Digital Image Denoising

CS229 Final Project
Thomas Liu (tcliu@stanford.edu)

Dataset

We generate a custom dataset for this project that consists of 20 scenes captured in color with a camera. Increasing the ISO sensitivity of the camera increases the amount of noise in the scene, and the lowest noise image is chosen as ground truth. Each image is divided into 150 subimages of 512x512 pixels and pixel intensities are normalized to values between 0 and 1.

Features

Features are the normalized RGB pixel intensities of each image. Since adjacent pixels are the most relevant for denoising, we select a patch of size K around the pixel being denoised as the features. This results in feature vectors of length $3K^2$.

Metrics

Peak Signal-to-Noise Ratio (PSNR) is an objective metric that looks at absolute error between the noisy and noiseless image. It is derived from squared loss.

$$MSE = \frac{1}{HW}\sum_{y=1}^{H}\sum_{x=1}^{W}\sum_{c\in\{R,G,B\}}\left[I_c(x,y) - \hat{I}_c(x,y)\right]^2$$

$$PSNR = -10\log_{10}(MSE)$$

Structural Similarity (SSIM) [1] is a perceptual metric that looks at differences in luminance, contrast and structure. Values range from -1 (least similar) to 1 (most similar).

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

Overview

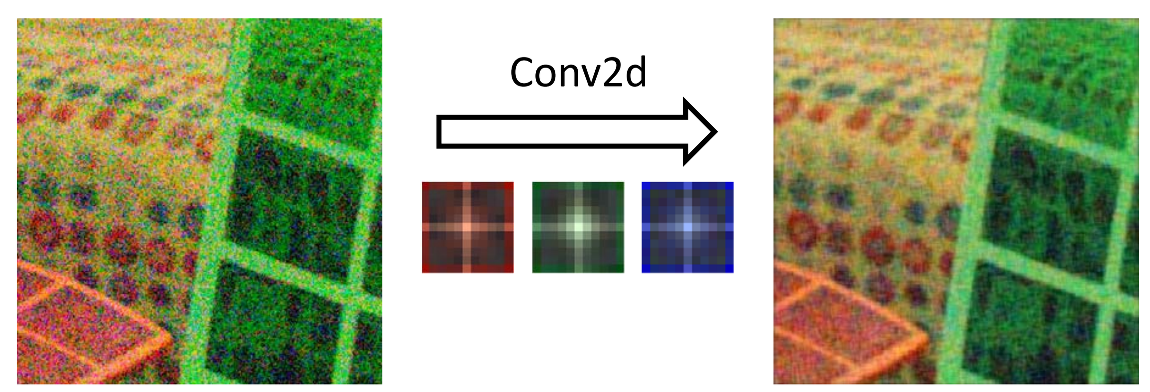
Denoising is an essential operation in digital image processing with applications in computer vision and photography. Given an image corrupted by noise, we want to improve image quality by removing as much noise as possible. We use supervised learning to develop models that are trained on noisy and noise-free versions of the same image. We briefly explore unsupervised learning methods to see if they are suitable for image denoising.

Models

Ordinary Least Squares

Filtering approach for denoising. Each output pixel is a linear combination of adjacent input pixels in a K by K patch. Use linear regression to learn optimal weights for predicting output pixels from the input image.

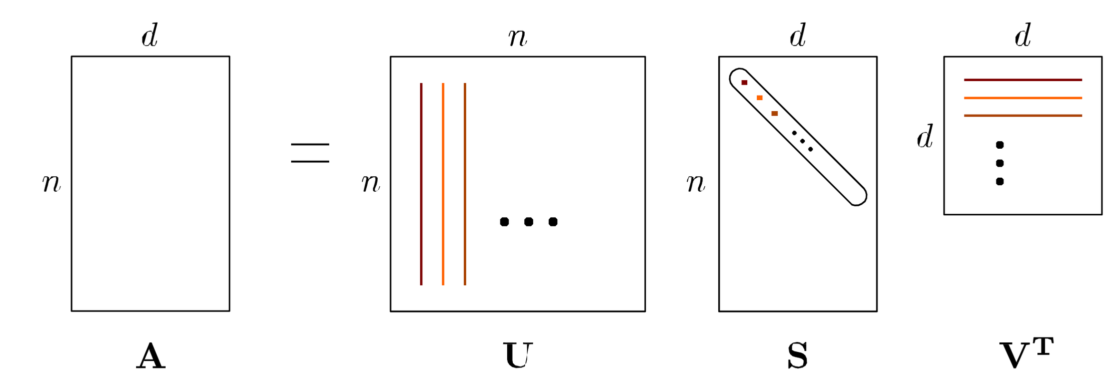
$$\hat{I}_c(x,y) = \sum_{i,j=-K/2}^{K/2}\sum_{\hat{c}} w_{i,j,\hat{c}} I_{\hat{c}}(x+i, y+j)$$



PCA: Best Rank k Approximation

Dimensionality reduction approach for denoising. Treat input image as matrix and use SVD to produce a rank k approximation. Optimal k for any given image can be learned.

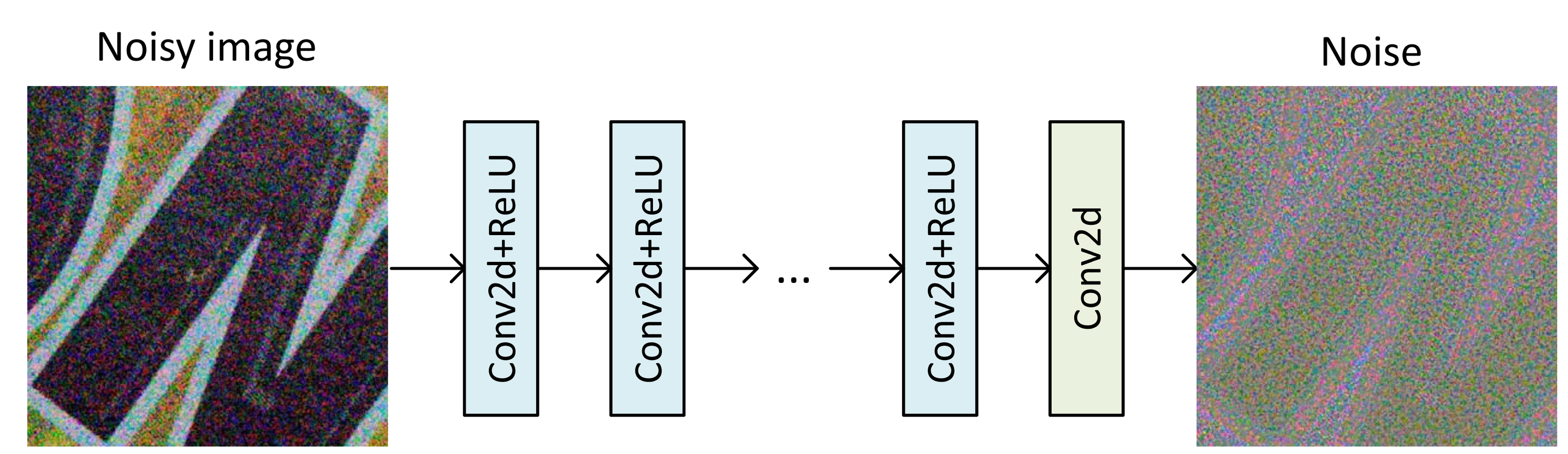
$$\mathbf{I} = \mathbf{U\Sigma V^T} \qquad \mathbf{I}_k = \sum_{i=1}^{k}\sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$$



Denoising CNN (DnCNN)

Rather than simply filtering out the noise in the image, we build a convolutional neural network (CNN) that predicts the noise instead of the denoised image. The network consists of repeated convolutional layers with ReLU nonlinearity and a linear convolutional output layer. Hyperparameters include network depth D, number of hidden units H and convolution kernel size K. This architecture is inspired by Zhang et al. [2] with modifications being choice of K ≥ 3 and removal of batch normalization layers.

D	H	K	Avg. PSNR
6	64	3	26.67
8	64	3	27.27
12	32	3	27.18
14	32	3	27.11
16	32	3	26.47
14	32	5	27.13
14	32	7	27.57



Results

Model	Avg. PSNR	Avg. SSIM
Identity	19.15	0.5679
Median	24.66	0.8749
OLS	23.57	0.8579
PCA Best k*	24.81	0.8536
DnCNN	27.57	0.9124

*Optimistic estimate, based on optimal choice of k

Discussion

Ordinary Least Squares: Reasonably good at denoising images, performs similarly to median filter but with lower computational complexity. Drawback is blurring of edges which is inherent to lowpass filtering but not an issue with median.

PCA Best Rank *k*: Better than expected at denoising images, considering it's meant to be a method for lossy compression. Works poorly when image has a lot of fine detail and denoised images often suffer from compression artifacts.

DnCNN: As expected, has the highest performance of the three models. Wide networks are better for low noise images and deep networks are better for the noisiest images. Appears to be memorizing the data slightly, halftone print pattern shows up in unrelated images.

Future

- Apply denoise operation before debayering
- Retrain models to target specific ISO sensitivities
- Include more sensor technologies and formats in the dataset

[1] Z. Wang, A.C. Bovik, H.R. Sheikh and E.P. Simoncelli. "Image quality assessment: from error visibility to structural similarity". *IEEE Trans. Image Process*. vol. 13, no. 4, pp. 600–612, 2004.
[2] K. Zhang, W. Zuo, Y. Chen, D. Meng and L. Zhang. "Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising". arXiv:1608.03981, 2016.