049 bpp, JPEG2000 at 0.051 bpp and JPEG at 0.132 bpp.

Visual Turing test (VTT) was performed where 5 Radiologist's were asked to visually inspect and identify the un-compressed image out of a pair where the other image was its corresponding compressed version. 10 mammograms were used in each test. With CAE 50% of images could be identified as uncompressed vs. 52% with JPEG2000.

6. Conclusion

We have proposed a fully convolutional approach for high-density compression of mammograms without loss of

