Structural Similarity (SSIM) Index Exploration - Po-Han Chen

Team Members: Po-Han Chen

Introduction

Structural similarity (SSIM) index [1] is a metric to evaluate how similar two images are. SSIM is widely adopted in image processing society to measure the performance of a technique. Unlike other error-based assessment methods such as mean square error (MSE) and peak signal to noise ratio (PSNR) that focus on pixel-wise errors, SSIM simultaneously looks at the luminance, the contrast, and the structure of two images. To understand more about this method, we explore how SSIM reacts to different distortion sources and how it performs on different image domains.

Background

Basics

SSIM index is calculated by comparing the luminance, contrast, and structure between two images. Given a reference image X and an input image Y (both in gray scale), we can first compute the following properties of these two images:

- Mean (μ_x , μ_y)
  - the average pixel value in the image, calculated as $\frac{1}{N}\sum_{i=1}^{N} x_i$
- Variance (σ_x , σ_y)
  - the variance of pixel value in the image, calculated as $\frac{1}{N-1}(\sum_{i=1}^{N}(x_i-\mu_x)^2)^{1/2}$
- Co-Variance (σ_{xy})
  - the co-variance of the pixel values between two images, calculated as $\frac{1}{N-1}\sum_{i=1}^{N}(x_i-\mu_x)(y_i-\mu_y)$

With these properties, we can compute the similarity scores in the following three categories:

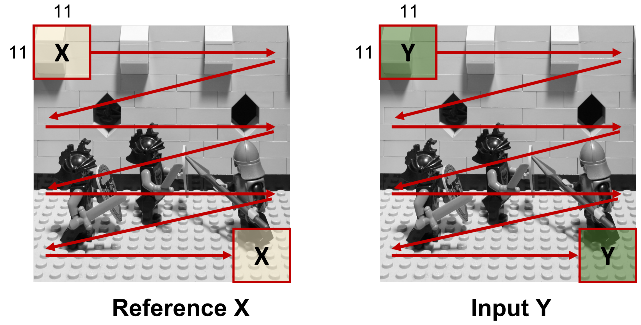
- Luminance Similarity $l(x,y)=\frac{2\mu_x\mu_y+C_1}{\mu_x^2+\mu_y^2+C_1}$
- Contrast Similarity $c(x,y)=\frac{2\sigma_x\sigma_y+C_2}{\sigma_x^2+\sigma_y^2+C_2}$
- Structure Similarity $s(x,y)=\frac{\sigma_{xy}+C_3}{\sigma_x\sigma_y+C_3}$

Finally, these similarity measures are blended into one final SSIM score.

$$SSIM(x,y)=l(x,y)^{\alpha}c(x,y)^{\beta}s(x,y)^{\gamma}$$

Apply to Images

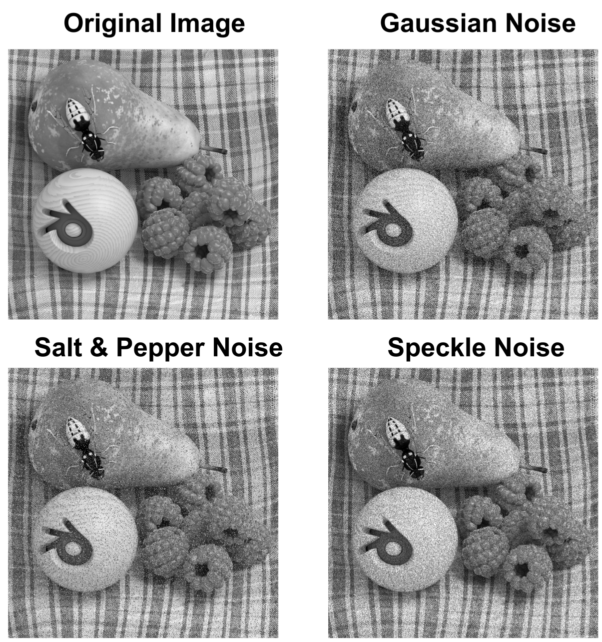
Because image features are typically not uniform across the whole image, it is better to use SSIM locally than globally. The authors suggested to apply SSIM index in a 11x11 window and slide it across the whole image. The final global SSIM index value is calculated as the average of all local SSIM index values.



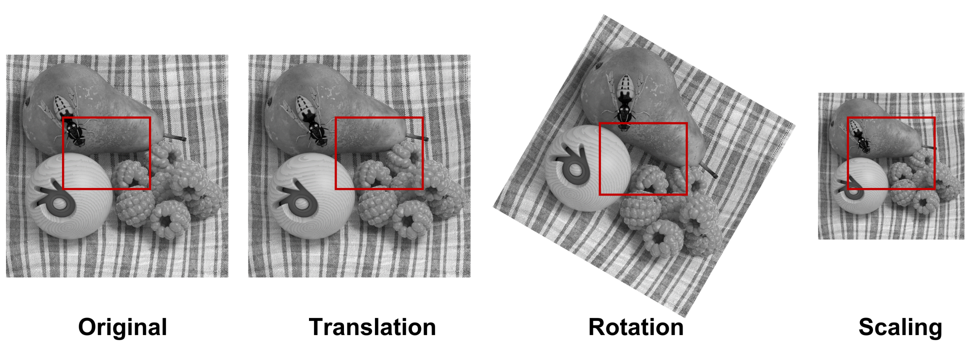
Method

SSIM versus Different Distortion Sources

Images can be distorted in multiple ways. We expect that SSIM will behave differently to different kinds of distortions. Specifically, we want to see how SSIM reacts to noises sources and transformations. For noises, we injected gaussian noise, salt & pepper noise, and speckle noise into the image. Then, we adjust the amount of noisiness and see how SSIM reacts.



For image transformation distortions, we tried translation, rotation, and scaling. Because SSIM works on two images with the same resolution, some cropping and border handling is needed to keep the transformed image in the same resolution. In order to make things easier, we only use a window of pixels sitting in the center of the image. For translation, we simply move the window in the direction that is oppose to the translation direction. For rotation, we rotate the image and use the same window (the window has to be small enough such that blank pixels will not be taken into account) to crop out the region of interest. For scaling, the window position has to be adjusted such that it is positioned in the center of the scaled image.



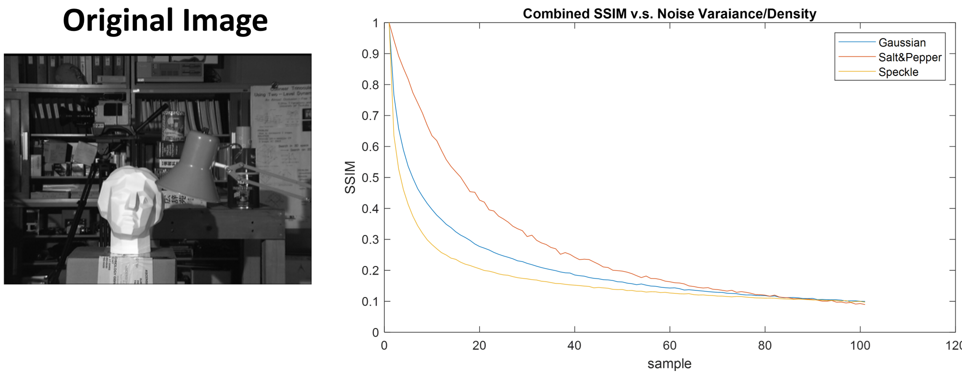
SSIM Evaluated in Different Image Domains

In the original SSIM paper, the authors apply SSIM to images that are encoded in 8bit non-linear domain. In this section, we will show how SSIM will react when we encode the image into linear domain. Typically, this is done be raising power of pixel values to 2.2 or 1/2.2 (known as gamma). For simplicity, we choose the gamma value to be 2.

Result

SSIM Experiments on Gaussian/Salt&Pepper/Speckle Noise

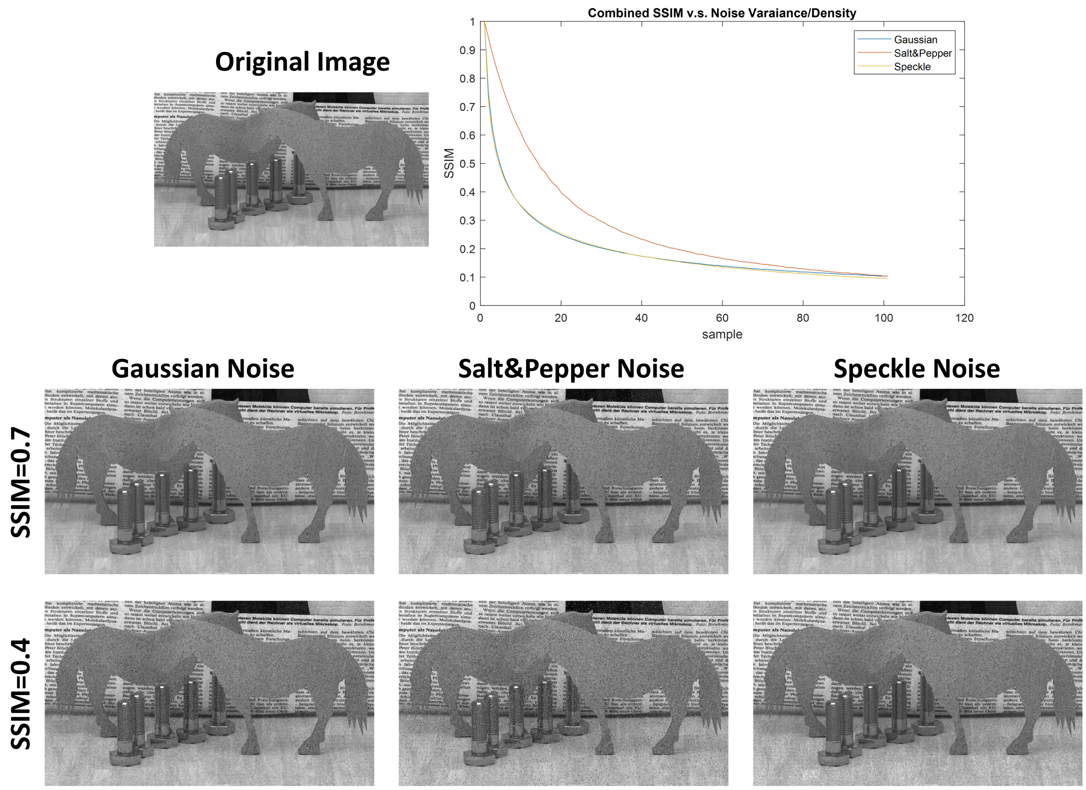
Each kind of noise has its own parameter to change how noisy it is. In this experiment, we gradually increase the noisiness until SSIM index drops from 1.0 to 0.1. Then, we record the parameter (usually variance) that yields 0.1 SSIM. From zero noise (clean image) to this parameter, we divided this range into N (here, we choose 100) samples and calculated its SSIM index. In this way, we can visualize how sensitive SSIM is with respect to different kinds of noise:



From the experiment results, we observed 3 things:

1. SSIM is very sensitive to Gaussian noise, but its impact to perceptual quality is minor when it compares to other types of noise
2. For Salt&Pepper noise, it is NOT as sensitive as Gaussian noise, but it creates bigger impact to perceptual quality
3. Speckle noise is the worst, and SSIM is good at detecting such noise

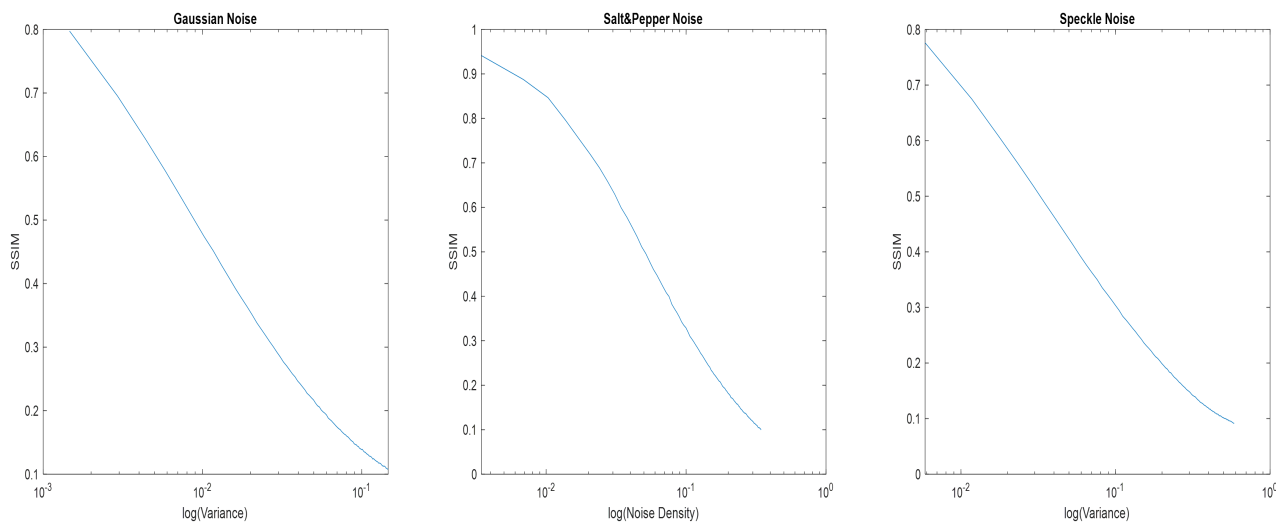
For most of our test cases, this observation stays true. However, there is one test case that has a different behavior:



In this test case, speckle noise does not seem to create huge impact to the perceptual quality. The curve of speckle noise is roughly the same as the curve of gaussian noise. We think it is because speckle noise is a data-dependent noise (i.e. multiplicative noise) so it doesn't really have a universal characteristic to SSIM index.

Log-scale Plot

As Brian pointed out during the presentation, the SSIM-versus-noise curve looks like an exponential function. Hence, we plot the figure with the x-axis displayed in logarithmic format:



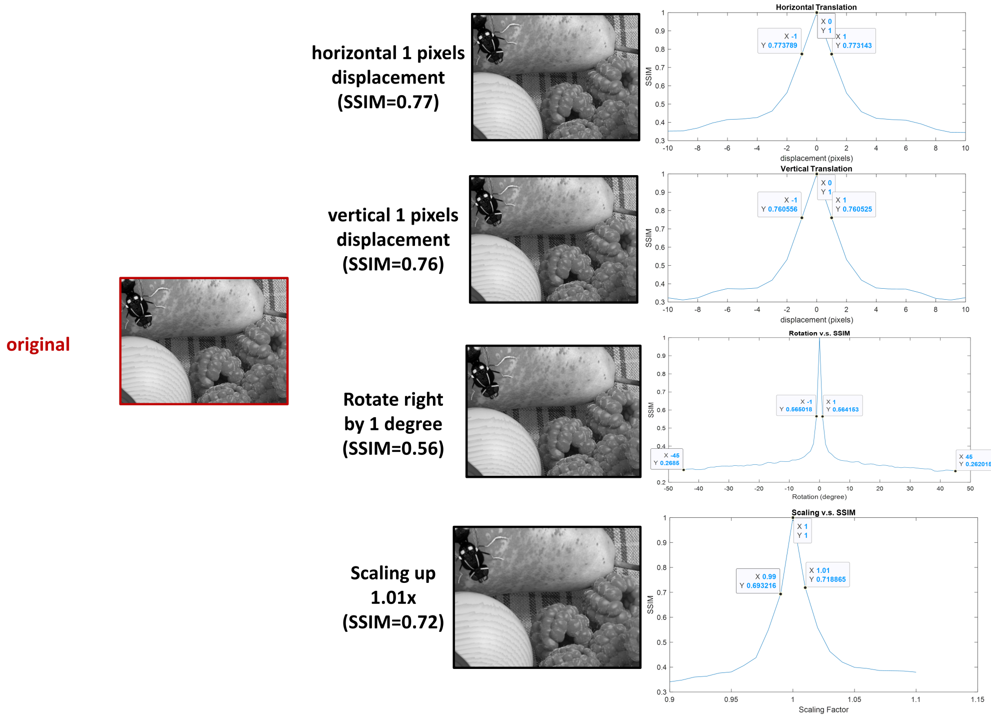
Indeed, the log curve looks straight in some range. If we look at the simplified SSIM formula:

$$\mathrm{SSIM}(\mathbf{x}, \mathbf{y}) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)\left(\sigma_x^2 + \sigma_y^2 + C_2\right)}.$$

We can see the amount of noisiness (x-axis), **which is the variance**, appears as a power term in the formula. Hence, from the figure we can see it looks close to a straight line. However, because the noise will also affect the mean and covariance, in some regions it will not appear as an obvious straight line due to other factors.

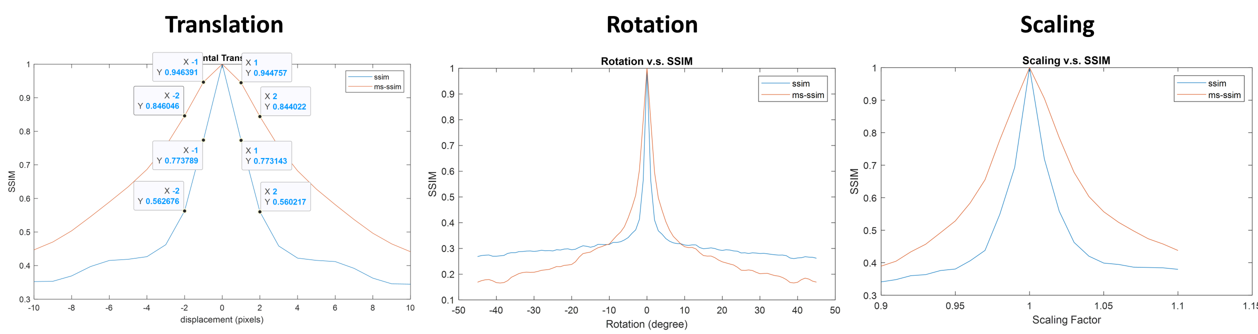
SSIM Experiments on Translation/Rotation/Scaling

In this section, we explore how SSIM reacts to different kinds of image transformation.



From the experiment results we can observe that each kind of image transformation method created a huge impact to the SSIM index even with very tiny amount of transformation. However, the perceptual quality of all of the distorted images is good. We can compare to the noise experiment in the previous section where a SSIM index value of 0.7 refers to a very distorted image. In this experiment, because the amount of transformation is small, we expect the luminance difference and contrast difference is small. However, the structure difference is calculated based on co-variance which is computed pixel-by-pixel. A small amount of transformation will create a huge impact to the structural measurement in this case.

Such pixel-wise co-variance measurement difference can be alleviated by multi-scale SSIM, which applies SSIM index in multiple resolution layers and blend them into a single SSIM index:



Conclusion

In this project, we explored how SSIM reacts to different kinds of noise types, we found that SSIM is sensitive to gaussian noise, but it has small impact to the perceptual quality. For salt&pepper noise, it is the opposite where SSIM is not very sensitive to the noise, but it hurts perceptual quality badly. We also explored how image transformation affect SSIM. We found that even a small amount of transformation will largely affect the SSIM index but not the quality. We think it is due to the pixelwise covariance difference in the structure comparison measurement. This effect can be alleviated by using multi-scale SSIM where such small displacement/rotation/scaling can be ignored in subsampled layers.

Future Work

Due to limited amount of time, we didn't explore how SSIM behaves when images are encoded in different image domains. For example, linear and non-linear domain. Future works can try to design experiments that compares the perceptual quality of a set of images that has the same SSIM index but are encoded in different image domains. A better encoding method should give a SSIM index that is better aligned with human perceptual quality.

Reference

[1] Zhou Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," in *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, April 2004

[2] pohantw/psych221-final (github.com)