Group B: Image Compression and Restoration

Shayla Luong
Aditi Agrawal
Arushi Singh

Background

Images and videos are compressed when sent over internet.

WhatsApp converts (& Compress) most images to jpeg format and it puts Lossy Compression capabilities to good use in reducing image size and to make it light.

Here are some sizes of the images sent over internet via various social media platforms

- **Facebook Profile Image Size** – 180 x 180 px
- **Facebook Shared Image Size** – 1200 x 630 px
- **Instagram Reel Image Size** – 1080 x 1920 px
- **Instagram Feed Image Size** – 1080 x 1350 px
- **Twitter Profile image** – 400 x 400 px
- **Twitter Share image** – 1200 x 675 px
- **WhatsApp Square Post to Send Image Size** – 800 x 800 px

Dataset

- MNIST (60,000 Train,10,000 Test)
- CIFAR10 (50,000 Train, 10,000 Test)
- Flicker2K (800 Train and, 100 Test)
- Personal (newly added)

Classical Approaches

Image Interpolation

- Nearest-neighbor, bilinear, bicubic

Translational motion in frequency domain

Regularization

Statistics-based

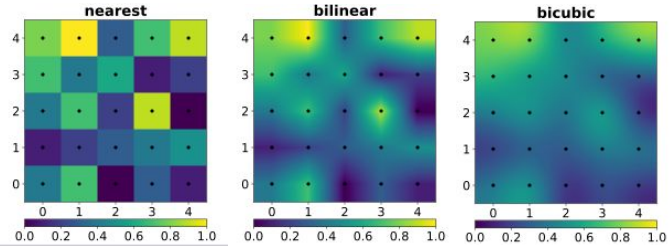


Image Source: https://en.wikipedia.org/wiki/Bicubic_interpolation

State-of-the-Art

- Convolutional Neural Network
- Residual Networks
- Generative Adversarial Network
- Transformers

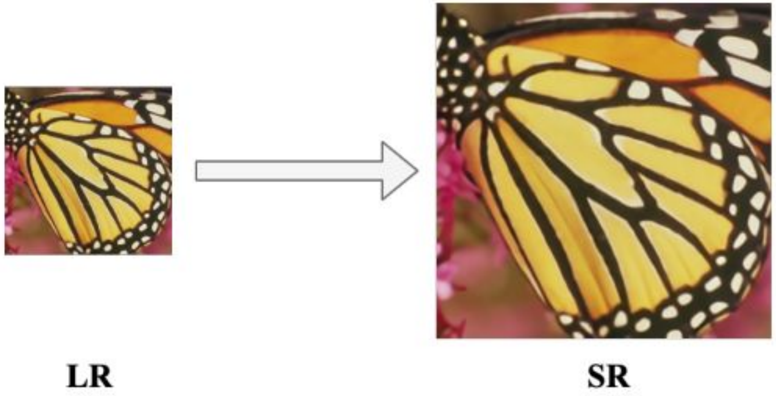
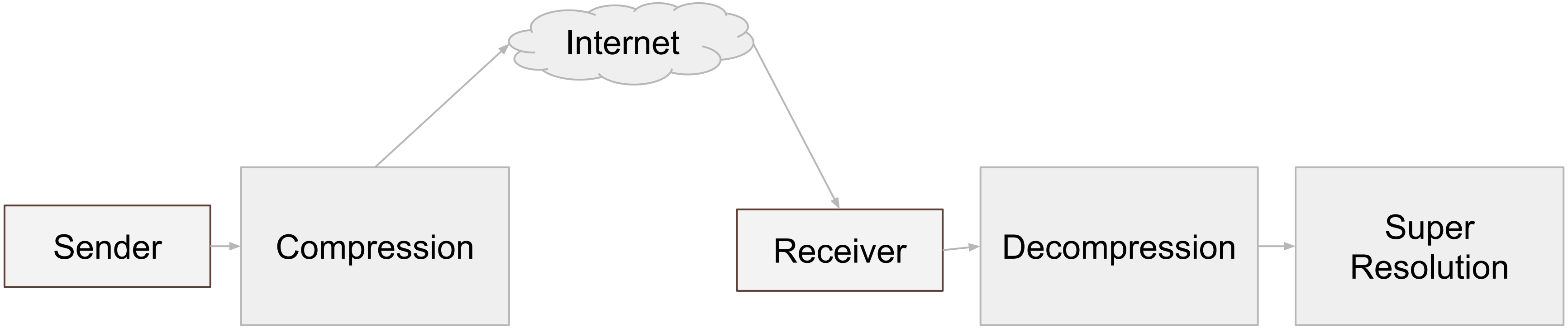


Image Source: https://arxiv.org/pdf/2109.14335.pdf

High Level Architecture



Deep Convolutional AutoEncoder based Lossy Compression

As we have already seen the , traditional image compression algorithms such as JPEG,JPEG 2000 used fixed transform matrix.

Drawback with this approach is:

Not optimal and flexible image coding solution for all image formats, therefore we would be using deep learning approach which contains autoencoder with convolution layers and PCA.

1) To replace the transform and inverse transform in traditional codecs, we design a symmetric CAE structure with multiple downsampling and upsampling units to generate feature maps with low dimensions. We optimize this CAE using an approximated rate-distortion loss function.

2) It's a principal components analysis (PCA)-based rotation to generate more zeros in the feature maps. Then, the quantization and entropy coder are utilized to compress the data further.

Block Diagram

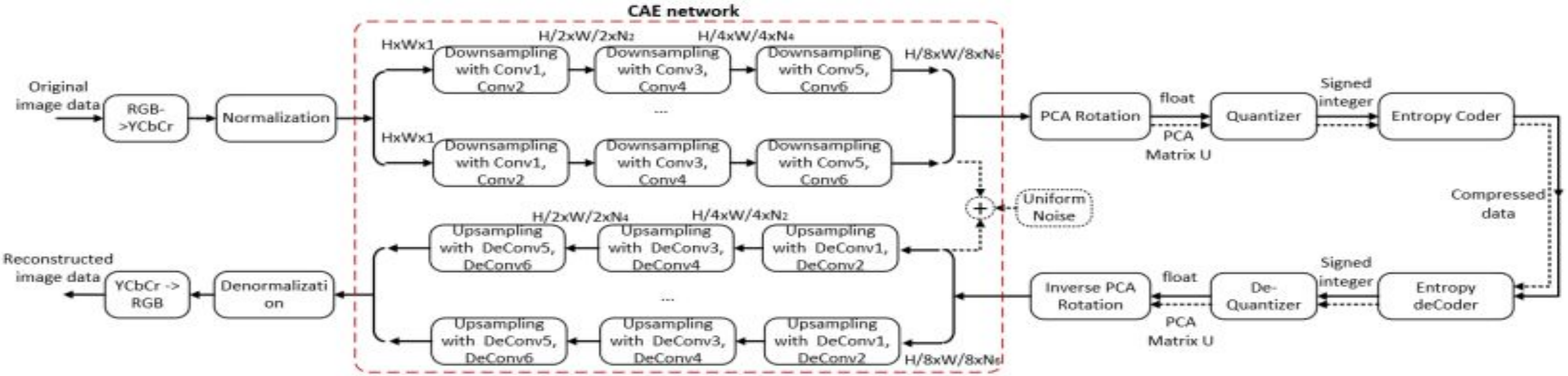


Fig. 1: Block diagram of the proposed CAE based image compression. (The detailed block for downsampling/upsampling is shown in Fig. 2)

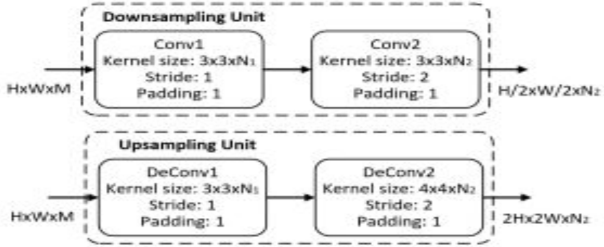


Fig. 2: Downsampling/Upsampling Units with two (De)Convolution Filters.

CAE Compression Model Results

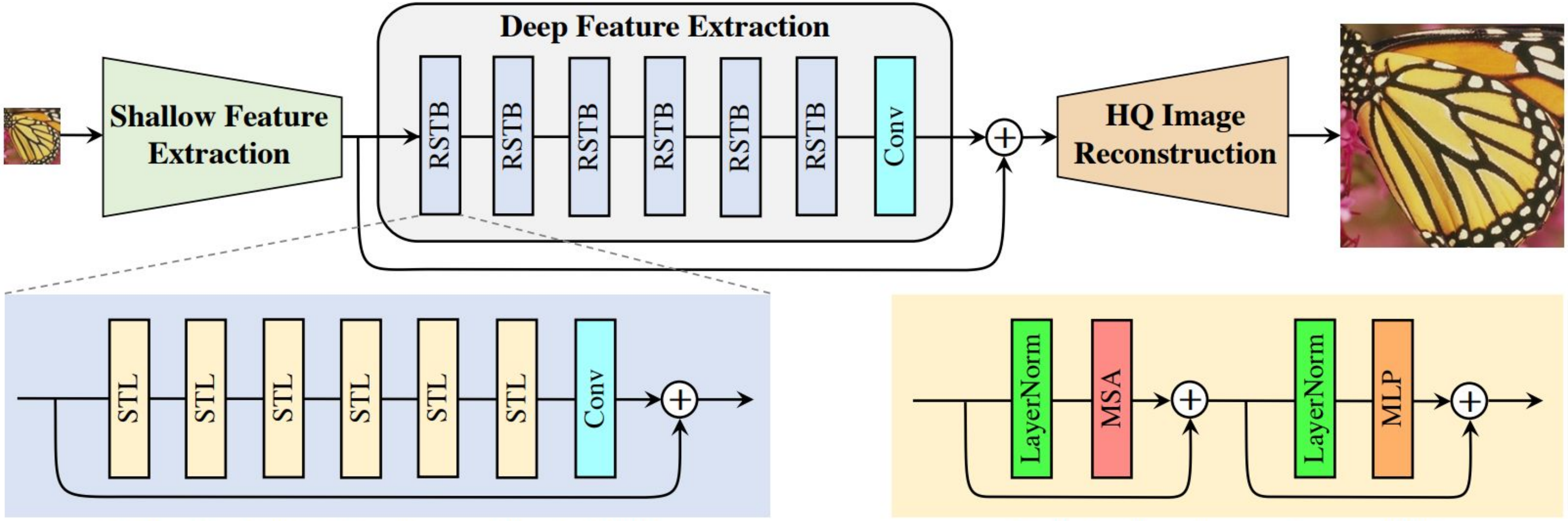
Images	JPEG_PSNR	CAE_PSNR
img1	25.133571730052722	68.80298894094759
img2	25.133571730052722	72.0766678288864
img3	23.492463587258207	71.06848423751096
img4	23.782298740309713	69.24729347565386
img5	24.498634719616142	70.9987425569537

Images	JPEG_SSIM	CAE_SSIM
img1	0.7324647203460863	0.9985165564310678
img2	0.8404630172197501	0.9985165564310678
img3	0.9898393374143969	0.9898393374143969
img4	0.6912463978940243	0.9980587167189803
img5	0.7678400774761748	0.9957950897580261

Transformer

- Self-attention mechanism
    - Learns relationships between elements of a sequence
    - Captures long-range dependency
    - Aggregate global information from complete input sequence
- Pre-trained on large-scale datasets
    - Finetune to specific tasks

SwinIR



(a) Residual Swin Transformer Block (RSTB)

(b) Swin Transformer Layer (STL)

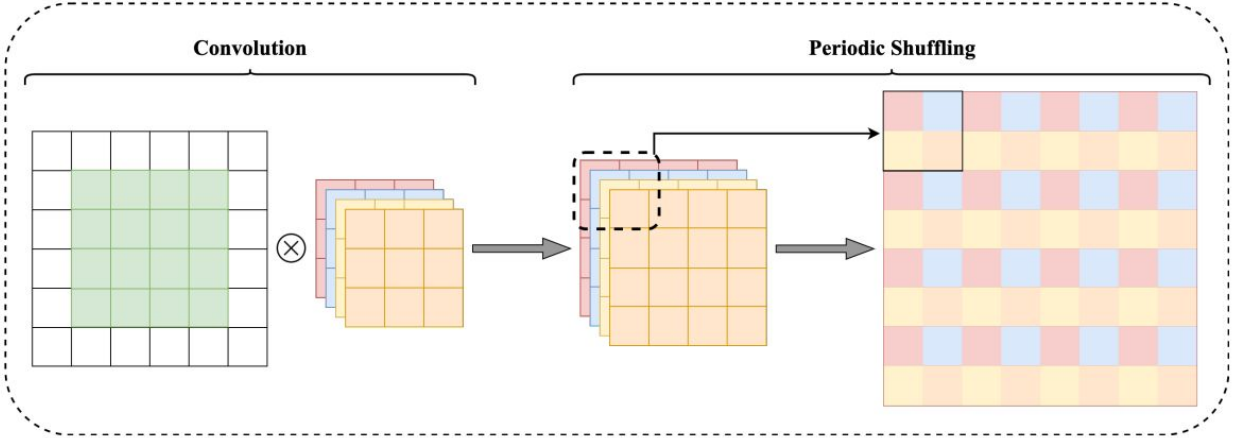
SwinIR

- Shallow features contain low-frequencies
- Deep features focus on restoring high-frequencies
- Local attention and shifted window mechanism
- Skip connection directly from shallow feature extraction to reconstruction module
- Multihead self-attention (MSA) and Multi-layer perceptron (MLP) in each Swin Transformer layer

Reconstruction

Upsampling method: sub-pixel convolution layer

Loss function: L_1 pixel loss, GAN loss, perceptual loss



Training SwinIR Results

Dataset - 60 high-resolution images
- 50 Training
- 10 Validation

Training
- 5 epochs
- Adam optimizer

Epochs	AVG PSNR
1	24.309989679055892
2	26.74641404859804
3	26.69878608011634
4	27.317388404477526
5	26.18571235406673


Proposed Method

1. Train a Convolutional Autoencoder (CAE)
2. Obtain new training dataset
3. Transfer-learning on pre-trained SwinIR model on new dataset
4. Compress images through CAE + restore the output through new SwinIR model
5. Evaluate results

Visual Results on Restoration


Original Image


JPEG Compression


CAE Compression + SWIN IR Restoration


Original Image


JPEG Compression


CAE Compression + SWIN IR Restoration

Future Work

Train SwinIR model using larger dataset

Generate dataset from online sharing

Transfer Learning on different SwinIR models

References

[1] V. Punj, “Whatsapp update on ios let’s beta users send HD photos and how it works.” https://www.livemint.com/technology/apps/whatsapp-update-on-ios-lets-beta-users-send-hd-photos-how-it-works-11627203919427.html, 2021.

[2] R. Y. Tsai and T. S. Huang. “Multipleframe image restoration and registration.” In Advances in Computer Vision and Image Processing, pages 317–339. Greenwich, CT: JAI Press Inc., 1984.

[3] Sean Borman and Robert L. Stevenson. “Super-resolution from image sequences - A review.” In Proceedings of the 1998 Midwest Symposium on Circuits and Systems, pages 374–378, 1998.

[4] Dong, C., Loy, C.C., He, K., et al. “Learning a deep convolutional network for image super-resolution.” *European Conf. on Computer Vision*, Zurich, Switzerland, 2014, pp. 184–199

[5] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *CVPR*, 2016.

[6] A. Vaswani et al. “Attention is all you need.” *arXiv preprint arXiv:1706.03762*, 2017.

[7] B. Garber et at., “Image Super-Resolution Via a Convolutional Neural Network”, http://cs229.stanford.edu/proj2020spr/report/Garber_Grossman_Johnson-Yu.pdf

[8] Wei, Zihao, et al. "A-ESRGAN: Training Real-World Blind Super-Resolution with Attention U-Net Discriminators." in *arXiv preprint arXiv:2112.10046*, 2021. Available: https://arxiv.org/abs/2112.10046

[9] J. Liang, J. Cao, G. Sun, K. Zhang, L. V. Gool, and R. Timofte. “SwinIR: Image Restoration Using Swin Tranformer,” in *arXiv:2108.10257v1 (eess.IV)* 23 Aug 2021. [Online]. Available: https://arxiv.org/abs/2108.10257

Thank you!!