PnP ADMM Assignment Report

Deconvolution using ADMM

In the deconvolution method, we try to extract an image from the blurred plus white gaussian noiseadded image. This is done using ADMM for Optimization. ADMM converts non-constraint terms into constraint terms. Then we convert the optimization equation to augmented lagrangian. For easy calculation.

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{u}) = f(\boldsymbol{x}) + \lambda g(\boldsymbol{v}) + \boldsymbol{u}^T(\boldsymbol{x} - \boldsymbol{v}) + \frac{\rho}{2}\|\boldsymbol{x} - \boldsymbol{v}\|^2.$$

$$\boldsymbol{x}^{(k+1)} = \underset{\boldsymbol{x}\in\mathbb{R}^n}{\operatorname{argmin}} \; f(\boldsymbol{x}) + \frac{\rho}{2}\|\boldsymbol{x} - \tilde{\boldsymbol{x}}^{(k)}\|^2, \tag{5}$$

$$\boldsymbol{v}^{(k+1)} = \underset{\boldsymbol{v}\in\mathbb{R}^n}{\operatorname{argmin}} \; \lambda g(\boldsymbol{v}) + \frac{\rho}{2}\|\boldsymbol{v} - \tilde{\boldsymbol{v}}^{(k)}\|^2, \tag{6}$$

$$\bar{\boldsymbol{u}}^{(k+1)} = \bar{\boldsymbol{u}}^{(k)} + (\boldsymbol{x}^{(k+1)} - \boldsymbol{v}^{(k+1)}), \tag{7}$$

The function corresponding to V^{k+1} basically is regularizer and in our problem it can be easily replaced by denoiser BM3D.The function corresponding to X^{k+1} is data fidelity term where f(x)=y-Ax.

For special case of Deconvolution

$x_{k+1} = F^{-1} \{\frac{F(h)^*. F(y) + \rho F(\tilde{x})}{|F(h).F(h)| + \rho}\}$

v_{k+1}= output of BM3D

Here F() is the fourier transform ,h is Blur kernel and y is the blurred image .

Parameters:

First kernel

ρ	1
γ	1
σ	0.01
λ	0.0001

η is not taken into account.

Results

(Image Blurred with kernel 1)

(Result)

Second Kernel

ρ	0.01
γ	1
σ	7
λ	49

(Image Blurred with kernel 2)

(Result)

After running the optimization code for 100 iterations, the blur image converts to deblur with PSNR 33.75 db and 27.29 dB

Note- Used to scipy to calculate FFT transformation

Compressed Sensing

The concept of compressive sensing is using undersampled data (lower than nyquist rate) to reconstruct the original image. The idea behind this is in Fourier or frequency domain the image can be represented by a combination of transform matrix sparse matrix. So with sparse data also the original image can be reconstructed.

In this algorithm to calculate a mask, we make a mask using a matrix of the size of 4096x128**2 with Gaussian distribution of variance 1/4096 and 0 mean. The original image is flattened and multiplied with the mask to get the output image(Y) then the optimization technique ADMM is used to reconstruct the original image given the Y. The similar thing is done for a Gaussian distribution of variance 1/8192 and 0 mean U and V are calculated the same as deconvolution. The following equation is used to compute x̂

$$\hat{\boldsymbol{x}} = (\boldsymbol{G}^T\boldsymbol{G} + \rho\boldsymbol{I})^{-1}(\boldsymbol{G}^T\boldsymbol{y} + \rho\tilde{\boldsymbol{x}}).$$

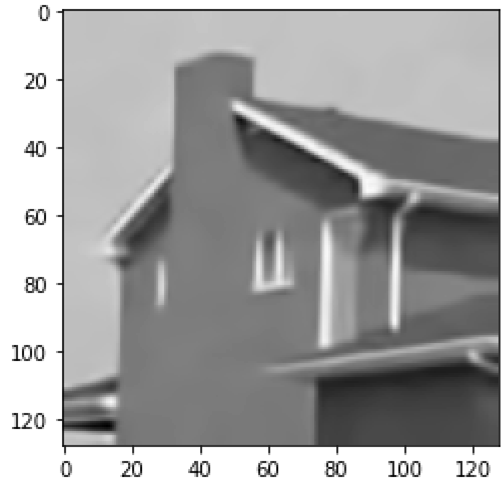
Results

For variance = 1/4096

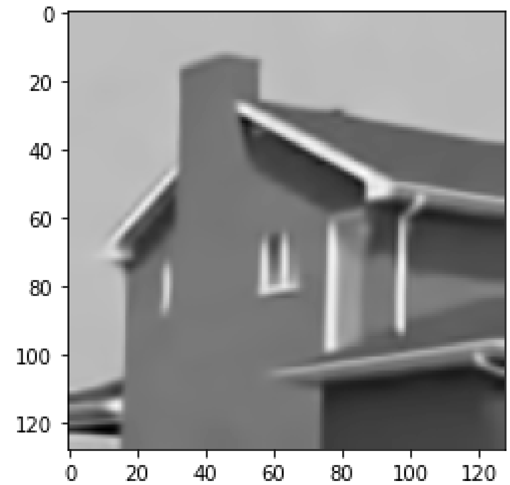
ρ	1
γ	1
σ	30
λ	900

For variance = 1/8192

ρ	1
γ	1
σ	30
λ	900



31.414 dB



32.204 dB

References

Chan, S.H., Wang, X. and Elgendy, O.A. (2016). Plug-and-play ADMM for image restoration: Fixed-point convergence and applications. IEEE Transactions on Computational Imaging, 3(1), pp.84-98.

James Gregson's Website