WHAT IS SUPERRESOLUTION
SUPERRESOLUTION

Build a high resolution version of a given low resolution image

upscaling

\[ \text{n\times W} \]

\[ \text{n\times H} \]

\[ \text{n\times W} \]

\[ \text{H} \]

\[ \text{W} \]
ZOOM! ENHANCE!

Sure!

Can you enhance that? Zoom on the license plate
EVEN THE INTERNET KNOWS...

ONE DOES NOT SIMPLY

ENHANCE THE IMAGE
EXISTING TECHNIQUES

Interpolation (bilinear, bicubic, lanczos, etc.)

Interpolation + Sharpening (and other filtration)

Such methods are data-independent

Very rough estimation of the data behavior
EXISTING TECHNIQUES (DEEP)

Perceptual Losses for Real-Time Style Transfer and Super-Res:
Johnson et al. 2016

EnhanceNet: Mehdi et al. 2017

Super-Res with Deep Adaptive Image Resampling:
Jia et al. 2017

A Fully Progressive Approach to Single-Image Super-Res:
Wang 2018

Image Super-Res via Deep Recursive ResNets:
2018
OUR SOLUTION
TRAINING PIPELINE

Downscaling → LR image → SR model $W$ → Reconstructed HR image
TRAINING PIPELINE

Model Input

Downscaling = Filtering + Decimation

cutoff frequency at (or below) nyquist
Solve the optimization problem:

\[ W = \text{argmin} \sum_i \text{Dist}(x_i, F_W(D(x_i))) \]

\( \{x_i\} \)- training set
**MODEL (GWMT)**

4x upscaling model

![Diagram of the GWMT model](image.png)

Developed by Dmitry Korobchenko
MODEL (GWMT)
4x upscaling model

Low-pass: Bilinear up-scaling of the input image.
MODEL (GWMT)
4x upscaling model

Developed by Dmitry Korobchenko
Training on fixed-size random crops

Input data issues

JPEG compression artifacts

DATASET
OpenImagesV4*

Raw
JPEG (over-compressed)

* https://storage.googleapis.com/openimages/web/index.html
LOSS FUNCTION

MSE  HFEN  VGG  TV  GAN
**LOSS FUNCTION**

**MSE**

**HFEN**

**VGG**

**TV**

**GAN**

\[ x \xrightarrow{\text{down}} \hat{x} \xrightarrow{\text{SR}} y = F(x) \]

**MSE loss:**

\[ L = \frac{1}{N} \| x - F(x) \|^2 \]

**PSNR**

Peak Signal-to-Noise Ratio

\[ 10 \times \log_{10} \left( \frac{\text{MAX}^2}{\text{MSE}} \right) \]
HFEN* loss: $L = \alpha_1 \|HP(x - F(x))\|^2$

- HFEN*: High Frequency Error Norm

**LOSS FUNCTION**

\[
\text{VGG}^* \text{ loss: } L = \alpha_2 \| G(x) - G(F(x)) \|^2
\]

- VGG19 features taken after the 4\textsuperscript{th} convolutional layer (before 5\textsuperscript{th} max-pooling)

https://arxiv.org/abs/1409.1556
TV loss: \( L = \alpha_3 \int_{\Omega} |\nabla F(x)| \)

- Serves as a regularizer and has little influence on the optimization
GAN loss $= -\alpha_4 \ln D(F(x))$

ONE DOES NOT SIMPLY

ENHANCE THE IMAGE
ONE DOES NOT SIMPLY
ENHANCE THE IMAGE
COMPARISON
Original vs downscaled
COMPARISON
downscaled vs bicubic
COMPARISON

downscaled vs perceptual
COMPARISON
downscaled vs perceptual+GAN
COMPARISON
original vs bicubic
COMPARISON
original vs perceptual
COMPARISON
original vs perceptual+GAN
COMPARISON

easy details (hat)

Original

Perceptual

Downscaled (input)

Bicubic

Perceptual + GAN
COMPARISON
details (eye)
COMPARISON

hard details (feathers plume)

Original

Perceptual

Downscaled (input)

Bicubic

Perceptual + GAN
WHAT ABOUT SYNTHETIC IMAGES?
COMPARISON

Synthetic Images
COMPARISON

Synthetic Images
COMPARISON
Synthetic Images
COMPARISON

Synthetic Images
OBSERVATIONS
On synthetic image upscaling

• Synthetic images have more high frequency details
• Synthetic images with dithering contains noise-like artifacts
• The Network has never seen synthetic images during trainings
• Presence of artifacts in training image is reflected into upscaling artifact
  • Especially with GANs
• We can probably improve these results
UPSCALING SYNTHETIC IMAGES
GOAL
Train Super Resolution for synthetic images
SOLUTION?
Train on game images!
SOLUTION?
Train on game images!

- Difficult to produce
SOLUTION?
Train on game images!

• Difficult to produce
• Extremely biased dataset
SOLUTION?
Train on game images!

- Difficult to produce
- Extremely biased dataset
- License issues?
NEW GOAL
Train SuperRes with natural images and apply to synthetic images
AUGMENTATION
SOLUTION

Augment photographic images

To reduce the compression artifacts, we will extract random crops and downscale them to our training crop size
SOLUTION
Downscale with aliasing

Filter the image with a cutoff above Nyquist limit

\[
x \times 1/4
\]
SOLUTION
Downscale with variable aliasing

Use different cutoff limits above Nyquist

Every downscaling now generate examples with different aliasing features.
SOLUTION
Stochastic decimation

After filtering, instead of sampling on a regular grid, jitter each sampling point.

Every downscaling now generate very different examples.
SOLUTION

Variable stochastic decimation

Full control over introduced noise/aliasing effect
COMPARISON WITH PREVIOUS METHOD
EVALUATION
Note. Not only unaliased but also denoised!

EVALUATION
COMPARISON WITH INPUT IMAGES
Note. The input images are interpolated by Nearest Neighbor algorithm to make it same size with upscaled image.
INPUT VS OUTPUT
INPUT VS OUTPUT
INPUT VS OUTPUT (REAL IMAGE)
INPUT VS OUTPUT (REAL IMAGE)
INPUT VS OUTPUT (REAL IMAGE)
QUESTIONS?
THANK YOU!
DISCUSSION

- GAN is hard to train: how to deal with artifacts?
- How to find the optimal weights for loss function?