Session 8 - Image Super-Resolution and Neural Style Transfer

Due Oct 10 by 11:59pm **Points** 1,500 **Submitting** a text entry box or a website url
Available Sep 26 at 10am - Oct 10 at 11:59pm 15 days

This assignment was locked Oct 10 at 11:59pm.

Session 8 - Image Super-Resolution and Neural Style Transfer

        - Pixel Loss
        - Charbonnier Loss
        - Content Loss
        - Texture Loss:
            - Gram Matrix
        - Adversarial Loss
    - **SRGAN**
    - **Neural Style Transfer**
    - **Assignment (https://github.com/wangzheallen/awesome-human-pose-estimation)**

Introduction

Image Super-Resolution (SR) refers to the process of recovering high-resolution images from low-resolution images.

It enjoys a wide range of real-world applications, such as:

- medical imaging
- surveillance
- security, etc.

In general, this problem is very challenging and inherently ill-posed since there are always multiple HR images corresponding to a single LR image.

Hierarchically-structured taxonomy

[SOURCE (https://arxiv.org/pdf/1902.06068.pdf)](https://arxiv.org/pdf/1902.06068.pdf)

PROBLEM DEFINITION

Image super-resolution aims at recovering the corresponding HR images from the LR images. We can formulate this as:

$$I_y \;=\; Sup\left(I_x;\;\theta\right)$$

but to confuse you, we can also formulate this as:

$$I_x \;=\; D\left(I_y;\;\delta\right)$$

and claim that the LR (I_x) image can be modeled as the output of a **Degradation** function D that works on HR (I_y) image and δ are its parameters.

Available Datasets

Besides these datasets, some datasets widely used for other vision tasks are:

- ImageNet
- MS-COCO
- VOC2012
- CelebA
- Flickr2K
- Waterloo Exploration Dataset

Image Quality Assessment

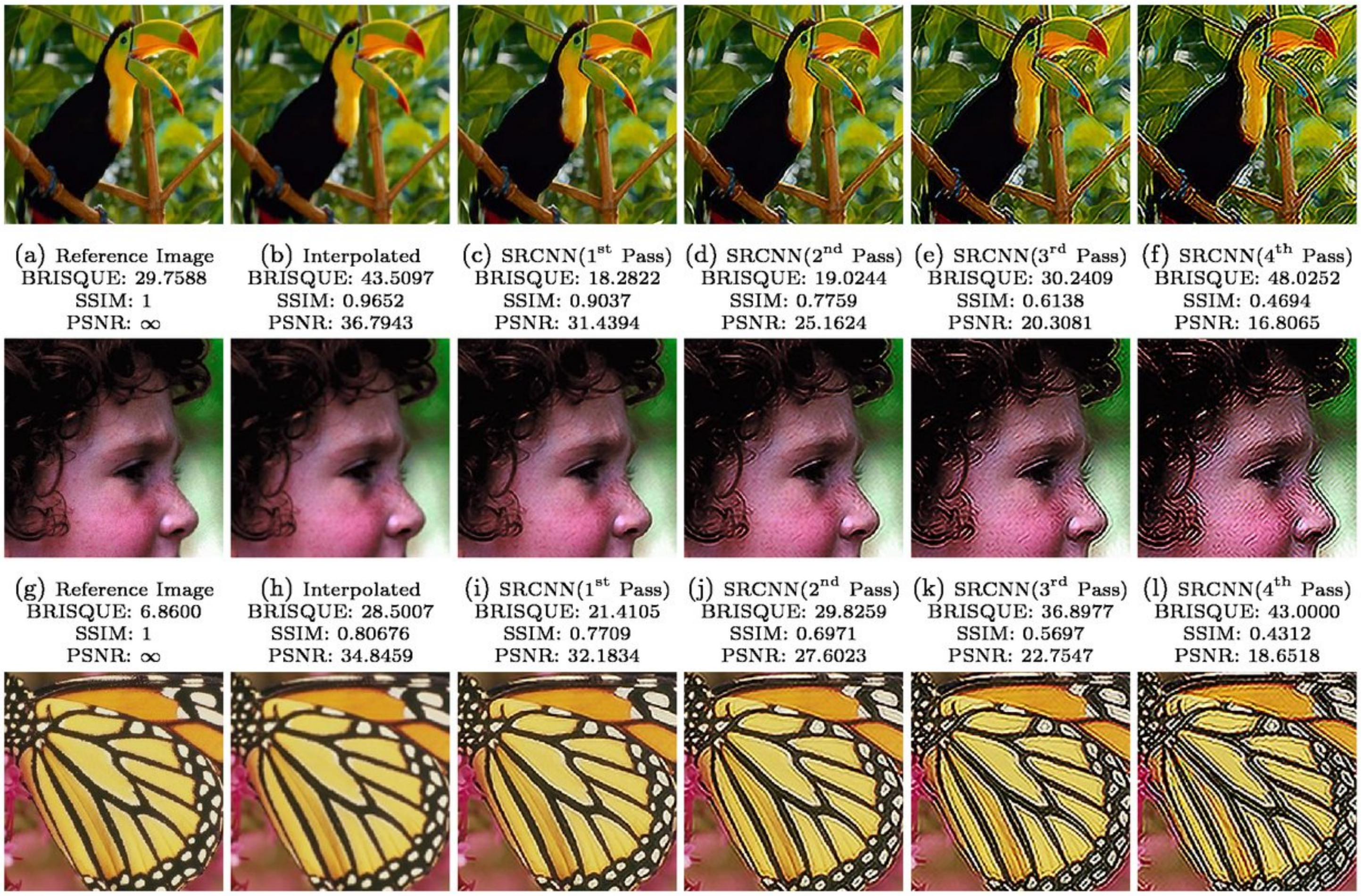
The image quality assessment includes:

- subjective methods based on humans' perception
- objective computational methods:
    - PSNR
    - SSIM

Objective computational methods are most used (time-saving), but often are unable to capture the human visual perception, and that may lead to large differences in IQA results.

Peak Signal-to-Noise Ratio - PSNR

PSNR is one of the most popular **reconstruction** quality measurements of lossy transformation (e.g. image compressions, image inpainting).



(a) Reference Image BRISQUE: 29.7588 SSIM: 1 PSNR: ∞
(b) Interpolated BRISQUE: 43.5097 SSIM: 0.9652 PSNR: 36.7943
(c) SRCNN(1st Pass) BRISQUE: 18.2822 SSIM: 0.9037 PSNR: 31.4394
(d) SRCNN(2nd Pass) BRISQUE: 19.0244 SSIM: 0.7759 PSNR: 25.1624
(e) SRCNN(3rd Pass) BRISQUE: 30.2409 SSIM: 0.6138 PSNR: 20.3081
(f) SRCNN(4th Pass) BRISQUE: 48.0252 SSIM: 0.4694 PSNR: 16.8065

(g) Reference Image BRISQUE: 6.8600 SSIM: 1 PSNR: ∞
(h) Interpolated BRISQUE: 28.5007 SSIM: 0.80676 PSNR: 34.8459
(i) SRCNN(1st Pass) BRISQUE: 21.4105 SSIM: 0.7709 PSNR: 32.1834
(j) SRCNN(2nd Pass) BRISQUE: 29.8259 SSIM: 0.6971 PSNR: 27.6023
(k) SRCNN(3rd Pass) BRISQUE: 36.8977 SSIM: 0.5697 PSNR: 22.7547
(l) SRCNN(4th Pass) BRISQUE: 43.0000 SSIM: 0.4312 PSNR: 18.6518

(m) Reference Image BRISQUE: 13.2584 SSIM: 1 PSNR: ∞
(n) Interpolated BRISQUE: 34.1258 SSIM: 0.9022 PSNR: 27.4325
(o) SRCNN(1st Pass) BRISQUE: 28.2522 SSIM: 0.8093 PSNR: 24.7472
(p) SRCNN(2nd Pass) BRISQUE: 62.2542 SSIM: 0.6197 PSNR: 19.3736
(q) SRCNN(3rdPass) BRISQUE: 95.2858 SSIM: 0.4547 PSNR: 15.3125
(r) SRCNN(4th Pass) BRISQUE: 140.783 SSIM: 0.3435 PSNR: 12.6557

For Image SR, PSNR is defined via the maximum pixel value (denoted by L) and the mean squared error (MSE) between images.

Let us consider this 2x2 image.

Assuming only Blue values are changed (or there is only 1 channel)

MSE = $\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$ = 1/4($(242 - 231)^2$ + $(231 - 219)^2$ + $(219 - 150)^2$ + $(150 - 100)^2$) = 1881.5

The maximum pixel value for an 8bit image would be $2^8 - 1 = 255$.

If this was easy let's look at it a complicated way:

Since the PSNR is only related to the pixel-level MSE, only caring about the differences between corresponding pixels instead of visual perception, **it often leads to poor performance in representing the reconstruction quality in real scenes, where we're usually more concerned with human perceptions.**

Structural Similarity - SSIM

SSIM (proposed to be closer to human perception compared to PSNR) measures the structural similarity between images in terms of luminance, contrast, and structures. For an Image I with N pixels, the luminance and contrast are estimated as the mean and standard deviation of the intensity of the image, i.e.

Comparisons on luminance and contrast denoted as and respectively are given by:

where k are constants for avoiding instability and and L is the maximum pixel value.

Besides, the image structure is represented by the normalized pixel values (   ),

whose correlations (i.e. inner product) measure the structural similarity, equivalent to the correlation coefficient between I and I'. Thus the structural comparison function is defined as:

k3 = $(k_2L)^2/2$

Finally!

where are control parameters for adjusting the relative importance (all are kept 1 generally).

CODE (https://github.com/VainF/pytorch-msssim/blob/master/pytorch_msssim/ssim.py)

Upsampling Methods

Of course, how we perform upsampling in our models is of utmost importance.

Nearest-neighbor Interpolation

Bilinear Interpolation

Bicubic Interpolation

Convolutional Methods

Transpose Convolution

Sub-pixel layer (Pixel Shuffle Algorithm)

Network Designs

Loss Functions

Pixel Loss

measures the pixel-wise difference between two images and mainly include L1 or L2 loss.

The pixel loss constrains the generated HR images to be close to the ground truth on the pixel value. Comparing with L1 loss, the L2 loss penalizes larger errors but is more tolerant to small errors, and thus often results in too smooth results.

In practice, the L1 loss shows improved performance and convergence over L2 loss. Training with pixel loss optimizes PSNR, but doesn't directly optimize the perceptual quality, and hence generates images which might not be pleasing to the human eye.

Charbonnier Loss

This loss function is used in LapSRN instead of the generic L2 loss. The results show that Charbonnier loss deals better with outliers and produces sharper images compared to those generated with L2 loss, which are generally smoother.

Content Loss or Perceptual Loss

A milestone paper for generating SR images with better-perceived quality is **Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network (https://arxiv.org/abs/1609.04802)** (SRGAN).

The authors use a perceptual loss function composed of a *content loss* and an *adversarial loss*. The content loss compares deep features extracted from SR and HR images with a pre-trained **VGG network. (https://arxiv.org/abs/1409.1556)**

Extract the feature map of HR images and fake HR image from VGG19 and compute the MSE between these two features.

Texture Loss

On account that the reconstructed image should have the same style (color, textures, contrast, etc) with the target image, texture loss is introduced in EnhanceNet. This loss function tries to optimize the Gram matrix of feature outputs inspired by the Style Transfer loss function.

Adversarial Loss

Used in all GAN-related architectures, adversarial loss helps in fooling the discriminator and generally produces images which have better perceptual quality.

SRGAN

CODE (http://github.com/leftthomas/SRGAN)

NEURAL STYLE TRANSFER

We have actually already covered it by now today.

Let's refer to the PyTorch **Tutorial Page (http://pytorch.org/tutorials/advanced/neural_style_tutorial.html)** for this.

ASSIGNMENT

1. Train SRGAN on the drone images dataset you created. Your target is to train an SRGAN network (pre-trained models are fine) to be awesome at drone/flying object Super Resolution. Once done, upload it to your webpage - 1000pts.
2. Add a neural style transfer page on your Webpage - 500

VIDEO

EVA4P2S8