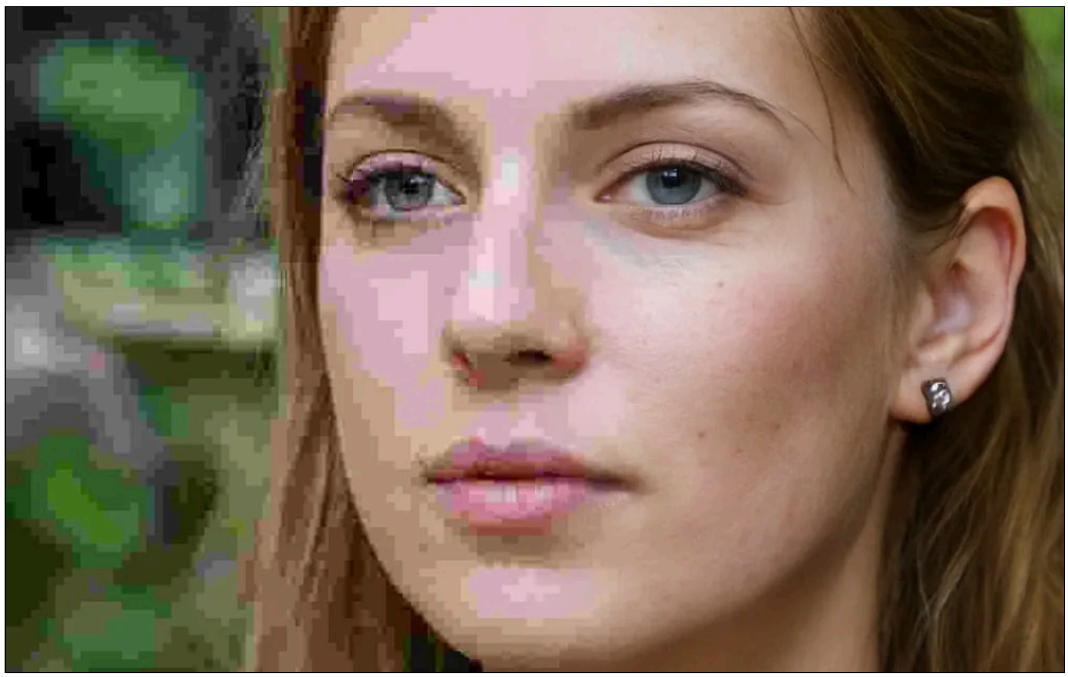
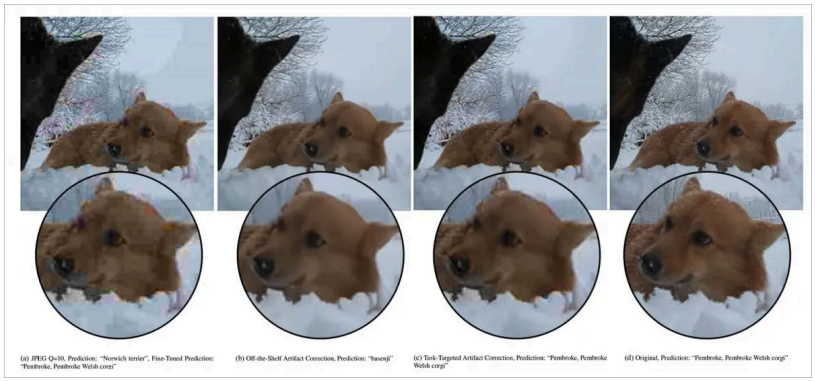
Solving the JPEG Artifact Problem in Computer Vision Datasets

Originally published at https://www.unite.ai/solving-the-jpeg-artifact-problem-in-computer-vision-datasets/



A new study from the University of Maryland and Facebook AI has found a 'significant performance penalty' for deep learning systems that use highly compressed JPEG images in their datasets, and offers some new methods to mitigate the effects of this.

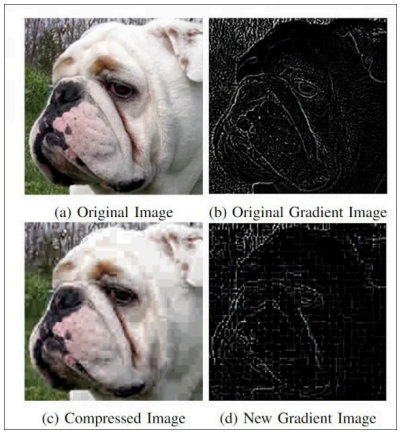
The report, titled *Analyzing and Mitigating JPEG Compression Defects in Deep Learning*, claims to be 'significantly more comprehensive' than previous studies into the effects of artifacts in training computer vision datasets. The paper finds that '[heavy] to moderate JPEG compression incurs a significant performance penalty on standard metrics', and that neural networks are perhaps not as resilient to such perturbations as previous work suggests.



A photo of a dog from the 2018 MobileNetV2 dataset. At quality 10 (left), a classification system fails to identify the correct breed 'Pembroke Welsh Corgi', instead system already knows this is a photo of a dog, but not the breed); second from left, an off-the-shelf JPEG artifact-corrected version of the image again fails to iden from right, targeted artifact correction restores the correct classification; and right, the original photo, correctly classified. Source: https://arxiv.org/pdf/2011.08932.p

Compression Artifacts as 'Data'

Extreme JPEG compression is likely to create visible or semi-visible borders around the 8×8 blocks out of which a JPEG is assembled into a pixel grid. Once these blocking or 'ringing' artifacts appear, they are likely to be misinterpreted by machine learning systems as real world elements of the subject of the image, unless some compensation is made for this.



Above, a computer vision machine learning system extracts a 'clean' gradient image from a good-quality picture. Below, 'blocking' artifacts in a lower-quality save of the image obscure the features of the subject, and may end up 'infecting' the features derived from an image set, particularly in cases where high and low quality images occur in the dataset, such as in web-scraped collections to which only generic data cleaning has been applied. Source: http://www.cs.utep.edu/ofuentes/papers/quijasfuentes2014.pdf

As seen in the first image above, such artifacts can affect image classification tasks, with implications also for text-recognition algorithms, which may fail to correctly identify artifact-affected characters.

In the case of image synthesis training systems (such as deepfake software or GAN-based image generation systems), a 'rogue' block of low-quality, highly compressed images in a dataset may either drag the median quality of reproduction down, or else be subsumed and essentially overridden by a greater number of higher quality features extracted from better images in the set. In either case, better data is desirable – or, at least, consistent data.

JPEG – Usually 'Good Enough'

JPEG compression is an irreversibly lossy codec that can be applied to various image formats, though it's primarily applied to the JFIF image file wrapper. Despite this, the JPEG (.jpg) format was named after its associated compression method, and not the JFIF wrapper for the image data.

Entire machine learning architectures have arisen in recent years that include JPEG-style artifact mitigation as part of AI-driven upscaling/restoring routines, and AI-based compression artifact removal is now incorporated into a number of commercial products, such as the Topaz image/video suite, and the neural features of recent versions of Adobe Photoshop.

Since the 1986 JPEG schema currently in common use was pretty much locked down in the early 1990s, it's not possible to add metadata to an image that would indicate which quality level (1-100) a JPEG image was saved at – at least, not without modifying over thirty years of legacy consumer, professional and academic software systems that weren't expecting such metadata to be available.

Consequently, it's not uncommon to tailor machine learning training routines to the evaluated or known quality of JPEG image data, as the researchers have done for the new paper (see below). In the absence of a 'quality' metadata entry, it's currently necessary to either know the details of how the image was compressed (i.e. compressed from a lossless source), or estimate the quality through perceptual algorithms or manual classification.

An Economical Compromise

JPEG is not the only lossy compression method that can affect the quality of machine learning datasets; compression settings in PDF files can also discard information in this way, and be set to very low quality levels in order to save disk space for local or network archival purposes.

This can be witnessed by sampling various PDFs across archive.org, some of which have been compressed so heavily as to be a notable challenge for image or text recognition systems. In many cases, such as copyrighted books, this intense compression seems to have been applied as a form of cheap DRM, in much the same way copyright holders may choose to lower the resolution of user-uploaded YouTube videos on which they hold the IP, leaving the 'blocky' videos as promotional tokens to inspire 'full res' purchases, rather than having them deleted.

In many other cases, the resolution or image quality is low simply because the data is very old, and hails from an era when local and network storage was more expensive, and when limited network speeds favored highly-optimized and portable images over high quality reproduction.

It's been argued that JPEG, while not the best solution *now*, has been 'enshrined' as irremovable legacy infrastructure that's essentially intertwined with the foundations of the internet.

Legacy Burden

Though later innovations such as JPEG 2000, PNG and (most recently) the.webp format offer superior quality, re-sampling older, highly popular machine learning datasets would arguably 'reset' the continuity and history of year-on-year computer vision challenges in the academic community – an impediment that would apply also in the case of resaving PNG dataset images at higher quality settings. This could be considered as a kind of technical debt.

While venerable server-driven image processing libraries such as ImageMagick support better formats, including.webp, image transformation requirements frequently occur in legacy systems that are not set up for anything other than JPG or PNG (which offers lossless compression, but at the expense of latency and disk space). Even WordPress, the CMS powering nearly 40% of all websites, only added.webp support three months ago.

PNG was a late (arguably too late) entry into the image format sector, arising as an open source solution in the latter part of the 1990s in response to a 1995 declaration by Unisys and CompuServe that royalties would henceforth be payable on the LZW compression format used in GIF files, which were commonly used back then for logos and flat-color elements, even if the format's resurrection in the early 2010s centered on its ability to provide low-bandwidth, snappy animated content (ironically, animated PNGs never gained popularity or wide support, and were even banned from Twitter in 2019).

Despite its shortcomings, JPEG compression is quick, space-efficient, and deeply embedded in systems of all types – and therefore not likely to disappear entirely from the machine learning scene in the near future.

Making the Best of the AI/JPEG Detente

To an extent, the machine learning community has accommodated itself to the foibles of JPEG compression: in 2011 the European Society of Radiology (ESR) published a study on the 'Usability of irreversible image compression in radiological imaging', providing guidelines for 'acceptable' loss; when the venerable MNIST text recognition dataset (whose image data was originally supplied in a novel binary format) was ported to a 'regular' image format, JPEG, not PNG, was chosen; and an earlier (2020) collaboration from the new paper's authors offered 'a novel architecture' for calibrating machine learning systems to the shortcomings of varying JPEG image quality, without the need for models to be trained at each JPEG quality setting – a feature utilized in the new work.

Indeed, research into the utility of quality-variant JPEG data is a relatively thriving field in machine learning. One (unrelated) 2016 project from the Center for Automation Research at the University of Maryland, actually centers on the DCT domain (where JPEG artifacts occur at low quality settings) as a route to deep feature extraction; another project from 2019 concentrates on byte-level reading of JPEG data without the time-consuming necessity to actually decompress the images (i.e. open them at some point in an automated workflow); and a study from France in 2019 actively leverages JPEG compression in service of object recognition routines.

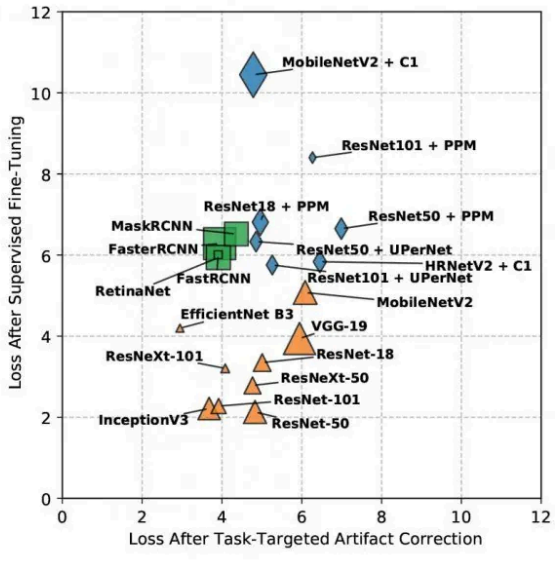
Testing and Conclusions

To return to the latest study from UoM and Facebook, the researchers sought to test JPEG comprehensibility and utility on images compressed between 10-90 (below which, the image is impossibly perturbed, and above which it's equal to lossless compression). Images used in the tests were pre-compressed at each value within the target quality range, entailing at least eight training sessions.

Models were trained on stochastic gradient descent across four methods: *baseline*, where no additional mitigations were added; *supervised fine-tuning*, where the training set has the advantage of pre-trained weights and labeled data (though the researchers concede that this is hard to replicate in consumer-level applications); *artifact correction*, where augmentation/amelioration is performed on the compressed images prior to training; and *task-targeted artifact correction*, where the artifact correct network is fine-tuned on returned errors.

Training occurred on a wide variety of apt datasets, including multiple variants of ResNet, FastRCNN, MobileNetV2, MaskRCNN and Keras' InceptionV3.

Sample loss results after task-targeted artifact correction are visualized below (lower = better).



It's not possible to dive deep into the details of the results obtained in the study, because the researchers' findings are split between the aim of evaluating JPEG artifacts and new methods to alleviate this; the training was iterated *per-quality* over so many datasets; and the tasks included multiple aims such as object detection, segmentation, and classification. Essentially, the new report positions itself as a comprehensive reference work addressing multiple issues.

Nonetheless, the paper broadly concludes that 'JPEG compression has a steep penalty across the board for heavy to moderate compression settings'. It also asserts that its novel unlabeled mitigation strategies achieve superior results among other similar approaches; that, for complex tasks, the researchers' supervised method also outperforms its peers, in spite of having no access to ground truth labels; and that these novel methodologies allow for model reuse, since the weights obtained are transferable between tasks.

In terms of classification tasks, the paper states explicitly that 'JPEG degrades the gradient quality as well as induces localization errors'.

The authors hope to extend future studies to cover other compression methods such as the largely disregarded JPEG 2000, as well as WebP, HEIF and BPG. They further suggest that their methodology could be applied to analogous research into video compression algorithms.

Since the task-targeted artifact correction method has proved so successful in the study, the authors also signal their intent to release the weights trained during the project, anticipating that ‘[many] applications will benefit from using our TTAC weights with no modification.’

n.b. Source image for the article comes from thispersondoesnotexist.com

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

Discover More