CS766 Proposal: Neural Image Compression

Megh Doshi
megh.doshi

Varun Sundar
vsundar4

Zachary Huemann
zhuemann

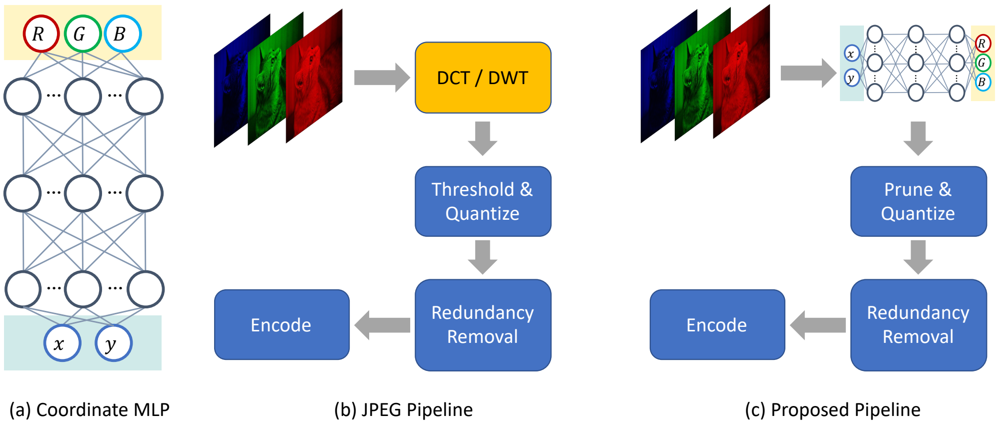
varun19299.github.io/implicit-image-compression

February 24, 2021

Implicit Neural Networks, being a continuous mapping, can serve as a compelling choice for representing a variety of commonly encountered 2D and 3D signals. In this proposal, we specifically consider the task of image compression via implicit networks. Unfortunately, owing to the over-parameterized nature of deep networks, a naive approach may require more parameters than samples present in the original signal. Furthermore, the capacity of such networks often saturates with increasing width or depth. In this proposal, we seek to explore two related directions: (a) efficiently increasing the capacity of implicit MLPs to fit natural images, and (b) reducing the storage requirement of such networks through a combination of structured hashing, quantization and entropy coding.

1 Introduction

A large proportion of recent success in a variety of computer vision (and graphics) problems has been attributed to implicitly defined representations parameterized by neural networks (typically a MLP). These include works on novel view-point rendering (Mildenhall et al., 2020; Martin-Brualla et al., 2020), image stabililization (Liu et al., 2021) and view-consistent image generation Schwarz et al. (2020); Chan et al. (2020). Such MLPs replace traditional grid-based representations, and map low-dimensional coordinates to output quantities such as pixel intensities or densities. Their inherent continuous and differentiable nature makes these representations a compelling choice. Additionally, in the particular case of 3D points, such networks are often much more compact than grid-based representations. Following Tancik et al. (2020), we shall refer to such neural networks as "coordinate MLPs".



(a) Coordinate MLP

(b) JPEG Pipeline

(c) Proposed Pipeline

Figure 1: Can coordinate MLPs efficiently represent 2D image signals? We propose to use a pipeline similar to JPEG with two major differences: (a) instead of a discrete cosine or wavelet transform (DCT or DWT), we use a multilayer MLP (b) we efficiently store the network's weights as opposed to corresponding DCT or DWT coefficients.

In this proposal, we examine if these benefits can be carried over to the simpler 2D case of images. We consider applying coordinate MLPs for the task of lossy image compression by mapping 2D grid-locations $(x, y) \in [0, 1]^2$ to RGB intensities. By fitting a MLP, we transfer the task of compressing a grid of pixels to compressing the corresponding network's weights. The representation is no longer limited by the grid-resolution but by the underlying network architecture. This however can be challenging since deep networks often have more parameters than data points itself. However, a wide body of research addresses the efficient storage and inference of deep networks, although generally targeted towards high-dimensional mapping tasks such as image classification.

Another challenge associated with coordinate MLPs is their diminishing increase in capacity with growing layer width and depth. This makes representing signals which are densely sampled (large resolution) or with finer detail difficult. Rebain et al. (2020) tackle this issue for the case of 3D points by decomposing the scene into soft Voronoi diagrams and dedicating smaller networks for each part. For images, frequency domain and wavelet decompositions are potential candidates to achieve similar workarounds. In summary, we focus on the following questions:

- Given a target image to fit, how can we train and efficiently store coordinate MLPs? Since image quality is usually sacrificed for storage space, we are interested in exploring the trade-off space for coordinate MLPs and comparing them to conventionally used image compression algorithms such as JPEG.
- For a fixed number of network parameters, can image decomposition help overcome the diminishing returns of naively scaling coordinate MLPs? Valuable insights could include understanding when such decompositions are useful and the range of image resolutions that can be represented.

2 Methodology

Network Architecture We adopt SIREN (Sitzmann et al., 2020) as our base MLP, which utilizes sinusoidal activation functions with a specialized initialization scheme. SIREN has been shown to be more effective than standard ReLU MLPs in learning high-frequency detail. An additional benefit is that derivatives of SIREN with respect to inputs and weights are well-defined upto any order, making it convenient for tasks providing supervision based on gradient information. Specifically, to fit a 512×512 image, we set the layer width as 256 and depth as 4. The number of parameters here is $256 * (2 + 256 * 2 + 3) = 132{,}352$, and if stored as `float32`, requires $4{,}235{,}264$ bits or 517 kilobytes (kB). In comparison, the `int8` RAW image requires $512 * 512 * 3 * 8 = 6{,}291{,}456$ bits or 6 MB.

Increasing Representation Capacity Wavelet / Pyramid Decomposition Fit the implicit MLP on the smaller low-frequency component (e.g. top-left corner in a Haar wavelet decomposition) and depending on the characteristics of residuals, map them to either a low-dimensional problem or learn them in a coordinate-value manner. Wavelets are better at representing features in an image as they encompass the frequency content and the spatial information in an image. Pyramid Decomposition analyzes and extracts features from an image under different sampling rates, hence, giving us access to the global and local features. For lossless compression, we additionally seek to store the residual of the decoded image and the original groundtruth.

Reducing Parameter Storage A combination of pruning, quantization and entropy coding can reduce parameter storage of implicit MLPs by around $20{-}100\times$ their original size. Iterative pruning (Zhu and Gupta, 2018) removes weights with the least magnitude in a gradual but decaying manner. Structured hashing (Eban et al., 2020) represents the weights of a neural network by mapping the unrolled weights to a low-rank representation. Removing unimportant weights makes the network more amenable to quantization from `float32` to `int8`: which could be linear, range-based or cluster-based. Similar to Han et al. (2015), the final step in storing network weights involves some form of entropy coding. Note that most of the enlisted techniques have conventionally been developed for high-dimensional domains such as image classification, segmentation and object detection. Here, we deal with low-dimensional domains, which can be significantly more challenging.

3 Related Work

Conventional Image Compression JPEG (Wallace, 1992) is the most commonly used lossy image compression technique even today, 28 years after its initial release. It exploits the sparse nature of natural images in the frequency domain via discrete cosine transforms (DCT). JPEG 2000 (Rabbani, 2002) further builds on this, replacing block-wise DCT transforms by Haar wavelet based discrete wavelet transforms (DWT), resulting in lesser block artefacts at higher compression ratios. However, both JPEG and JPEG 2000 are linear transform coding techniques, representing local features with localized linear basis functions. Consequently, extreme quantization can lead to distortion in the synthesis transformation, giving rise to prominent visual artefacts. In the lossless regime, PNG (Boutell and Lane, 1997) and TIFF (Parsons and Rafferty, 2002) are among the popular compression algorithms used today.

Deep Learning based Image Compression Toderici et al. (2016) propose a recurrent neural network (RNN) based backbone in their pioneering work for variable bit-rate, end-to-end compression. To improve compression ratios, several prior work propose a surrogate for quantization with either uniform noise (Ballé et al., 2016), rounding in direction of gradient or soft-to-hard vector quantization (Agustsson et al., 2017). Autoencoder based methods have also been used in the past for compression (Hinton and Salakhutdinov, 2006), although only recently did (Toderici et al., 2017) demonstrate feasibility for resolutions larger than a thumbnail ($> 64 \times 64$ pixels).

The above-mentioned techniques rely on either convolutional neural networks (CNNs) or RNNs to remove spatial redundancy from images and extract a compact latent representation. Prior to the deep learning renaissance of the 2010s, MLPs have also been used to achieve vector quantization, where the hidden layer of the network is treated as the latent representation. While these works utilize a network to transform the input image into a compressible embedding, we instead choose to represent the image by the *network itself*. Additionally, our method is single-shot and does not require a dataset of images to work with.

Compressing Weights of a Deep Network Of the large body of literature in network compression, most pertinent to our application are pruning, low-rank hashing and quantization. Pruning removes network connections according to an importance criterion (magnitude, gradient, etc.), trading off accuracy to reach a desired parameter count. This can be done either at the end of training (LeCun et al., 2017) or by pruning iteratively at regular intervals (Zhu and Gupta, 2018). Quantization can be performed linearly or range-aware, with the former often trained with surrogate quantization to prevent loss of performance. For a more comprehensive overview, we refer readers to the excellent review paper of Cheng et al. (2017).

Implicit or Coordinate MLPs for Image Representation Finally, there have been a few methods utilizing coordinate MLPs for image representation, although not targeted towards compression. Tancik et al. (2020) proposes embeddings based on random Fourier features to fit high-frequency detail better. Sitzmann et al. (2020) instead uses sinusoidal activation functions to the same effect while also demonstrating more general applicability of differentiable mappings. On the small 32×32, images of CIFAR-10 (Alex Krizhevsky, 2009), Bricman and Ionescu (2018) used ReLU based coordinate-MLPs for image denoising, super-resolution and storage.

As noted above, our approach is fundamentally different from both conventional compression techniques and recent deep learning techniques. If efficient at representing images, it can potentially lead to a host of advantages continuous mappings offer.

4 Experimentation Details

Comparing Performance We benchmark the proposed method against JPEG (and its variants) using the peak signal-to-noise ratio (PSNR) over a range of compression rates. Compression rate is defined as the bits required to represent the original grid of pixels image compared to its encoded variant. Since PSNR is biased towards low-frequency components and does not accurately capture perceptual quality, we also propose to evaluate using MS-SSIM (Zhao et al., 2016) and the recently proposed LPIPS as metrics (Zhang et al., 2018). At this point, we are mainly interested in showing that coordinate MLPs can be good candidate algorithms for image compression and hence shall not compare against more recent state-of-the-art image compression techniques.

Image Collections A popular choice to benchmark compression algorithms has been the Kodak Photo CD (Franzen, 1999), comprising of 24 768×512 images. We also consider the more recent CVPR Workshop and Challenge on Learned Image Compression (CLI) and Image Compression Benchmark (ima). Additionally, we propose to examine domains outside natural images challenging to JPEG, but can be compressed more efficiently by our pipeline. For instance, text-based images and scientific captures, which may not necessarily adhere to natural image statistics.

Table 1: Intended Timeline. Midterm report is due at the end of week 4, and the final presentation after week 7.

Time	Task
Week 1	Setup SIREN and evaluate its capacity
Week 2	Test different pruning and quantization techniques
Week 3	SIREN with wavelet decomposition
Week 4	Buffer time for debugging
Week 5	Survey other target image domains
Week 6 Week 7	Test and understand strengths and limitations of proposed method

References

CLIC challenge on learnt image compression. `http://compression.cc`. Accessed: February 24, 2021.

Image Compression Benchmark. `https://imagecompression.info`. Accessed: February 24, 2021.

E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, and L. Van Gool. Soft-to-hard vector quantization for end-to-end learning compressible representations. 2017.

G. H. Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

J. Ballé, V. Laparra, and E. P. Simoncelli. End-to-end optimized image compression. In *Proceedings of the International Conference on Learning Representations (ICLR)*, April 2016.

T. Boutell and T. Lane. Png (portable network graphics) specification version 1.0. *Network Working Group*, pages 1–102, 1997.

P. A. Bricman and R. T. Ionescu. Coconet: A deep neural network for mapping pixel coordinates to color values. In *International Conference on Neural Information Processing*, pages 64–76. Springer, 2018.

E. Chan, M. Monteiro, P. Kellnhofer, J. Wu, and G. Wetzstein. pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *arXiv*, 2020.

Y. Cheng, D. Wang, P. Zhou, and T. Zhang. A survey of model compression and acceleration for deep neural networks, 2017.

E. Eban, Y. Movshovitz-Attias, H. Wu, M. Sandler, A. Poon, Y. Idelbayev, and M. A. Carreira-Perpinan. Structured multi-hashing for model compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

R. Franzen. Kodak lossless true color image suite. 1999.

S. Han, H. Mao, and W. J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. In *Proceedings of the International Conference on Learning Representations (ICLR)*, April 2015.

G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786): 504–507, 2006.

Y. LeCun, J. S. Denker, S. A. Solla, R. E. Howard, and L. D. Jackel. Optimal brain damage. In *Advances in Neural Information Processing Systems (NeurIPS)*, December 2017.

Y.-L. Liu, W.-S. Lai, M.-H. Yang, Y.-Y. Chuang, and J.-B. Huang. Neural re-rendering for full-frame video stabilization. In *arXiv*, 2021.

R. Martin-Brualla, N. Radwan, M. S. Sajjadi, J. T. Barron, A. Dosovitskiy, and D. Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. 2020.

B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2020.

G. Parsons and J. Rafferty. Rfc3302: Tag image file format (tiff)-image/tiff mime sub-type registration, 2002.

M. Rabbani. Jpeg2000: Image compression fundamentals, standards and practice. *Journal of Electronic Imaging*, 11 (2):286, 2002.

D. Rebain, W. Jiang, S. Yazdani, K. Li, K. M. Yi, and A. Tagliasacchi. Derf: Decomposed radiance fields. 2020.

K. Schwarz, Y. Liao, M. Niemeyer, and A. Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

V. Sitzmann, J. N. Martel, A. W. Bergman, D. B. Lindell, and G. Wetzstein. Implicit neural representations with periodic activation functions. 2020.

M. Tancik, P. P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. T. Barron, and R. Ng. Fourier features let networks learn high frequency functions in low dimensional domains. 2020.

G. Toderici, S. M. O'Malley, S. J. Hwang, D. Vincent, D. Minnen, S. Baluja, M. Covell, and R. Sukthankar. Variable rate image compression with recurrent neural networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, April 2016.

G. Toderici, D. Vincent, N. Johnston, S. Jin Hwang, D. Minnen, J. Shor, and M. Covell. Full resolution image compression with recurrent neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2017.

G. K. Wallace. The jpeg still picture compression standard. *IEEE transactions on consumer electronics*, 38(1):xviii–xxxiv, 1992.

R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

H. Zhao, O. Gallo, I. Frosio, and J. Kautz. Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging*, 3(1):47–57, 2016.

M. Zhu and S. Gupta. To prune, or not to prune: Exploring the efficacy of pruning for model compression. In *Proceedings of the International Conference on Learning Representations (ICLR)*, April 2018.