Practical Tutorial #1

Optimization for Computer Vision (OVO)
CentraleSupélec - Masters in Math and AI

November 16th, 2023

Introduction

In today's lab we will implement some of the methods we have seen for image processing using the techniques from convex optimization that we have covered. Your final submission for this tutorial should be contained in a python jupyter notebook. You should only be using the NumPy and SciPy libraries (and the Image function from the PIL library if you want) - any optimization algorithms or gradients you use must be implemented "by hand".

1 Basic Image Manipulation

1. To begin, load the image stripe.png and compute its Fourier transform. Plot the log of the magnitude of its Fourier transform. Comment on what you find.
2. Load the image cat.png and compute its Fourier transform. Plot the log of the magnitude of its Fourier transform. Comment on what you find and contrast it with the previous image's Fourier transform. What types of signals do these images represent?
3. The Signal-to-Noise Ratio (SNR) of a degraded or approximate image x_d given the ground truth x is defined to be

$$\mathrm{SNR}(x_d, x) := -20 \log_{10} \left(\frac{\|x_d - x\|}{\|x\|} \right)$$

   The units of this quantity are decibels (dB); a high SNR heuristically indicates a better image approximation. Add Gaussian noise to the cat.png image with mean 0 and standard deviation 5 and compute the SNR. Repeat with Gaussian noise that has mean 0 and standard deviation 20. (Note: don't forget to clip the values to $[0, 255]$ as integeres).
4. Using the gaussian_kernel function, generate a point-spread function with kernel size 21 and $\sigma = 3$ for an image of size 790×790. Plot the point-spread function and its Fourier transform using imshow. How does this relate to the identities about the Fourier transform we discussed?
5. Apply the blur filter you computed in step 4 to cat.png.

6. Implement two discretizations of the total variation regularizer - one using forward finite-differences

$$\|\nabla u\|_1 = \sum_{(i,j)\in x} |u_{i,j+1} - u_{i,j}| + |u_{i+1,j} - u_{i,j}|$$

and one using centered finite-differences

$$\|\nabla u\|_1 = \sum_{(i,j)\in x} |u_{i,j+1} - 2u_{i,j} + u_{i,j-1}| + |u_{i+1,j} - 2u_{i,j} + u_{i-1,j}|$$

2 Denoising with the ROF Model

In this section, we will consider denoising a noisy image using the ROF model. The forward model for this problem is that we observe a noisy version of x:

$$x_d = x + \delta$$

where δ is Gaussian white noise. Throughout this section, take cat_noise.png as x_d and cat.png as x. We will solve the ROF model by going to the dual and using either Forward-Backward or FISTA:

Algorithm 1 FISTA

Require: $x^0 \in \mathbb{R}^n$, $y^0 = x^0$ and $\lambda_0 = 1$

for $k = 0, 1, \ldots$ **do**

$x^{k+1} = \mathrm{prox}_{\frac{1}{L}g}\left(y^k - \frac{1}{L}\nabla f(y^k)\right)$

$\lambda_{k+1} = \frac{1+\sqrt{1+4\lambda_k^2}}{2}$

$y^{k+1} = x^{k+1} + \frac{\lambda_k - 1}{\lambda_{k+1}}(x^{k+1} - x^k)$.

end for

Forward-Backward

1. Write the ROF model for this denoising problem.
2. Recover a denoised image using Forward-Backward splitting on the dual of the ROF model using the forward finite-difference discertization of the TV regularizer. Try several different values for the regularization parameter and pick the best one by comparing the SNR using the ground truth. (Note: we can recover a primal solution from a dual solution using the relation $x^\star = x_d - \nabla^* y^\star$ where ∇^* is the adjoint to the discrete gradient in the total variation regularizer.)
3. Recover a denoised image using Forward-Backward splitting on the dual of the ROF model using the centered finite-difference discretization of the TV regularizer. How does this compare with the image recovered using the forward finite-difference discretization?

FISTA

4. Recall the accelerated Forward-Backward algorithm or FISTA that we discussed in Lecture 2. Recover a denoised image using FISTA on the dual of the ROF model. Try several different values for the regularization parameter and pick the best one by comparing the SNR using the ground truth.
5. Compare the plot with the functional values $\phi(y_k) - \phi^\star$ of the dual problem when solved by FISTA with the same thing for Forward-Backward splitting, for the same regularization parameter. What behavior can you see when solving the problem with FISTA? How does this relate to momentum?

Restarting FISTA

Algorithm 2 Restarting FISTA

Require: $x^0 \in \mathbb{R}^n$, $y^1 = x^0$, $\lambda_1 = 1$

1: set $k = 0$
2: **while** $\nabla f(y^{k-1})^T(x^k - x^{k-1}) \leq 0$ **do**
3: $\quad x^{k+1} = \text{prox}_{\frac{1}{L}g}\left(y^k - \frac{1}{L}\nabla f(y^k)\right)$
4: $\quad \lambda_{k+1} = \frac{1+\sqrt{1+4\lambda_k^2}}{2}$
5: $\quad y^{k+1} = x^{k+1} + \frac{\lambda_k - 1}{\lambda_{k+1}}(x^{k+1} - x^k)$.
6: $\quad$ set $k = k + 1$.
7: **end while**
8: set $x^0 = x^k$, $y^0 = x^k$ and $\lambda_0 = 1$.
9: return to 1:

6. The convergence rate of FISTA can be improved using the so-called *restarting rules*. The idea behind restarting is to take the current iterates and start a new instance of FISTA with these iterates as our initialization. We will perform a restart every time the gradient condition

$$\nabla f(y^{k-1})^T(x^k - x^{k-1}) \leq 0$$

fails to be satisfied. What is the significance of this condition, i.e., how can we interpret it geometrically?
6) Apply restarted FISTA using the gradient condition to recover a denoised image using the dual of the ROF model. Compare with previous methods (vanilla FISTA, Forward-Backward). Plot the functional values compared to FISTA and Forward-Backward.

3 Deconvolution Using the Primal-Dual Hybrid Gradient Algorithm

In this section, we will consider deblurring a blurred, noisy image using the variational deconvolution we discussed in class. The forward model for this problem is that we observe the blurred, noisy version of x:

$$x_d = x * h + \delta$$

where δ is Gaussian white noise and h is a point-spread function corresponding to a Gaussian blur. Throughout this section we take x_d to be cat_blur_noise.png and x to be cat.png. We will solve this deconvolution problem using the Primal-Dual Hybrid Gradient method with the total variation regularizer.

PDHG

1. Write the primal deconvolution problem.
2. Derive the primal-dual formulation of this problem using the Lagrangian. You can assume that strong duality holds (i.e., you can interchange mins and maxes).
3. Solve the primal-dual formulation of this problem using the PDHG algorithm. Compare the image you get in terms of SNR for several different values of the regularization parameter.
4. Plot the Lagrangian gap along the iterations: $\mathcal{L}(x_k, \mu^\star) - \mathcal{L}(x^\star, \mu_k)$. How does this compare with the predicted convergence rate?

Nesterov's FGM

5. A priori we cannot apply Nesterov's FGM to this problem because the total variation regularizer is not smooth. Replace the total variation by a smoothed version

$$\|\nabla u\|_1 \sim \sum_{(i,j)\in x} \sqrt{(x_{i,j+1} - x_{i,j})^2 + (x_{i+1,j} - x_{i,j})^2 + \epsilon}$$

   for some positive ϵ very small and apply Nesterov's FGM to this problem. Plot the functional values along the iterations.
6. Compare the results of Nesterov's FGM on the smoothed problem to those of the PDHG.