Denoising Image Analysis

Undergraduate Project Report

MTH393A

by

Sandeep Parmar

Roll no - 210922

DEPARTMENT OF MATHEMATICS AND STATISTICS

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

April 2024

T. Sandhan

Acknowledgement

I would like to express my deepest gratitude to Professor **Dr. Tushar Sandhan**, for their unwavering guidance, encouragement, and invaluable mentorship throughout the completion of this project report. I would also like to thank my Department Supervisor **Dr. BV Rathish Kumar** for always supporting and motivating me to conduct research.

Mentor Details:

Dr. Tushar Sandhan
Dept. of Electrical Engineering
IIT Kanpur

Dr. BV Rathish Kumar
Dept. of Mathematics and Statistics
IIT Kanpur

Declaration

I hereby declare that the work presented in the project report titled **"Denoising Image Analysis"** contains my own ideas in my own words. At places, where ideas and words are borrowed from other sources, proper references, as applicable, have been cited. To the best of our knowledge this work does not emanate from or resemble other work created by person(s) other than mentioned herein.

Sandeep Parmar
Roll no - 210922
BS-SDS

Abstract

IMAGE denoising is a classical image processing problem,but it still remains very active nowadays with the massive and easy production of digital images. A digital image is like a big grid, or matrix, made up of tiny squares called pixels. Each pixel represents a color or shade. In a black and white (grey level) image, each pixel shows a shade of grey. In color images, each pixel has three colors mixed together: red, green, and blue (RGB). These colors blend in different amounts to create all the colors we see in the pictures. Image noise is like visual static, appearing as random specks or grain over a picture. There are various reason like poor camera quality in which image get noise as a result we are not able to some useful information from image. For example, Microscopic images are often affected by noise due to environmental factors, such as temperature fluctuations, and physical errors in the imaging process.This noise can obscure relevant features and patterns, making it challenging to interpret the images accurately and we know how important to know those image accurately.

0.0.1 Denoising Algorithms

There are various algorithsm available for image denoising, but each has pros and cons One category of denoising methods concerns transform based methodsThe main idea is to calculate wavelet coefficients of images, shrink the coefficients and finally reconstruct images by inverse transform.Another category is related to patch-based method, which explores the non-local self-similarity of natural images. In this project, we mainly look into algorithms which are based on optimization, also called Low Rank Matrix Recovery Problem.

0.0.2 Low Rank Matrix Recovery

In mathematics,**low-rank approximation** is a *minimization* problem, in which the cost function measures the fit between a given matrix (the data) and an approximating matrix (the optimization variable), subject to a constraint that the approximating matrix has reduced rank.

There are various use-case of LRM,few of them are - Low rank matrix approximation, which aims to recover the underlying low rank matrix from its degraded observation, has a wide range of applications in computer vision and machine learning.The low rank nature of matrix formed by human facial images allows us to reconstruct the corrupted faces . The Netflix customer data matrix is believed to be low rank due to the fact that the customers' choices are mostly affected by a few common factors. The video clip captured by a static camera has a clear low rank property, based on which background modeling and foreground extraction can be conducted .

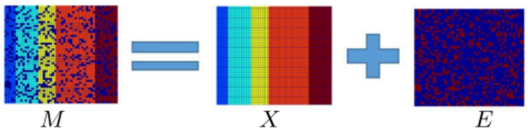

Fig. 1. Illustration of low-rank matrix recovery.

FIGURE 1

Main motivation of this project is to explore image denoising algorithms based on optimization. In recent years, nuclear norm based image denoising get attracted due to its high accuracy. First, we implement all those algorithms followed by test, how they perform on Microscopic Images. Microscopic images are very sensitive to its features, every minute pixel box may contain some important information.

Contents

Chapter 1

Weighted Nuclear Norm Minimization - WNNM

1.1 Defintions

1. ***Singular Value***
A singular value of real matrix X is positive square root of eigen-value of symmetric matrix X^TX or XX^T.

2. ***Nuclear Norm***
The nuclear norm of a matrix X is defined as the sum of its singular values, i.e.

$$||X||_* = \sum_i |\sigma_i(X)|$$

where $\sigma_i(X)$ is the ith singular value of matrix X.

1.2 Background

Nuclear Norm Minimisation (NNM)
It is the one of the method for solving problems of low rank matrix approximation. NNM aims to approximate matrix Y by X while minimizing the *nuclear norm* of

X.
One distinct advantage of NNM lies in that it is the tightest *convex* relaxation to the non-convex LRMF problem with certain data fidelity term, and hence it has been attracting great research interest in recent years. One of the low rank minimisation problem formulated as Cai et al [reference insert]

$$\hat{X} = argmin_X ||Y - X||_F^2 + \lambda ||X||_*$$

where λ is a positive constant, and corresponding solution can be obtained by

$$\hat{X} = US_\lambda(\Sigma)V^T$$

where $Y = U\Sigma V^T$ is the SVD of Y and $S_\lambda(\Sigma)$ is the soft-thresholding function on diagonal matrix Σ with parameter .

Problem with NNM
In order to pursue the convex property, the standard nuclear norm treats each singular value equally, and as a result, the soft thresholding operator in (3) shrinks each singular value with the same amount λ.

To improve the flexibiltiy of nuclear norm, weighted nucleaur norm is introduced. For a matrix X WNNM

$$||X||_{w,*} = \sum_i |w_i \sigma_i(X)|_1$$

. where w is the weight vector s.t. $w_i \geq 0$.

1.3 WNNM

Authors[1] proposed WNNM algorithm for image denoising. The goal of image denoising is to estimate the latent clean image from its noisy observation.

Formulation of problem
Given a observed data matrix Y (original image that contain noise), we have to estimate latent data matrix X using Y. Based on above setting proposed WNNM

problem is

$$\min_X \|Y - X\|_F^2 + \|X\|_{w,*} \tag{1.1}$$

where $\|\cdot\|_F$ is the Frobenius norm (F-norm) to measure the difference between Y and X.

Optimization

Authors solve the optimization problem - (1) by considering different cases of weights. Every assumption on weights leads to different solution.

Case 1 : Weights are in non-ascending order

If weights satisfy $w_1 \geq \ldots \geq w_n \geq 0$ the WNNM problem in (1) has a globally optimal solution

$$\hat{X} = US_w(\Sigma)V^T$$

where $Y = U\Sigma V^T$ is the svd of Y and $S_w(\Sigma)$ is the generalised soft threshholding operator with weight vector w,

$$S_w(\Sigma)_{ii} = max(\Sigma_{ii} - w_i, 0)$$

Case 2 : Weights are in arbitrary order

If the weights are in arbitrary order then the WNNM probelm in (1) is non-convex and thus we cannot have a global minimum of it. For that authors use iterative algorithm to solve it.

Method

Y is given data matrix and $Y = U\Sigma V^T$ the SVD of it. Let $B = P\Lambda Q^T$. Authors solve the following optimization problem iteratively

$$(\hat{P}, \hat{\Lambda}, \hat{Q}) = arg\ min_{P,\Lambda,Q}||P\Lambda Q^T - \Sigma||F^2 + ||P\Lambda Q^T||w,*,$$

s.t. $P^T P = I$, $Q^T Q = I$, where I is identity matrix.

Above optimization problem can be solved in two part. In one case keeping P and Q constant and optimise Λ and other case fixed Λ and optimize P and Q. By breaking it into two step we get closed form solution.

For k-th iteration we do following updates.

$$(P_{k+1}^T, \Phi, Q_{k+1}^T) = SVD(\Lambda_K)\Lambda_{k+1} = P_{k+1}^T S_w(\Sigma) P Q_{k+1}$$

The final estimation $\hat{X}$ can be obtained by

$$\hat{X} = U\hat{P}^T S_w(\Sigma)\hat{Q}V^T$$

Case - 3 : If the weights satisfy $0 \leq w_1 \leq .... \leq w_n$ then $\hat{X} = US_w(\Sigma)V^T$

1.4 WNNM for Image Denosing

Image denoising aims to reconstruct the original image X from its noisy observation $Y = X + N$, where N is assumed to be additive Gaussian white noise with zero mean and variance σ_n^2.

For implementation, we are not directly optimizing equation (1.1), but we break the images into patches, get an estimate of the patches, and finally aggregate all denoised patches. For a local patch Y_j in image Y, we can search for its non-local similar patches across the image by methods such as *block matching*. In the algorithm, we find all possible patches of size $d \times d$ of the matrix (Overlapping of patches is allowed).

we have $Y_j = X_j + N_j$. By using the noise variance σ_n^2 to normalize the F-norm data fidelity term $||Y_j - X_j||_F^2$, we have the following energy function:

$$\hat{X}_j = argmin_X \frac{1}{\sigma_n^2}||Y_j - X_j||_F^2 + ||X_j||_{w,*}$$

Determination of weight vector w

$$w_i = \frac{c\sqrt{n}}{\sigma_i(X_j) + \epsilon}$$

$\sigma_i(X_j)$ is the i-th singular value of X_j and n is the number of similar patch , $\epsilon = 10^{-16}$ to avoid dividing by zero and c is constant.

Algorithm 1 Image Denoising by WNNM

Require: Noisy image y

Initialize $\hat{x}^{(0)} = y$, $y^{(0)} = y$
for $k = 1$ to K **do**
    Iterative regularization $y^{(k)} = \hat{x}^{(k-1)} + \delta(y - \hat{y}^{(k-1)})$
    for each patch y_i in $y^{(k)}$ **do**
        Find similar patch group Y_j
        Estimate weight vector w
        Singular value decomposition $[U, \Sigma, V] = \mathrm{SVD}(Y_j)$
        Get the estimation: $\hat{X}_j = US_w(\Sigma)V^T$
    end for
    Aggregate X_i to form the clean image $\hat{x}^{(k)}$
end for

Ensure: Clean image $\hat{x}^{(k)}$

1.5 Experiment Setting

1.5.1 Image Data

I have tested the WNNM algorithm on 4 different gray-scale images. First image is Lena of size 512×512, second is Amplitude image (A Microscopic Image) of size 512×512, third is House image of size 256×256 and last one is Camera-man test image of size 70×70.

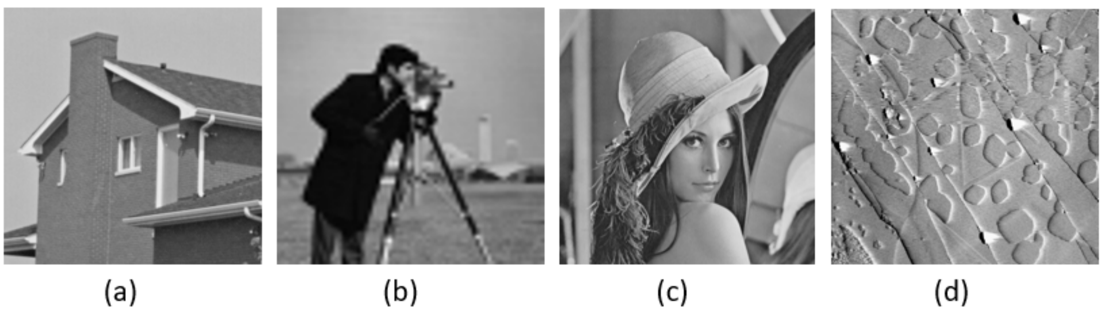

Figure 1.1: Test Images: (a) House (b) Cameramen (c) Lena (d) Amplitude

1.5.2 Parameter

There are several parameters (δ, c, K and patch size) in the proposed algorithm. For all noise levels, the iterative regularization parameter δ and the parameter c are fixed to 0.1 and 2.8, respectively. Iteration number K and patch size are set based on noise level. For higher noise level, we need to choose bigger patches and run more times the iteration. Authors Suggested patch size to6×6, 7×7, 8×8 and 9×9 for $\sigma_n^2 \leq 20$, $20 < \sigma_n^2 \leq 40$, $40\sigma_n^2 \leq 60$ and $60\sigma_n^2$, respectively. K is set to 8, 12, 14, and 14 respectively, on these noise levels.

1.5.3 Accuracy Measure

PSNR (peak signal-to-noise ratio) is the ratio between the image's maximum possible power and the power of corrupting noise that affects the quality of its representation. To estimate the PSNR of an image, compare it to an ideal clean image (or ground truth) with the maximum possible power. A higher PSNR value indicates better image quality.

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right)$$

where

$$MSE = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\left(I(i,j) - K(i,j)\right)^2$$

, I and K image matrix. And Max_I is the maximum possible pixel value of the image Since we used 0 to 1 pixel range so $Max_I = 1$

1.6 Results

I first tested Lena Image, add the Gaussian Noise with zero mean and variance σ_n^2 to the original clean image for 5 different $\sigma_n = \{10, 20, 30, 50, 100\}$.

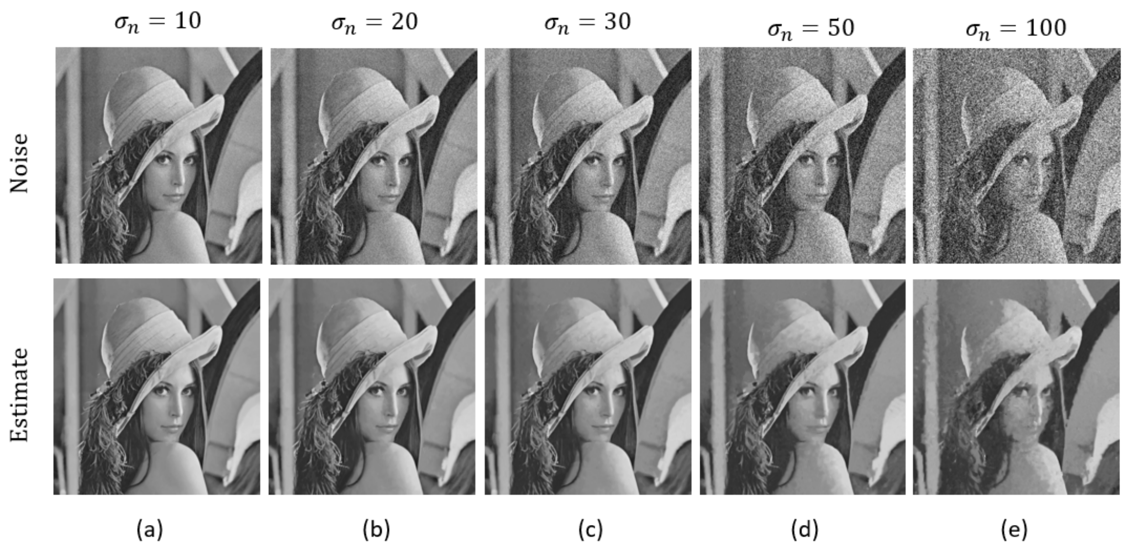

FIGURE 1.2: PSNR changes from (a) 28.13 db to 36.66 db (b) 22.10 db to 33.07 db (c) 18.58 db to 31.26 db (d) 14.14 db to 28.73 db (e) 8.12 db to 25.24 db

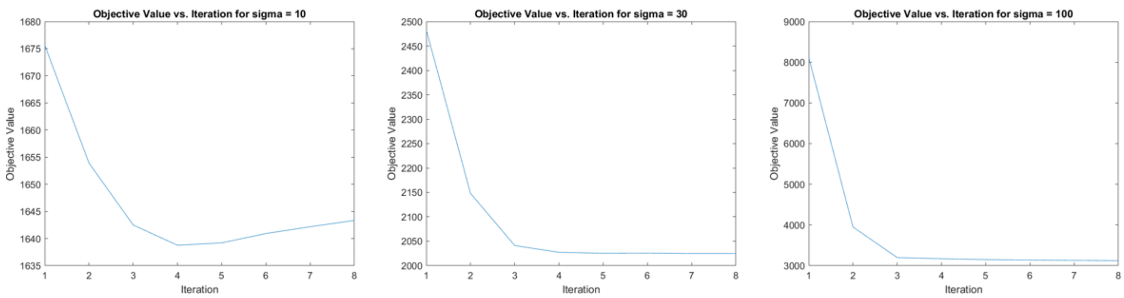

FIGURE 1.3: objective function value after each iteration for Amplitude image

Our main objective is to minimise main function to get, I observed in experiment that it works well but it get blurred on high variance noise. In first few iterations it decrease very fast then it become on saturation level. In a some cases I observe that after getting minimum value (saturation level) function value start increasing, like in first image of fig-1.2, but PSNR value still increasing after each iteration.

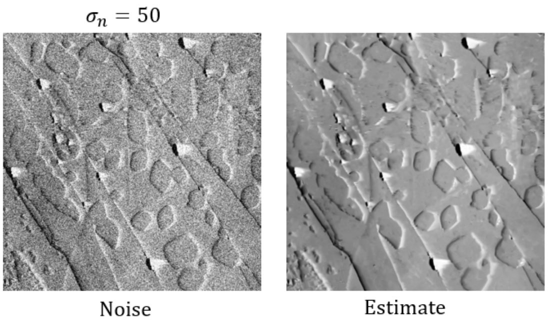

FIGURE 1.4: Amplitude: PSNR change from 14.12 db to 24.45 db

On implementation of WNNM on amplitude image I observed that, PSNR of output image is less than that of Lena image perhaps both images are of same size, this implies quality of WNNM depends on structure of image. In Amplitude we got more blur image which is not good.

Other images results is given in Chapter-5 in table format.

Chapter 2

Weighted Schatten p-Norm Minimization for Image Denoising

2.1 Introduction

This method is based on Nuclear Norm Minimisation (NNM), a method of Low rank matrix approximation. It is extended version of WNNM.

Defintion :- Schatten p-norm Given a real matrix X and $\sigma_i s$'s singular values of X, then Schatten p-norm defined as l_p-norm of singular values $= (\sum_i \sigma_i^p)^{\frac{1}{p}}$ with $0 \leq p \leq 1$.

Need of the Algorithm
Despite the convexity relaxation to optimization problem of the NNM model, it has been indicated that the recovery performance will degrade in the presence of measurement noise, and the solution can seriously deviate from the original solution of rank minimization problem. More specifically, as observed in experiments, the NNM based model will shrink too much the low rank components of the data.

Therefore, it has been proposed in various research to enforce low rank regularization by using the Schatten p-norm. Theoretically, Schatten p-norm will guarantee a more accurate recovery of the signal while requiring only a weaker restricted isometric property than traditional trace norm.

2.2 Problem Formulation

Inspired by the Schatten p-norm minimization and WNNM, "authors" propose a new low rank regularizer namely Weighted Schatten p-Norm Minimization (WSNM) for Low Rank Matrix Analysis (LRMA).

The proposed weighted Schatten p-norm of matrix $X \in R^{m\times n}$ is defined as

$$||X||_{w,S_p} = (\sum_{i=1}^{min(n,m)} w_i\sigma_i^P)^{\frac{1}{p}} \implies ||X||_{w,S_p}^p = \sum_{i=1}^{min(n,m)} w_i\sigma_i^P = tr(W\Delta^p)$$

where $w = [w_1, \ldots, w_{min(n,m)}]$ is non-negative vector , σ_i is the i-th singular value of X and both W and Δ are diagonal matrices whose diagonal entries composed of w_i and σ_i respectively.

Given a matrix Y, authors proposed LRMA model aims to find a matrix X, which is as close to Y as possible under the F-norm data fidelity and the weighted Schatten p-norm regularization::

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} \|\mathbf{X} - \mathbf{Y}\|_F^2 + \lambda\|\mathbf{X}\|_{\mathbf{w},S_p}^p.$$

where λ is a trade-off parameter to balance the data fidelity and regularization.

2.3 WSNM for Image Denoising

Authors used patch based approach to implement as they had done in WNNM. For a patch $Y_j = X_j + N_j$, (All assumption of WNNM applicable here). Then our WSNM optimisation problem can be defined as,

$$\hat{\mathbf{X}}_i = \arg\min_{\mathbf{X}_i} \frac{1}{\sigma_n^2} \|\mathbf{Y}_i - \mathbf{X}_i\|_F^2 + \|\mathbf{X}_i\|_{w,S_p}^p ,$$

where σ_n^2 denotes the noise variance, the first term of represents the F-norm data fidelity term, and the second term plays the role of low rank regularization.

Algorithm 2 Image Denoising by WSNM

Require: Noisy image y

Ensure: Denoised image $\hat{x}^{(K)}$

```
Initialization:
for k = 1 to K do
    Iterative regularization y^(k) = x̂^(k−1) + α(y − x^(k−1))
    for each patch y_j^k in y^(k) do
        Find similar patches to form matrix Y_j
        Singular value decomposition [U, Σ, V] = SVD(Y_j)
        Estimate weight vector w.
        Calculate Δ by using Algorithm 3.
        Get the estimation: x̂_j = UΔV^T
    end for
    Aggregate X_i to form the denoised image x^(k)
end for
return The final denoised image X̂^(K)
```

Algorithm 3 WSNM via GST

Require: Matrix Y, weight set $\{w_i\}_{i=1}^n$ in non-descending order, p

Ensure: Δ

```
Decompose Y into UΣV^T, where Σ = diag(σ_1, ..., σ_τ)
for i = 1 to r /* can calculate in parallel */ do
    δ_i = GST(σ_i, w_i, p)
end for
return Δ = diag(δ_1, ..., δ_r)
```

GST :- It is Generalised soft thresholding function which threshold the value of σ_i using w_i and p. Detail explanation is given in paper.

Weight Initialisation

$$w_j = c\sqrt{n}/(\delta_j^{1/p}(\hat{X}_j) + \epsilon)$$

where n is the number of similar patch in Y_i , $\epsilon = 10^{-16}$ and $c = 2\sqrt{2}\sigma_n^2$. Since $\delta_j(\hat{(X)}_i$ is unavailable before $\hat{x}$ estimated it can initialised by

$$\delta_j(\hat{X}_i) = \sqrt{max\{\sigma_j^2(Y_i) - n\sigma_n^2, 0\}}$$

.

2.4 Experiment setting

I used same test images as WNNM. All parameters setting is also same as WNNM. In this method, p is new parameter, authors suggested to choose p is $\{1.0, 0.85, 0.75, 0.7, 0.1, 0.05\}$, they provide analysis in Appendix of paper[2].

2.5 Results

I tested WSNM for 5-different $\sigma = \{10, 20, 30, 50, 100\}$. I choosed $p = 0.95$. Also at $p = 1$ WSNM algorithm become WNNM, so I discard it.

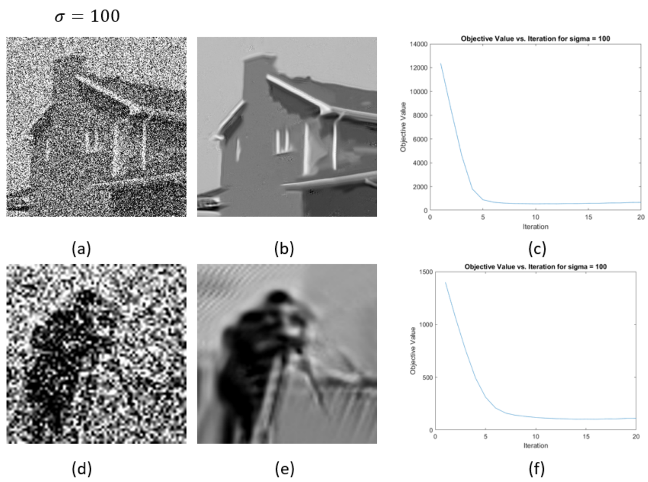

FIGURE 2.1: At $\sigma = 100$, (a) Noise: PSNR, 8.13 db (b) Est: PSNR, 26.67 db (c),Variation of objective function in house img (d) Noise: PSNR, 8.03 db, (e) Est: PSNR, 20.93 db (f) Variation objective function in c-man img

As we can observe, due to high variance and small image size camera-man image is getting total lost but its estimation relatively good. Also objective function variation is same as WNNM, but its saturation level is more close to the 0 as comapared to WSNM, this implies adding Schatten p-norm to the objective function of WNMM is good for minimisation.

Other images experimental results are in chapter 5 in table format.

Chapter 3

Patch-Based Low-Rank Minimization

3.1 Introduction

Patch-based low-rank minimization for image processing attracts much attention in recent years. Patch-based methods, inspired by the inherent non-local self-similarity in natural images, represent a breakthrough. Techniques like K-SVD and BM3D focus on small local patches, using either fixed or adaptive, compact or overcomplete dictionaries. This approach enhances sparsity and significantly improves denoising performance. Despite their success, many of these methods, including K-SVD and BM3D, come with a high computational cost due to their iterative nature.

Low-rank matrix approximation, another prominent method, has been applied to various image processing tasks. It offers a simpler and theoretically grounded approach to denoising, leveraging the nuclear norm minimization as we have seen in WNNM and WSNM.

Building upon these insights, we propose a novel denoising technique known as the patch-based low-rank minimization (PLR) method. PLR simplifies the denoising process by constructing similarity matrices from similar patches and employing PCA, SVD, or low-rank minimization for denoising. This method stands out for its computational efficiency. Experiments demonstrate that the proposed method is rather rapid, and it is effective for a variety of natural grayscale images and color images.

3.2 Proposed Algorithm

Consider the following noise model ;

$$Y = X + E$$

where X is the original image, Y is noise and E is Gaussian noise with mean 0 and standard deviation σ. All $Y, X, E \in R^{M \times N}$, WLOG assume M=N.

Algorithm
Divide the noisy image Y into overlapped patch of size $d \times d$. Denote the set of all these patches as

$$\chi = \{x_i : \ i = 1, 2, \ldots, (N - d + 1)^2\}$$

For each patch $x \in \chi$ called reference patch consider all overlapped patches contained in its $n \times n$ neighborhood (Total patches = $(n - d + 1)^2$).

Then choose the m most similar patch to the reference patch. The similarity is determined by F-norm distance measure.

For each reference patch the similar patches as column vector and put one next to another to form the matrix of size $d^2 \times m$ called similarity matrix.

The similarity matrix is denoted as $S = (s_!, s_2, \ldots, s_m)$, where $s_i, i = 1, 2, \ldots m$, columns of S , are vectored similar patches. Then all the patches in the matrix S are denoised together using the hard thresholding method with the principal component (PC) basis.

For convenience, we assume that the mean of the patches in S, denoted by $s_c := \frac{1}{m}\sum_{l=1}^{m} s_l$ is 0.

Since the patches are overlapped, every pixel is finally estimated as the average of repeated estimates.

The process of denoising the matrix S -.

We derive the basis vector using PCA. The PC basis is the set of the eigen vector of SS^T, its eigenvalue decomposition is

$$SS^T = P\Lambda P^{-1}$$

with

$$P = (g_1, g_2, \dots, g_{d^2}), \Lambda = diag(m\lambda_1^2, m\lambda_2^2, \dots, m\lambda_{d^2}^2)$$

where g_i is the i-th column of P. The PC basis is the set of the columns of P that is ,$(g_1, g_2, \dots, g_{d^2})$.

The noisy patches s_i in the similarity matrix S are denoised as follow:

$$\bar{s_l} = \sum a_k \langle s_l, g_k \rangle g_k, \quad \forall \quad l = 1, 2, \dots, m$$

where

$$a_k = \begin{cases} 1 & if\ \lambda_k^2 > t^2 \\ 0 & \text{otherwise} \end{cases}$$

t being the threshold.
Or equivalently, the matrix composed of estimated patches i.e.

$$\bar{S} := (\bar{s_1}, \bar{s_2}, \dots, \bar{s_m}) = Ph(\Lambda)P^{-1}S$$

with $h(\Lambda) = diag(a_1, a_2, \dots, a_{d^2})$.

Authors proved in paper that the solution which we obtain using PCA is same as the solution of the following low rank minimisation problem

$$\hat{X} = argmin_X ||Y - X||_F^2 + mt^2\text{Rank}(X)$$

Also the choice of threshold t is 1.5σ (Authors mainly emphasis on this part that how they get the threshold theoretically).

Algorithm 4 Patch-based Low-Rank Minimization Denoising Method

Input: Y

Output: X

for each patch x_i in χ **do**

  Find similar patches to form similarity matrix $S = (\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_m)$

  Subtract the mean $\mathbf{s}_c$: $\mathbf{s}_l \leftarrow \mathbf{s}_l - \mathbf{s}_c$ $(l = 1, 2, ..., m)$ with $\mathbf{s}_c = \frac{1}{m}\sum_{l=1}^{m} \mathbf{S}_l$

  Compute the eigenvalue decomposition $SST = PAP^{-1}$

  Get the estimation $\hat{S} = (\hat{\mathbf{s}}_1, \hat{\mathbf{s}}_2, ..., \hat{\mathbf{s}}_m) = Ph(\Lambda)P^{-1}S$

  Add the mean $\mathbf{s}_c$: $\hat{\mathbf{s}}_l \leftarrow \hat{\mathbf{s}}_l + \mathbf{s}_c$ $(l = 1, 2, ..., m)$

  Return $\hat{\mathbf{s}}_1, \hat{\mathbf{s}}_2, ..., \hat{\mathbf{s}}_m$ to original locations, and take averages for repeated estimates

end for

3.3 Results

The only requirement of this algorithm to perform is input image and level of noise σ.

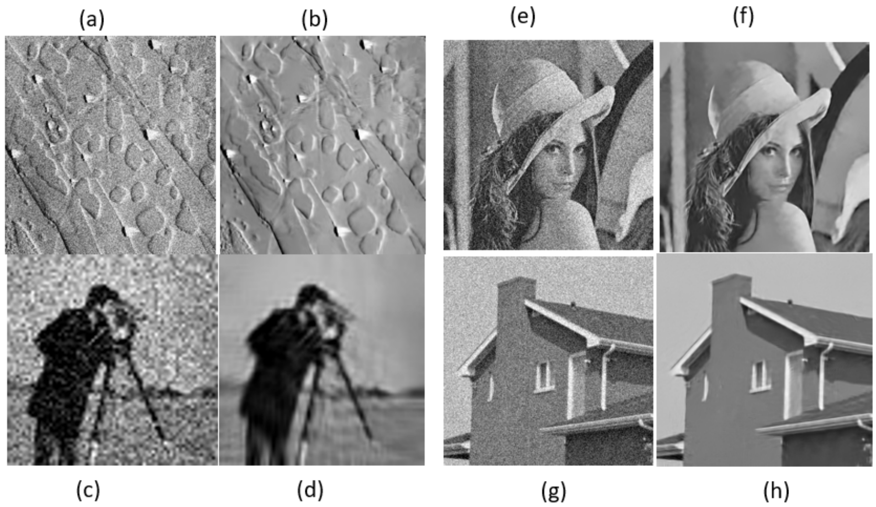

FIGURE 3.1: PLR results: (a) Amplitude noise: $\sigma = 50$,14.14 db (b) estimate , 24.32 db (c) c-man noise:$\sigma = 30$,18.60 db(d) c-man estimate:26.09 db (e) Lena noise:$\sigma = 50$, 14.14 db (f) Estimate,29db, (g) houes noise: $\sigma = 20$,22 db (h) house estimate, 34.04

We can see that this method gives us very good results and competing with results of WSNM and WNNM. This method is straight forward, no use of iterations, conceptually simple, use of PCA. One important point of this algorithm that it is very fast. So we can adapt this algorithm where we require fast result.

Other results is given in chapter-5 in table format.

Chapter 4

Reweighted Low-Rank Matirx Analysis with Structural Smoothness

4.1 Introduction

In this approach, a novel algorithm for low-rank matrix recovery aimed at image denoising is introduced. By integrating the total variation (TV) norm and pixel range constraints with the current reweighted low-rank matrix analysis methods, our aim is to attain structural smoothness.

Why this method :- In existing Low rank matrix analysis methods image structural smoothness and pixel range constraints has not been carefully addressed.

Main contribution of this paper can be summarized as follow.
1. Authors incorporate the TV norm into the reweighted lowrank matrix recovery model, which is able to effectively separate the clean low-rank matrix from the high-density sparse noise while being able to maintain the smoothness of image structures. 2. Authors incorporate the pixel range constraint into the low-rank image recovery model to regulate the recovery of image pixels, which is able to improve the performance of image de-noising. 3. The optimization problem with the TV norm added into the objective function and the pixel range constraint becomes a

challenging non-convex optimization problem. To effectively solve this problem, we have developed a new numerical solution based on inexact augmented Lagrangian multipliers (IALM) and non-uniform singular value thresholding (NSVT).

4.2 Related Work or Definitions

1. ***NNM based Low-Rank Matrix Recovery***
The low-rank minimisation problem can be formulated as:

$$min_{X,E}||X||_* + \lambda||E||_1 s.t. \;\; M = X + E$$

where $M \in R^{m\times n}$ is the observed matrix.

$||X||_*$ is the nuclear norm of matrix X and E is sparse matrix (Noise).

2. ***Reweighted Low-Rank Matrix Recovery***
The reweighted $l1$ norm low-rank matrix recovery method was recently proposed by Deng et al. to enhance the sparsity of the error matrix.

This method aim to minimise the following function

$$min_{X,E}||X||_* + \lambda||W_E \odot E||_1 s.t. \;\; M = X + E$$

Where $W_E \in R^{m\times n}$ is the weight matrix.

3. ***TV Norm-Based Image Restoration***
For a image matrix $X \in R^{m\times n}$ the isotropic TV norm is defined as

$$\begin{aligned} TV(X) = &\sum_{i=1}^{m-1}\sum_{j=1}^{n-1}\sqrt{(x_{i,j}-x_{i+1,j})^2+(x_{i,j}-x_{i,j+1})^2} \\ &+\sum_{i=1}^{m-1}|x_{i,n}-x_{i+1,n}| + \sum_{j=1}^{n-1}|x_{m,j}-x_{m,j+1}| \end{aligned}$$

TV norm used to solve the image de-noising problem due to its capability in preserving edge information and enhance region smoothness.

The problem of image denoising using TV norm

$$\begin{array}{ll} \min_X & \|X - M\|_F^2 + 2\rho\|X\|_{TV} \\ \text{s.t.} & X \in B_{l,u} \equiv \{x_{i,j}, l \leq x_{i,j} \leq u\}, \end{array}$$

where $B_{\{}l, u\}$ is the bounded set, which set the range of pixel value.

4. ***Operators***

Operator -1 Non-uniform soft threshholding NST

Let $X \in R^{m\times n}$ and $W \in R^n_{++}$ the NST operator

$$S_W\{X\} = \{sign(x_i)max((|x_i| - w_i), 0)\}$$

Operator -2 Non-uniform Singular value thresholding NSVT

Given $X \in R^{m\times n}$ is matrix of rank r, its SVD $X = U\Sigma V^T$. And threshhold weight vector $W \in R^r_{++}$, the NSVT operator

$$\mathcal{D}_W\{X\} = US_W\{\Sigma\}V^T$$

where $S_W\{\Sigma\} = diag(max((\sigma_i - w_i), 0)$

4.3 Problem Formulation

Inspired by the low-rank matrix recovery methods, we develop a smoothed low-rank matrix recovery algorithm using the re-weighted nuclear norm, which incorporates the TV norm into the low-rank matrix analysis framework. This new formulation is able to capture the local image correlation and structure.

Mathematically,

$$\begin{gathered} \min_{X,E} \sum_{j=1}^{n} w_{X,j} \cdot \tilde{\sigma}_j + \lambda \left\|W_E \odot E\right\|_1 + \eta\|X\|_{TV} \\ \text{s.t. } X \in B_{l,u} \equiv \{x_{i,j}, l \leq x_{i,j} \leq u\}, \\ M = X + E, \end{gathered}$$

4.4 Proposed Solution

Adding the TV norm into objective function and the pixel range constraint make the problem considerably more challenging. we will develop the iterative method to solve our problem. To solve the the optimisation a new variable is added to the model as follow.

$$\min_{H,X,E} \sum_{j=1}^{n} w_{H,j} \cdot \sigma_j + \lambda \left\| W_E \odot E \right\|_1 + \eta \|X\|_{TV}$$
$$\text{s.t. } X \in B_{l,u} \equiv \{x_{i,j}, l \leq x_{i,j} \leq u\},$$
$$M = H + E, \quad H = X.$$

where $W_H = \{w_H, j\}$ are the weights of for σ_j and σ_j are the singular value of matrix H and $w_{H,j} = w_{X,j}$ for $j = 1, 2, \ldots, n$

authors adjust the optimisation constraint using Lagrange Multiplier, then the augmented Lagrangian function of objective function is constructed as.

$$\begin{aligned} f(H, X, E, Y_1, Y_2) = & \sum_{j=1} w_{H,j} \cdot \sigma_j + \lambda \left\| W_E \odot E \right\|_1 + \eta \|X\|_{TV} \\ & + \langle Y_1, M - H - E \rangle + \langle Y_2, X - H \rangle \\ & + \frac{\mu}{2} \left(\|M - H - E\|_F^2 + \|X - H\|_F^2 \right) \\ & \text{s.t. } X \in B_{l,u} \equiv \{x_{i,j}, l \leq x_{i,j} \leq u\} \end{aligned}$$

where Y_1, Y_2 are dual variables and μ is penalty parameter.

To solve the problem, authors use the iterative alternating direction method, which optimizes one variable while fixing the remaining optimization variables in an iterative manner.

A. **Optimizing H**
If we fix other variables then

$$\hat{H} = argmin_H f(H, X, E, Y_1, Y_2)$$

Define $L = \frac{1}{2}(M + X - E + Y_1/\mu + Y_2/\mu)$, then above problem can be solved by NSVT operator (Detailed proof in paper)

$$\hat{H} = \mathcal{D}_{(2\mu)^{-1}W_H}(L)$$

B. **Optimizing X**

$$argmin_X f(H, X, E, Y_1, Y_2)$$

The solution to above optimization problem can be given by

$$X = P_{B_{l,u}}(R - \rho\mathcal{L}(p, q))$$

which can be computed using Fast Gradient projection (FGP) algorithm[reference must], $\rho = \eta/\mu$ be passed to the FGP algorithm.

C. **Optimizing E**
Optimization of E can be done by NST operator,

$$E = S_{\lambda\mu^{-1}W_E}[M - H + Y_1/\mu]$$

D. **Optimizing $\mathbf{Y_1}$,$\mathbf{Y_2}$**
Y_1 and Y_2 are the Lagrange multiplier matrices of the original optimization problem. They should be updated after the other variables.

If Y_1 is unknown and the other variables are fixed, then Y_1 can be updated as follows:

$$Y_1 = Y_1 + \mu(M - H - E)$$

If Y_2 is unknown and the other variables are fixed, then Y_2 can be updated as follows:

$$Y_2 = Y_2 + \mu(X - H)$$

4.5 Algorithm

Algorithm 5 Smoothed and Reweighted Low-Rank Matrix Recovery (SRL-RMR)

Require: Data matrix $M \in \mathbb{R}^{m\times n}$ (assuming $m \geq n$).

Ensure: Initialize $W_X = (w_{X,j}) \in \mathbb{R}^n, W_E = (w_{E,ij}) \in \mathbb{R}^{m\times n}$, $\varepsilon = 10^{-7}$, $k = 0$ and *maxiter*.

1: Compute $X^{(0)} = U\Sigma V^T$ and $E^{(0)}$ using IALM [reference]. Set $w^{(0)}_{X,j} = \frac{1}{\text{diag}(\Sigma)_j+\epsilon_X}, w^{(0)}_{E,ij} = \frac{1}{|e^{(k)}_{ij}|+\epsilon_E}$.

2: **while** $\|M - X - E\|_F/\|M\|_F > \varepsilon$ and $k < \mathit{maxiter}$ **do**

3: **Step 1:** Using Algorithm 6 with weights $W^{(k)}_X$ and $W^{(k)}_E$ to compute $X^{(k)} \leftarrow X^*$ and $E^{(k)} \leftarrow E^*$;

4: **Step 2:** Update weights: The weights for each $i = 1, \ldots, m$ and $j = 1, \ldots, n$ are updated by

$$w^{(k+1)}_{X,j} = \frac{1}{\sigma^{(k)}_j + \epsilon_X}, w^{(k+1)}_{E,ij} = \frac{1}{|e^{(k)}_{ij}| + \epsilon_E}.$$

where ϵ_X and ϵ_E are predetermined positive constants, and the singular value matrix

$$\Sigma^{(k)} = \text{diag}([\sigma^{(k)}_1, \ldots, \sigma^{(k)}_n]) \in \mathbb{R}^{n\times n}$$

with $[U^{(k)}, \Sigma^{(k)}, V^{(k)}] = \text{svd}(X^{(k)})$.

5: **end while**

Output $X^{(k)}, E^{(k)}$

Algorithm 6 IALM Algorithm for Solving the Problem of Smoothed and Reweighted Low-Rank Matrix Recovery

Require: Data matrix $M \in \mathbb{R}^{m\times n}$, *lambda*, η and δ.

Ensure: Initialize $X_0 \in \mathbb{R}^{m\times n}, E_0 \in \mathbb{R}^{m\times n}, H_0 \in \mathbb{R}^{m\times n}, Y_{1,0} \in \mathbb{R}^{m\times n}, Y_{2,0} \in \mathbb{R}^{m\times n}, \mu_0 > 0, \xi = 10^{-7}, t = 0$ and inneriter $= 100$.

1: **while** $\|M - H - E\|_F/\|M\|_F > \xi$ and $t <$ inneriter **do**

2: **Step 1:** Let $L_{t+1} = M+X_t-E_t+\frac{Y_{1,t}}{\mu_t}+\frac{Y_{2,t}}{\mu_t}, \rho = \frac{\eta}{\mu_t}$; then, $H_{t+1} = \mathcal{D}_{\frac{1}{\mu_t}W_X}(L_{t+1})$;

3: **Step 2:** Let $R_{t+1} = H_{t+1} - Y_{2,t}$; use the FGP Algorithm to compute $X_{t+1} = P_{B_{l,u}}(R_{t+1} - \rho\mathcal{L}(p, q))$;

4: **Step 3:** $E_{t+1} = \mathcal{S}_{\lambda\mu_t^{-1}W_E}\left[M - H_{t+1} + \frac{Y_{1,t}}{\mu_t}\right]$;

5: **Step 4:** $Y_{1,t+1} = Y_{1,t} + \mu_t(M - H_{t+1} - E_{t+1})$;

6: **Step 5:** $Y_{2,t+1} = Y_{2,t} + \mu_t(X_{t+1} - H_{t+1})$;

7: **Step 6:** $\mu_{t+1} = \delta\mu_t$, $t \leftarrow t + 1$;

8: **end while**

Output X^*, E^*

4.6 Experiment Setting

4.6.1 Parameters

The size of the input matrix is $m \times n$. The max iteration of the fast gradient projection (FGP) algorithm [reference] is set to be 2.In Algorithm-5 the constants ϵ_X and ϵ_E is set to be 0.01. The bounds of constraints we set to be $[l, u] = [0, 1]$. $maxiter$ is said to $\{1, 2, 3\}$.

In Algorithm 6, we set $X_0 = 0$, $E_0 = 0$, $H_0 = 0$, $Y1, 0 = /max(||M||, \lambda^{-1}||M||_{\text{inf}})$, and $Y2, 0 = 0$, where $||M||$ is the spectral norm of matrix M and $||M||_{\text{inf}}$ is the maximum absolute value of the entries in matrix M. In addition, $\delta = 1.5$.

In main objective function λ and η represent the tradeoff between those three components. Specifically, they control the sensitivity level of the model to sparse errors, TV norm, and rank of the approximation matrix. Authors suggested λ to be $1/\sqrt{max(m, n)}$. $\eta = 0.00025/\sqrt{max(m, n)}$.. At last, $= \eta/\mu$.

4.7 Results

I have used same test images and same $\sigma's$. When I implemented the algorithm based on above setting, we totally lost our output images. So I have checked algorithm line-by-line and see what happen to the input image. I have found that In Algorithm-6, $inneriter$ should be around $12 - 13$. After update $inneriter = 15$ I get the following results.

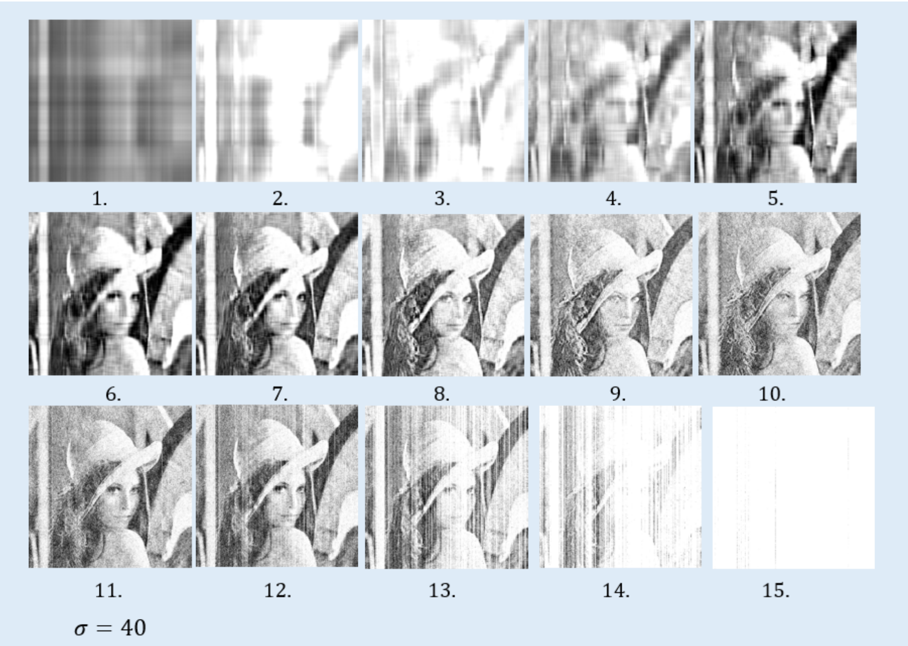

FIGURE 4.1: Enter Caption

I observed that this algorithm tries to destruct the input image first and then start to recover image. The problem that I found in this algorithm is after 12-13 iteration images started vanishing. And PSNR value of estimated image in 12-th value is less then the PSNR of noise image,(mean quality estimated image is worse then noise image).

To check where I found error in denoising we check main objective value fuction after each update of X. In algo-5, X is only updated in IALM function (before entering into while loop), at this step, we get minima and value is around 0. Then after entering in while loop of algo-5, we directed to the algo-6. In algo-6, X is updated in while loop. Below is the objective function vs while loop iteration curve.

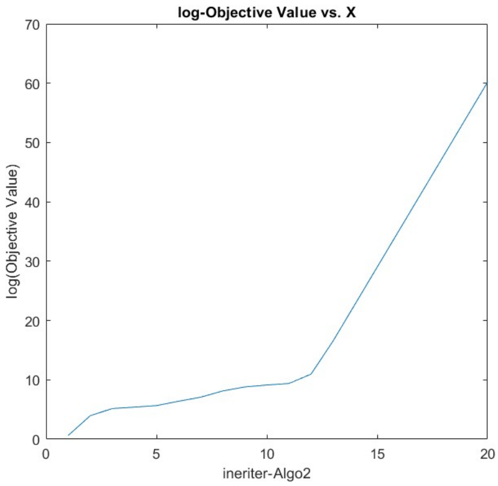

FIGURE 4.2: objective fn variation in algo-6

We can see in above figure that the value is increasing drastically, even after taking the log of value and we see that variation is exponential.

Also this algorithm uses other denoising algorithm like FGP, IALM and NSVT, I have tested that individually those algorithm working fine. Below are some result of FGP of IALM.

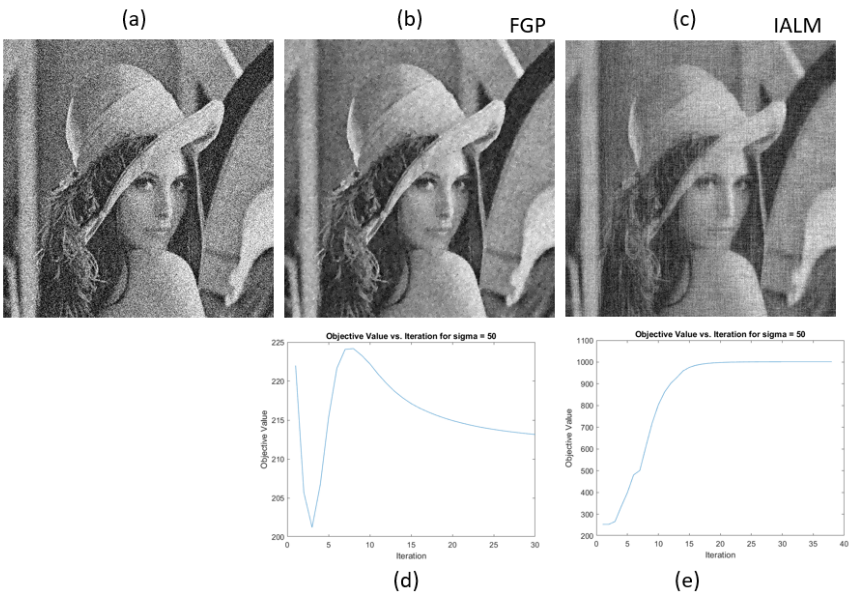

FIGURE 4.3: At σ = 50, (a) Noise image,PSNR:14.14 db (b) FGP estimate, PSNR:21.79 db(c) IALM estimate: PSNR:19.34 db (d) FGP obj. function variation (e) IALM obj fn variation

FGP algortihm at around $N = 2$ attain minima, and we have taken $N = 2$ iteration in our implementation, from FGP there is no problem. But there in IALM we see that function increases then attain saturation, algo-6 is based on IALM, so this might be reason of increase. But on the other side IALM estimation (individualy) is comparatively good, So there is possibility that there is problem in update inside while loop and parameter initialisation.

As a result we discard this algorithm from our experiment. instead of this, we evaluate the test images on FGP and IALM. RLM Algorithm need to be modified, atleast we are sure that there is way to denoise image using this method.

Chapter 5

Comparison

5.1 Comparison

I have combined the result of all implemented algorithm. Below are the result.

Table 5.1: $\sigma = 10$, PSNR : Original & Noise - 28.10 db

Image	WNNM	WSNM	PLR	FGP	IALM
Cameramen	33.25	33.29	32.9	25.13	23.34
House	36.88	36.84	36.53	26.57	27.51
Lena	36.06	36.02	35.91	27.07	27.77
Amplitude	31.28	31.16	31.18	25.11	24.73

Table 5.2: $\sigma = 20$, PSNR : Original & Noise - 22 db

Image	WNNM	WSNM	PLR	FGP	IALM
Cameramen	29.22	29.22	28.69	24.86	22.15
House	34.13	34.04	33.53	27.05	24.96
Lena	33.07	33.08	32.97	26.95	25.08
Amplitude	28.09	28.02	27.96	25	23.09

TABLE 5.3: $\sigma = 30$, PSNR : Original & Noise - 18.60 db

Image	WNNM	WSNM	PLR	FGP	IALM
Cameramen	27.04	27.08	26.09	23.96	20.82
House	32.45	32.63	31.58	26.06	22.76
Lena	31.60	31.42	31.35	25.98	22.75
Amplitude	26.57	26.59	26.25	24.31	21.49

TABLE 5.4: $\sigma = 50$, PSNR : Original & Noise - 14.14 db

Image	WNNM	WSNM	PLR	FGP	IALM
Cameramen	24.45	24.49	23.30	20.93	17.85
House	29.87	30.49	29.08	21.72	19.34
Lena	28.73	29.13	29.00	21.79	19.56
Amplitude	24.66	24.77	24.32	21.08	18.86

TABLE 5.5: $\sigma = 100$, PSNR : Original & Noise - 8.13 db

Image	WNNM	WSNM	PLR	FGP	IALM
Cameramen	20.94	20.91	20.25	13.29	12.99
House	25.59	26.67	25.42	13.36	14.15
Lena	25.24	26.05	25.97	13.40	14.46
Amplitude	21.94	22.43	22.13	13.21	14.27

Observations

1. Results of WNNM and WSNM are approximately similar, one possible reason is that we use $p = 0.95$ which is close to 1 and at $p = 1$ WSNM is same as WNNM.

2. On Microscopic Image, results of WSNM and WNNM are not good as Lena image, perhaps both of same size. We can say that WSNM and WNNM not so good for microscopic images.

3. PLR results are competing with WSNM and WNNM.

4. FGP and IALM results similar but not good as WNNM and WSNM.

Chapter 6

Conclusion

I implemented total 6-denoising algorithms based on optimisation. Also tested their performance on Microscopic Image. The RLM algorithm not working well, its algorithm need to be modify. WSNM and WNMM are time consuming algorithms on the other hand PLR is fast among all of them. FGP and IALM also quiet fast but results are not good. In future work, we can put together all ideas and make more efficient denoising algorithm.

Bibliography

[1] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2862–2869, 2014.

[2] Y. Xie, S. Gu, Y. Liu, W. Zuo, W. Zhang, and L. Zhang, "Weighted schatten p -norm minimization for image denoising and background subtraction," *IEEE Transactions on Image Processing*, vol. 25, no. 10, pp. 4842–4857, 2016.

[3] H. Hu, J. Froment, and Q. Liu, "A note on patch-based low-rank minimization for fast image denoising," *Journal of Visual Communication and Image Representation*, vol. 50, pp. 100–110, 2018.

[4] H. Wang, Y. Cen, Z. He, Z. He, R. Zhao, and F. Zhang, "Reweighted low-rank matrix analysis with structural smoothness for image denoising," *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 1777–1792, 2018.

[5] A. Beck and M. Teboulle, "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems," *IEEE Transactions on Image Processing*, vol. 18, no. 11, pp. 2419–2434, 2009.

[6] X. M. Yuan and J. F. Yang, "Sparse and low-rank matrix decomposition via alternating direction methods," *Online*, 2009.