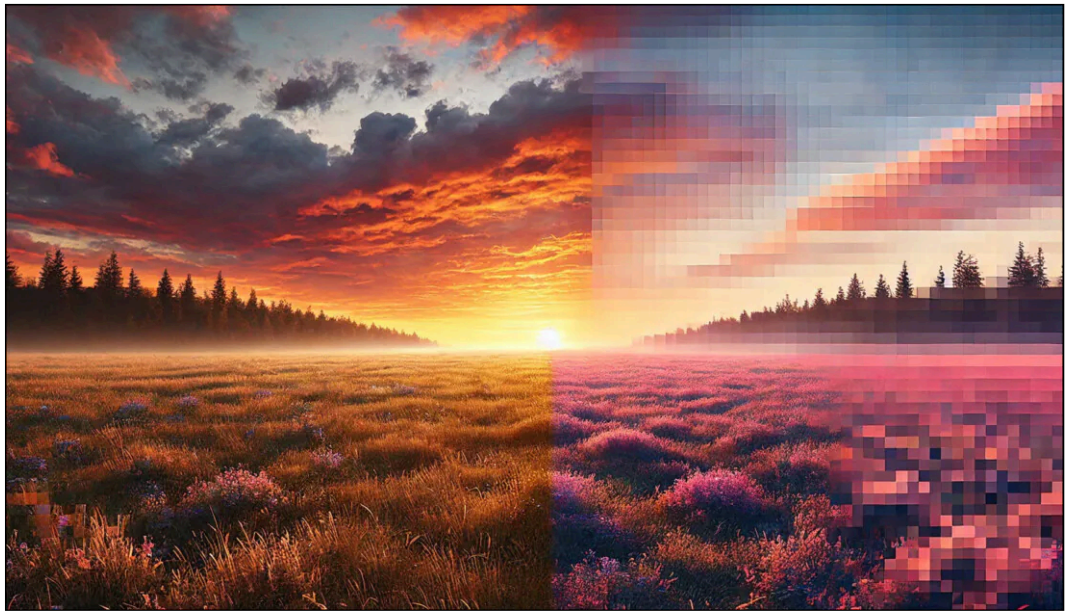
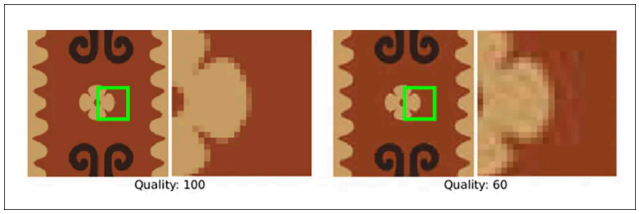
Using JPEG Compression to Improve Neural Network Training

Originally published at https://www.unite.ai/using-jpeg-compression-to-improve-neural-network-training/



A new research paper from Canada has proposed a framework that deliberately introduces JPEG compression into the training scheme of a neural network, and manages to obtain better results – and better resistance to adversarial attacks.

This is a fairly radical idea, since the current general wisdom is that JPEG artifacts, which are optimized for human viewing, and not for machine learning, generally have a deleterious effect on neural networks trained on JPEG data.



An example of the difference in clarity between JPEG images compressed at different loss values (higher loss permits a smaller file size, at the expense of delineation and banding across color gradients, among other types of artifact). Source: https://forums.jetphotos.com/forum/aviation-photography-videography-forums/digital-photo-processing-forum/1131923-how-to-fix-jpg-compression-artefacts?p=1131937#post1131937

A 2022 report from the University of Maryland and Facebook AI asserted that JPEG compression 'incurs a significant performance penalty' in the training of neural networks, in spite of previous work that claimed neural networks are relatively resilient to image compression artefacts.

A year prior to this, a new strand of thought had emerged in the literature: that JPEG compression could actually be leveraged for improved results in model training.

However, though the authors of that paper were able to obtain improved results in the training of JPEG images of varying quality levels, the model they proposed was so complex and burdensome that it was not practicable. Additionally, the system's use of default JPEG optimization settings (quantization) proved a barrier to training efficacy.

A later project (2023's *JPEG Compliant Compression for DNN Vision*) experimented with a system that obtained slightly better results from JPEG-compressed training images with the use of a frozen deep neural network (DNN) model. However, freezing parts of a model during training tends to reduce the versatility of the model, as well as its broader resilience to novel data.

JPEG-DL

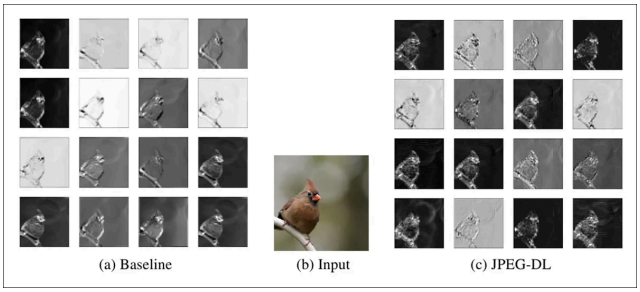
Instead, the new work, titled *JPEG Inspired Deep Learning*, offers a much simpler architecture, which can even be imposed upon existing models.

The researchers, from the University of Waterloo, state:

> *'Results show that JPEG-DL significantly and consistently outperforms the standard DL across various DNN architectures, with a negligible increase in model complexity.*

> *Specifically, JPEG-DL improves classification accuracy by up to 20.9% on some fine-grained classification dataset, while adding only 128 trainable parameters to the DL pipeline. Moreover, the superiority of JPEG-DL over the standard DL is further demonstrated by the enhanced adversarial robustness of the learned models and reduced file sizes of the input images.'*

The authors contend that an optimal JPEG compression quality level can help a neural network distinguish the central subject/s of an image. In the example below, we see baseline results (left) blending the bird into the background when features are obtained by the neural network. In contrast, JPEG-DL (right) succeeds in distinguishing and delineating the subject of the photo.



Tests against baseline methods for JPEG-DL. Source: https://arxiv.org/pdf/2410.07081

'This phenomenon,' they explain, *'termed "compression helps" in the [2021] paper, is justified by the fact that compression can remove noise and disturbing background features, thereby highlighting the main object in an image, which helps DNNs make better prediction.'*

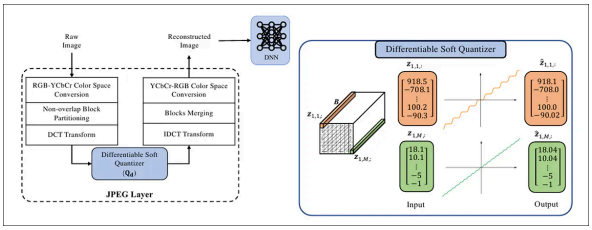
Method

JPEG-DL introduces a differentiable soft quantizer, which replaces the non-differentiable quantization operation in a standard JPEG optimization routine.

This allows for gradient-based optimization of the images. This is not possible in conventional JPEG encoding, which uses a *uniform quantizer* with a rounding operation that approximates the nearest coefficient.

The differentiability of JPEG-DL's schema permits joint optimization of both the training model's parameters and the JPEG quantization (compression level). Joint optimization means that both the model and the training data are accommodated to each other in the end-to-end process, and no freezing of layers is needed.

Essentially, the system customizes the JPEG compression of a (raw) dataset to fit the logic of the generalization process.



Conceptual schema for JPEG-DL.

One might assume that raw data would be the ideal fodder for training; after all, images are completely decompressed into an appropriate full-length color space when they are run in batches; so what difference does the original format make?

Well, since JPEG compression is optimized for human viewing, it throws areas of detail or color away in a manner concordant with this aim. Given a picture of a lake under a blue sky, increased levels of compression will be applied to the sky, because it contains no 'essential' detail.

On the other hand, a neural network lacks the eccentric filters which allow us to zero in on central subjects. Instead, it is likely to consider any banding artefacts in the sky as valid data to be assimilated into its latent space.

Though a human will dismiss the banding in the sky, in a heavily compressed image (left), a neural network has no idea that this content should be thrown away, and will need a higher-quality image (right). Source: https://lensvid.com/post-processing/fix-jpeg-artifacts-in-photoshop/

Therefore, one level of JPEG compression is unlikely to suit the entire contents of a training dataset, unless it represents a very specific domain. Pictures of crowds will require much less compression than a narrow-focus picture of a bird, for instance.

The authors observe that those unfamiliar with the challenges of quantization, but who are familiar with the basics of the transformers architecture, can consider these processes as an 'attention operation', broadly.

Data and Tests

JPEG-DL was evaluated against transformer-based architectures and convolutional neural networks (CNNs). Architectures used were EfficientFormer-L1; ResNet; VGG; MobileNet; and ShuffleNet.

The ResNet versions used were specific to the CIFAR dataset: ResNet32, ResNet56, and ResNet110. VGG8 and VGG13 were chosen for the VGG-based tests.

For CNN, the training methodology was derived from the 2020 work *Contrastive Representation Distillation* (CRD). For EfficientFormer-L1 (transformer-based), the training method from the 2023 outing *Initializing Models with Larger Ones* was used.

For fine-grained tasks featured in the tests, four datasets were used: Stanford Dogs; the University of Oxford's Flowers; CUB-200-2011 (CalTech Birds); and Pets (*'Cats and Dogs'*, a collaboration between the University of Oxford and Hyderabad in India).

For fine-grained tasks on CNNs, the authors used PreAct ResNet-18 and DenseNet-BC. For EfficientFormer-L1, the methodology outlined in the aforementioned *Initializing Models With Larger Ones* was used.

Across the CIFAR-100 and fine-grained tasks, the varying magnitudes of Discrete Cosine Transform (DCT) frequencies in the JPEG compression approach was handled with the Adam optimizer, in order to adapt the learning rate for the JPEG layer across the models that were tested.

In tests on ImageNet-1K, across all experiments, the authors used PyTorch, with SqueezeNet, ResNet-18 and ResNet-34 as the core models.

For the JPEG-layer optimization evaluation, the researchers used Stochastic Gradient Descent (SGD) instead of Adam, for more stable performance. However, for the ImageNet-1K tests, the method from the 2019 paper *Learned Step Size Quantization* was employed.

Method	Res32	Res56	Res110	VGG8	VGG13	MobileNetV2	ShuffleNetV2
Baseline	71.14	72.34	73.79	70.36	73.77	64.6	71.82
JPEG-DL	**71.92**$_{\pm 0.31}$ (+0.78)	**73.39**$_{\pm 0.19}$ (+1.05)	**74.46**$_{\pm 0.11}$ (+0.67)	**71.10**$_{\pm 0.41}$ (+0.74)	**75.32**$_{\pm 0.10}$ (+1.55)	**65.91**$_{\pm 0.11}$ (+1.31)	**73.04**$_{\pm 0.16}$ (+1.22)

Model	Method	CUB-200	Dogs	Flowers	Pets
ResNet-18	Baseline	54.00$_{\pm 1.43}$	63.71$_{\pm 0.32}$	57.13$_{\pm 1.28}$	70.37$_{\pm 0.84}$
	JPEG-DL	**58.81**$_{\pm 0.12}$ (+4.81)	**65.57**$_{\pm 0.37}$ (+1.86)	**68.76**$_{\pm 0.57}$ (+11.63)	**74.84**$_{\pm 0.66}$ (+4.47)
DenseNet-121	Baseline	57.70$_{\pm 0.44}$	66.61$_{\pm 0.17}$	51.32$_{\pm 0.57}$	70.26$_{\pm 0.79}$
	JPEG-DL	**61.32**$_{\pm 0.43}$ (+3.62)	**69.67**$_{\pm 0.58}$ (+3.06)	**72.22**$_{\pm 1.05}$ (+20.90)	**75.90**$_{\pm 0.68}$ (+5.64)

Above the top-1 validation accuracy for the baseline vs. JPEG-DL on CIFAR-100, with standard and mean deviations averaged over three runs. Below, the top-1 validation accuracy on diverse fine-grained image classification tasks, across various model architectures, again, averaged from three passes.

Commenting on the initial round of results illustrated above, the authors state:

> *'Across all seven tested models for CIFAR-100, JPEG-DL consistently provides improvements, with gains of up to 1.53% in top-1 accuracy. In the fine-grained tasks, JPEG-DL offers a substantial performance increase, with improvements of up to 20.90% across all datasets using two different models.'*

Results for the ImageNet-1K tests are shown below:

Method	SqueezeNetV1.1	Resnet18	Resnet34
Baseline	57.95	69.75	73.31
JPEG-DL	**58.26** (+0.31)	**70.13** (+0.38)	**73.54** (+0.23)

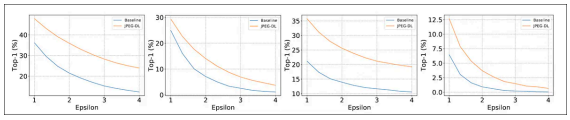
Top-1 validation accuracy results on ImageNet across diverse frameworks.

Here the paper states:

> *'With a trivial increase in complexity (adding 128 parameters), JPEG-DL achieves a gain of 0.31% in top-1 accuracy for SqueezeNetV1.1 compared to the baseline using a single round of [quantization] operation.*
>
> *'By increasing the number of quantization rounds to five, we observe an additional improvement of 0.20%, leading to a total gain of 0.51% over the baseline.'*

The researchers also tested the system using data compromised by the adversarial attack approaches Fast Gradient Signed Method (FGSM) and Projected Gradient Descent (PGD).

The attacks were conducted on CIFAR-100 across two of the models:

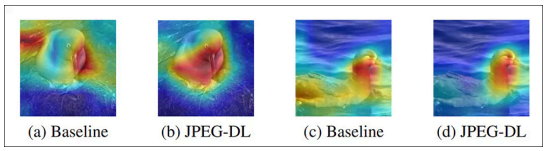


Testing results for JPEG-DL, against two standard adversarial attack frameworks.

The authors state:

'[The] JPEG-DL models significantly improve the adversarial robustness compared to the standard DNN models, with improvements of up to 15% for FGSM and 6% for PGD.'

Additionally, as illustrated earlier in the article, the authors conducted a comparison of extracted feature maps using GradCAM++ – a framework that can highlight extracted features in a visual manner.



A GradCAM++ illustration for baseline and JPEG-DL image classification, with extracted features highlighted.

The paper notes that JPEG-DL produces an improved result, and that in one instance it was even able to classify an image that the baseline failed to identify. Regarding the earlier-illustrated image featuring birds, the authors state:

'[It] is evident that the feature maps from the JPEG-DL model show significantly better contrast between the foreground information (the bird) and the background compared to the feature maps generated by the baseline model.

'Specifically, the foreground object in the JPEG-DL feature maps is enclosed within a well-defined contour, making it visually distinguishable from the background.

'In contrast, the baseline model's feature maps show a more blended structure, where the foreground contains higher energy in low frequencies, causing it to blend more smoothly with the background.'

Conclusion

JPEG-DL is intended for use in situations where raw data is available – but it would be most interesting to see if some of the principles featured in this project could be applied to conventional dataset training, wherein the content may be of lower quality (as frequently occurs with hyperscale datasets scraped from the internet).

As it stands, that largely remains an annotation problem, though it has been addressed in traffic-based image recognition, and elsewhere.

First published Thursday, October 10, 2024

Writer on machine learning, domain specialist in human image synthesis. Former head of research content at Metaphysic.ai, until its dissolution into DNEG's Brahma.ai.
Portfolio site: martinanderson.ai
Contact: martin@martinanderson.ai