ECE6560 Final Project

Super Resolution

Hugo BLECH

April 27, 2018

Contents

List of Figures

1 Introduction

In all digital signal processing applications, a signal with a high resolution is often desired for later analysis. In image processing, having a high resolution image gives more details for human interpretation, and helps machine perception algorithms to characterize the image. Here, we will focus only on the spatial resolution of an image, although we could also consider other metrics, such as spectral resolution.

But getting high resolution images can be a hard task. Building a high resolution sensor compatible with most applications is prohibitively expensive and impractical. The resolution of a digital sensor is in part limited by the worst resolution of its optics and its digital sensor. A perfect digital sensor will not be able to capture more detailed information than what the lens is letting through. The same reasoning can be said of a perfect lens with a imperfect sensor: a system will not be better than the worst of its components.

Super resolution techniques are techniques that produce high resolution (HR) images from a set of low resolution (LR) images. They have become popular as low-resolution sensors are easier and cheaper to produce in quantity. This technique combines multiple LR images to form a single HR image. This process is close to single-image interpolation, used to increase the image size. However, in super-resolution, combining multiple LR images gives more detailed information that is not present in a single image interpolation. Reconstructing an HR image from LR images has proven useful in many contexts and applications, such as medical X-ray CT and MRI imaging, and surveillance video.

2 Mathematical formulation

2.1 Low resolution image generation

In super resolution, a model relating the desired HR image to the LR images is necessary to operate a reconstruction. This model can be approximated from the LR images themselves, but this approach is not considered here. To make LR images generation simpler, a single HR image is used. Each LR frame corresponds to a down-sampled version of the HR image. Noise is added on every LR image to simulate a noisy, low-resolution sensor.

Let X be the original HR frame from which the LR frames Y_k are generated. The relation between the HR image and the LR images is:

$$Y_k = D_k B_k M_k X + N_k \quad \text{for } 1 \leq k \leq p$$

where M_k is a geometric warp matrix, representing geometric deformation, B_k is a blurring matrix, which accounts for the point spread function (PSF) of the camera sensor, and D_k is a down-sampling matrix, responsible for representing the loss of spatial resolution. N_k is a matrix representing a normally distributed additive noise.

In the rest of this work, the matrices B_k and M_k were set to the identity matrix: there was no blurring or geometric warping in the LR images.

2.2 Estimating the HR image

To estimate the HR image, we produce HR images X_n the same size as the original HR one X. We try to estimate X_n by minimizing the energy:

$$E_{DF}(I) = \sum_{k=1}^{p} ||D_k B_k M_k I - Y_k||^2$$

Minimizing this energy is a simple way to get as close as possible to the original HR image, given the information provided by the LR images Y_k. To compute this energy, the desired HR image is down-sampled to the same size as the LR images Y_k. Minimizing the square norm of the difference of these matrices is equivalent to reduce the distance between them. The resulting HR image should closely resemble the original HR image.

However, if this energy was the only one considered as a super-resolution reconstruction solution, numerous pixels in the reconstructed image would never have received any information from the LR images. Hence, multiple pixels of the reconstructed image would be zero, and give a poor image quality as a result.

Here is an example of such a reconstruction. The original image is a hedgehog's head. It was chosen because it presents multiple desirable features to test: this image is small, and has smooth surfaces as well as sharp edges. It is therefore difficult to reconstruct correctly. The attributes that make this image interesting are further discussed in section 5.

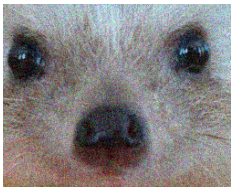
We generate 8 LR images from our original HR image as shown in figure 1. Each of these LR images is sub-sampled from the HR image by a factor of 2. We then add a random noise with 0 mean, and its variance is set to 10% of the maximum amplitude of the original image. As said before, no blurring or geometric warping has been introduced.

2.3 Choice of regularization

Filling in the remaining pixels of the HR image can be seen as a smoothing and denoising process, which is well known in image processing. The image to reconstruct has several features that we want to preserve: sharp edges and smooth surfaces. But reconstructing an HR image from LR images induces a lot of noise in the resulting image.

Using a denoising regularization, we can prevent the presence of noise in the resulting HR image. We chose to use TV regularization for this purpose. It is an approach that preserves sharp edges in denoising by reducing the L1 norm of the gradient of the image. It can be written as the

(a) Example of original HR image



(b) Example of a sub-sampled noisy LR image

Figure 1: Example of degradation to create 8 low resolution images

following energy functional to minimize:

$$E_{TV}(I) = \int_\Omega |\nabla I| dxdy = \int_\Omega \sqrt{||\nabla I||^2 + \beta^2} dxdy$$

Here, the parameter β is used to ensure differentiability of the energy functional when the gradient ∇I is small. The presence of random noise in the image induces a high gradient, and TV regularization is known to successfully reduce this noise while preserving edges in the image.

However, TV regularization is also known to induce a "stair-case effect" in smooth regions. This approach is not best suited for regions where the gradient is small. In these regions, a smoothing regularization would render more realist surfaces. This is why we also used a smoothing regularization in addition to TV regularization.

To smooth the image, some literature recommend using fourth order partial differential equations (PDE), as they recover smoother surfaces. An attempt was made using a linear fourth order equation. However, this equation is highly non-linear and deriving a stability condition was too complex. Instead, a second order linear equation is used to smooth the image. The smoothing used is the heat equation:

$$E_S(I) = \int_\Omega ||\nabla I||^2 dxdy$$

This energy is more adapted than TV regularization to recover smooth surfaces, as it decays in a quadratic form when the gradient is small. This enables us to use distinctively the TV regularization when the gradient is high, e.g. when denoising is required, or near edges, and to use the quadratic smoothing regularization in regions where the gradient is small, e.g. the surface is smooth.

2.4 Adaptive regularization

In order to adapt the regularization to the image, we compute a weighting function τ that has values in $[0, 1]$. To emphasize the restoration properties of TV, the weighting function τ is 1 along edges, and to emphasize the smoothing, the weighting function τ is $0 \leq \tau < 1$ in smooth regions. The regularization term can be written as:

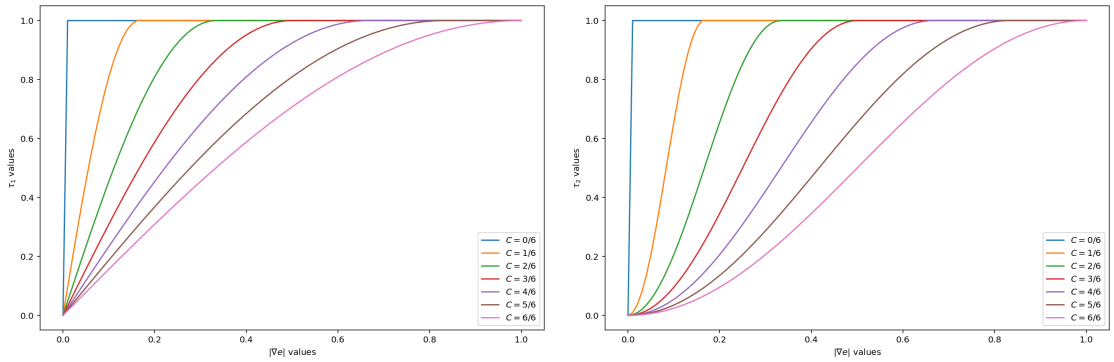
$$E_{reg}(I) = \tau E_{TV}(I) + (1 - \tau) E_S(I)$$

Two weighting functions have been tested in this project, τ_1 and τ_2. We reduce the gradient of the image in the interval $[0, 1]$ by dividing it by its maximum to give ∇e. C is a parameter that varies with our solution.

$$\tau_1 = \begin{cases} 1 & \text{if } |\nabla e| \geq C \\ sin(\frac{\pi |\nabla e|}{2C}) & \text{if } |\nabla e| < C \end{cases}$$

$$\tau_2 = \begin{cases} 1 & \text{if } |\nabla e| \geq C \\ \dfrac{1 - cos(\frac{\pi|\nabla e|}{2C})}{2} & \text{if } |\nabla e| < C \end{cases}$$

As before, varying C gives different behaviors for the same image. The functions τ are represented on figure 2.



(a) Weighting function τ_1

(b) Weighting function τ_2

Figure 2: Functions τ_1 and τ_2 for values of C in $[0, 1]$

These schemes share some characteristics. For example, they are all parametric. This allows us to tune the influence of smoothing and denoising while reconstructing the HR image. We are interested in the difference of behavior as $|\nabla e|$ tends to zero. In τ_1, the function is nearly linear in $|\nabla e|$, while the function τ_2 is quadratic in the neighborhood of 0. As the gradient of the image tends to zero, the image will be more aggressively smoothed by τ_1 that it is by τ_2. We will study the effects in the results.

We introduce λ as the parameter weighting the influence of regularization compared to reconstruction. Finally, the energy to minimize is:

$$E(I) = \sum_{k=1}^{p} ||D_k B_k M_k I - Y_k||^2 + \lambda \left(\tau \int_\Omega \sqrt{||\nabla I||^2 + \beta^2} dxdy + (1 - \tau) \int_\Omega ||\nabla I||^2 dxdy \right)$$

3 Gradient descent equation

In this part, we will describe how the gradient descent equation is derived from the energy functional.

3.1 Energy gradients

The energy described before is made of two components. We expose how to easily get to the energy gradient by considering each of these two components of the energy.

The first part, $\sum_{k=1}^{p} ||D_k B_k M_k I - Y_k||^2$, describes the difference between the HR image and the LR pictures. It is not a geometric energy, and does not depend on any gradient of the image. We show in the annex that the gradient of a squared norm is: $\nabla_X ||AX + B||^2 = 2A^T(AX + B)$. So the gradient of the data fidelity term is:

$$\nabla_I E_{DF}(I) = \sum_{k=1}^{p} 2(D_k B_k M_k)^T (D_k B_k M_k I - Y_k) = 2 \sum_{k=1}^{p} I - M_k^T B_k^T D_k^T Y_k)$$

The second part is the regularization part. Here, the energy functionals are geometric. The gradient of these energies must be calculated using a Lagrangian. As these equations are linear, we can derive the equations separately for simplicity. For the TV regularization:

$$E_{TV}(I) = \int_\Omega \sqrt{||\nabla I||^2 + \beta^2} dxdy = \int_\Omega L(I, I_x, I_y, x, y) dxdy$$

The Lagrangian in this energy is $L(I, I_x, I_y, x, y) = \sqrt{I_x^2 + I_y^2 + \beta^2}$. As usual, we can compute the derivatives of the Lagrangian according to I, I_x, and I_y:

$$\begin{aligned} L_I &= 0 \\ L_{I_x} &= \frac{I_x}{\sqrt{I_x^2 + I_y^2 + \beta^2}} \\ L_{I_y} &= \frac{I_y}{\sqrt{I_x^2 + I_y^2 + \beta^2}} \end{aligned}$$

The gradient of the TV regularization energy can be simply deducted from these quantities:

$$\begin{aligned} \frac{\partial}{\partial I_x} L_{I_x} &= \frac{I_{xx} I_y^2 - I_x I_y I_{xy} + \beta^2 I_{xx}}{(I_x^2 + I_y^2 + \beta^2)(3/2)} \\ \frac{\partial}{\partial I_y} L_{I_y} &= \frac{I_{yy} I_x^2 - I_x I_y I_{xy} + \beta^2 I_{yy}}{(I_x^2 + I_y^2 + \beta^2)(3/2)} \end{aligned}$$

Finally,

$$\begin{aligned} \nabla_I E_{TV}(I) &= 0 - \frac{\partial}{\partial I_x} L_{I_x} - \frac{\partial}{\partial I_y} L_{I_y} \\ \nabla_I E_{TV}(I) &= -\frac{I_{xx} I_y^2 - 2 I_x I_y I_{xy} + I_{yy} I_x^2 + \beta^2 (I_{xx} + I_{yy})}{(I_x^2 + I_y^2 + \beta^2)(3/2)} \end{aligned}$$

Similarly for the regularization using the heat equation, we define the Lagrangian from the energy definition $E_S(I) = \int_\Omega ||\nabla I||^2 dxdy$: $L(I, I_x, I_y, x, y) = I_x^2 + I_y^2$. Its derivatives are:

$$\begin{aligned} \frac{\partial}{\partial I} L_I &= 0 \\ \frac{\partial}{\partial I_x} L_{I_x} &= \frac{I_{xx} I_y^2 - I_x I_y I_{xy}}{I_x^2 + I_y^2} \\ \frac{\partial}{\partial I_y} L_{I_y} &= \frac{I_{yy} I_x^2 - I_x I_y I_{xy}}{I_x^2 + I_y^2} \end{aligned}$$

The gradient of this smoothing energy is:

$$\nabla_I E_S(I) = 0 - \frac{\partial}{\partial I_x} L_{I_x} - \frac{\partial}{\partial I_y} L_{I_y}$$
$$\nabla_I E_S(I) = -\frac{I_{xx} I_y^2 - 2 I_x I_y I_{xy} + I_{yy} I_x^2}{I_x^2 + I_y^2}$$

3.2 Equation

The gradient descent equation that we will apply to reconstruct HR images $X_0, X_1, ...$ is:

$$I_t = -(\nabla_I E_{DF}(I) + \lambda(\tau \nabla_I E_{TV}(I) + (1-\tau)\nabla_I E_S(I))$$

which gives

$$I_t = -(2\sum_{k=1}^{p} I - M_k^T B_k^T D_k^T Y_k) + \lambda(\tau(-\frac{I_{xx} I_y^2 - 2 I_x I_y I_{xy} + I_{yy} I_x^2 + \beta^2 (I_{xx} + I_{yy})}{(I_x^2 + I_y^2 + \beta^2)(3/2)})$$
$$+ (1-\tau)(-\frac{I_{xx} I_y^2 - 2 I_x I_y I_{xy} + I_{yy} I_x^2}{I_x^2 + I_y^2})))$$
$$I_t = -(\sum_{k=1}^{p} I - M_k^T B_k^T D_k^T Y_k) + \lambda\tau \frac{I_{xx} I_y^2 - 2 I_x I_y I_{xy} + I_{yy} I_x^2 + \beta^2 (I_{xx} + I_{yy})}{(I_x^2 + I_y^2 + \beta^2)(3/2)}$$
$$+ \lambda(1-\tau)\frac{I_{xx} I_y^2 - 2 I_x I_y I_{xy} + I_{yy} I_x^2}{I_x^2 + I_y^2}$$

4 Discretization

The previous equation depends on the image I and its spatial and temporal derivatives. This equation can be approximated using differentiated approximations of the derivatives of I. The discretization scheme presented here is Forward Time, Central Space (FTCS) differentiation. As we have seen in class, the heat equation and TV regularization can be stable under this scheme. A suitable stability condition should exist when using a linear combination of these equations.

4.1 Discretization of the derivatives

We approximate $I_t, I_x, I_y, I_{xx}, I_{yy}$ using central differences as:

$$
\begin{aligned}
I_t(x,y,t) &= \frac{I(x,y,t+\Delta t) - I(x,y,t)}{\Delta t} \\
I_x(x,y,t) &= \frac{I(x+\Delta x,y,t) - I(x-\Delta x,y,t)}{2\Delta x} \\
I_y(x,y,t) &= \frac{I(x,y+\Delta y,t) - I(x,y-\Delta y,t)}{2\Delta y} \\
I_{xx}(x,y,t) &= \frac{I(x-\Delta x,y,t) - 2I(x,y,t) + I(x+\Delta x,y,t)}{\Delta x^2} \\
I_{yy}(x,y,t) &= \frac{I(x,y-\Delta y,t) - 2I(x,y,t) + I(x,y+\Delta y,t)}{\Delta y^2}
\end{aligned}
$$

However, these expressions do not hold along the borders of the image. Let N_1, N_2 the dimensions of I. Along the borders of I, I_x, I_y, I_{xx}, I_{yy} become:

$$
\begin{aligned}
I_x(0,y,t) &= \frac{-3I(0,y,t) - I(1,y,t) + 4I(2,y,t)}{2\Delta x} \\
I_x(N_1,y,t) &= \frac{-3I(N_1,y,t) - I(N_1-1,y,t) + 4I(N_1-2,y,t)}{2\Delta x} \\
I_y(x,0,t) &= \frac{-3I(x,0,t) - I(x,1,t) + 4I(x,2,t)}{2\Delta y} \\
I_y(x,N_2,t) &= \frac{-3I(x,N_2,t) - I(x,N_2-1,t) + 4I(x,N_2-2,t)}{2\Delta y} \\
I_{xx}(0,y,t) &= \frac{2I(0,y,t) - 5I(1,y,t) + 4I(2,y,t) - I(3,y,t)}{\Delta x^2} \\
I_{xx}(N_1,y,t) &= \frac{2I(N_1,y,t) - 5I(N_1-1,y,t) + 4I(N_1-2,y,t) - I(N_1-3,y,t)}{\Delta x^2} \\
I_{yy}(x,0,t) &= \frac{2I(x,0,t) - 5I(x,1,t) + 4I(x,2,t) - I(x,3,t)}{\Delta y^2} \\
I_{yy}(x,N_2,t) &= \frac{2I(x,N_2,t) - 5I(x,N_2-1,t) + 4I(x,N_2-2,t) - I(x,N_2-3,t)}{\Delta y^2}
\end{aligned}
$$

This allows us to approximate derivatives using $\mathcal{O}(h^2)$ precision. Since one of the most important features of the regularization is smoothing the image to fill in the pixels that were not assigned, it is useful to ensure that a quadratic precision is used. That way, a empty pixel whose neighbors are not empty will rapidly converge to a value close to the true one.

4.2 Stability of the discretization

We will now analyze the stability of the previous PDE using Von Neumann stability analysis.

To simplify this equation without compromising the validity of the analysis, it is useful to note that the data fidelity terms does not diverge under any circumstance, as it tries to assign a constant value to the reconstructed image. It can then be taken out of the stability analysis.

Then, note that the TV regularization term does not diverge as $||\nabla I||$ tends to 0. In that case, the gradient of $E_{TV}(I)$ becomes:

$$\nabla_I E_{TV}(I) \sim \frac{1}{\beta}(I_{xx} + I_{yy})$$

which is a version of the linear heat equation. The equation becomes:

$$I_t = \lambda\tau\frac{1}{\beta}(I_{xx} + I_{yy}) + \lambda(1-\tau)\frac{I_{xx}I_y^2 - 2I_xI_yI_{xy} + I_{yy}I_x^2}{I_x^2 + I_y^2}$$

Finally, applying Von Neumann analysis on this equation yields the following relation:

$$\begin{aligned} I(\omega,\phi,t+\Delta t) &= I(\omega,\phi,t) + \Delta t(\lambda\tau(\frac{2}{\beta\Delta x}(cos(\omega\Delta x)-1) + \frac{2}{\beta\Delta y}(cos(\phi\Delta y)-1))I(\omega,\phi,t) \\ &\quad + \lambda(1-\tau)(cos(\omega\Delta x) - 1 + cos(\phi\Delta y) - 1)) \\ I(\omega,\phi,t+\Delta t) &= (1 + \Delta t\frac{4\lambda\tau}{\beta}(cos(\omega\Delta x)-1) + 2\Delta t\lambda(1-\tau)(cos(\omega\Delta x)-1))I(\omega,\phi,t) \end{aligned}$$

It follows that the coefficient $\alpha(\omega)$ that governs the stability of the discretization of the equation is:

$$\alpha(\omega) = 1 + \Delta t\frac{4\lambda\tau}{\beta}(cos(\omega\Delta x)-1) + 2\Delta t\lambda(1-\tau)(cos(\omega\Delta x)-1)$$

Since α is a real function, it evolves between its maximum, 1, and its minimum, $1 - \Delta t\frac{8\lambda\tau}{\beta} - 4\Delta t\lambda(1-\tau)$. Therefore, to have the condition $|\alpha(\omega)| \leq 1$, Δt must respect the following condition:

$$\begin{aligned} 1 - \Delta t\frac{8\lambda\tau}{\beta} - 4\Delta t\lambda(1-\tau) &\geq -1 \\ \Delta t &\leq \frac{1}{\lambda(\frac{4}{\beta}\tau + 2(1-\tau))} \end{aligned}$$

Here, since the weighting coefficient τ is not fixed, but varies in $[0,1]$, we have

$$\Delta t \leq \frac{1}{\lambda\max(\frac{4}{\beta}, 2)}$$

This is the CFL condition of the adaptive weighted regularization scheme. As we expected, this scheme can be stable, as a it is a linear combination of linear equations.

5 Experiments

5.1 Image used in this study

The image studied in this report pictures a hedgehog, and is taken from an online source[1].

It was chosen because of the variety of its features : its spines present many edges, which provide regions of high gradients. There are also smooth regions, such as the eyes and the nose of the hedgehog. Some areas are relatively smooth but present a very fine texture, such as the watermelon. Adding noise to this area will make it difficult to reconstruct via super-resolution.

We study a small Region of Interest (ROI) of the initial image, to help with the computation speed and the analysis. This ROI is centered on the face of the hedgehog: it presents small spines and smooth areas. This region is also in the focus of the camera, and presents more details than the rest of the image, which is slightly out of focus.

To measure the deformation of the reconstructed image compared to the original one, three metrics will be used. The Mean Squared Error (MSE) measures the absolute distance between the two images, and represents the absolute error made during the reconstruction process. For a good quality, it should be as low as possible. The Peak Signal-to-Noise Ratio(PSNR) is linked to the MSE. It evaluates the influence of noise in the image, and is expressed in decibels (dB). The higher the PSNR, the lower the noise disturbs the image. A more sophisticated measure is the Structural Similarity Index. It compares the difference of perception between two images, by measuring how the structural difference between images is perceived. This index has values between 0 and 1, and exactly similar images have a SSIM of 1.

Figure 3 shows the high resolution (2048x1598 pixels) hedgehog image, and the selected region of interest (around the face).

(a) HR hedgehog image

(b) ROI selected

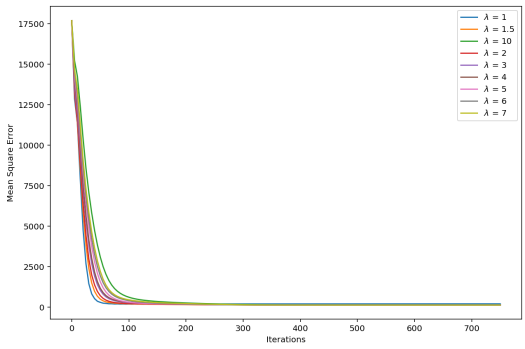
Figure 3: Example of degradation to create 8 low resolution images

5.2 Data fidelity versus regularization

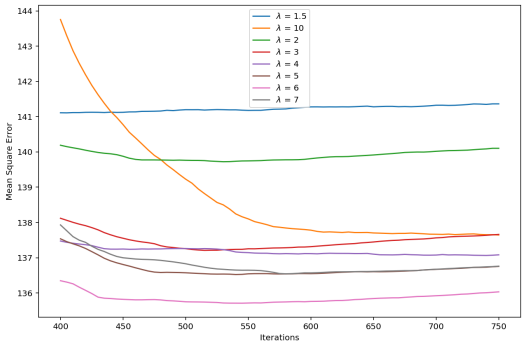
In this section, we will study the quality of the image reconstructed by super resolution while varying the coefficient λ. Modifying this parameter allows us to know the relative importance of the regularization term compared to the data fidelity term in the gradient descent equation to get a good image. We can then find the optimal value of λ for our model.

The values of λ tested are $1, 1.5, 2, 3, 4, 5, 6, 7, 10$. The other parameters of the experiment were $\beta = 0.8$, $\tau = \tau_1$, $C = \frac{1}{4}$. The time step dt was chosen according to the parameters to be as big as possible while keeping a stable equation. We measured for each experiment the MSE, PSNR and SSIM of the resulting SR reconstruction compared to the original one. The following figures on the left show the results, and the figures on the right show a zoom of the end of the graph.

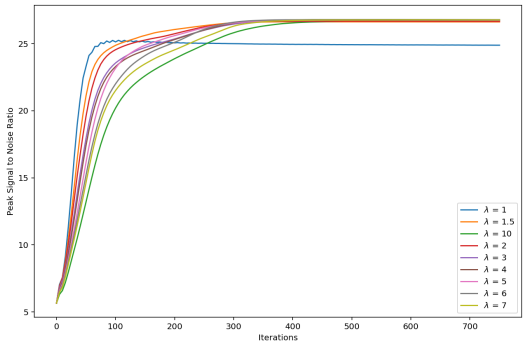
[1] https://www.pinterest.com/pin/315814992604976841/



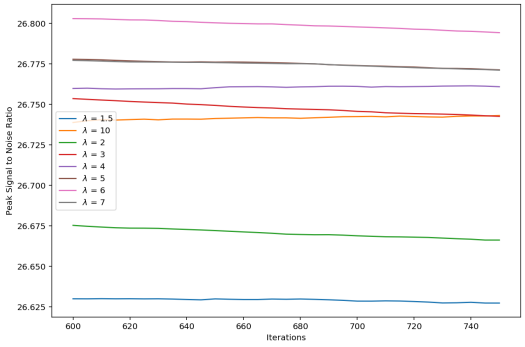
(a) Evolution of MSE during gradient descent



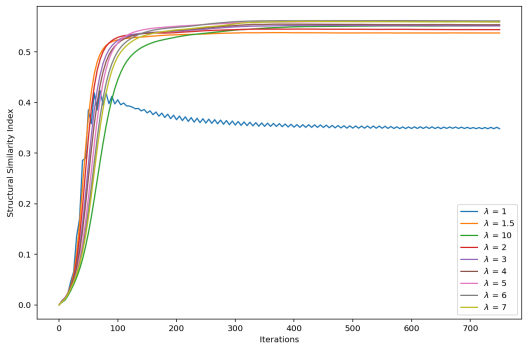
(b) Evolution of MSE during the end of gradient descent



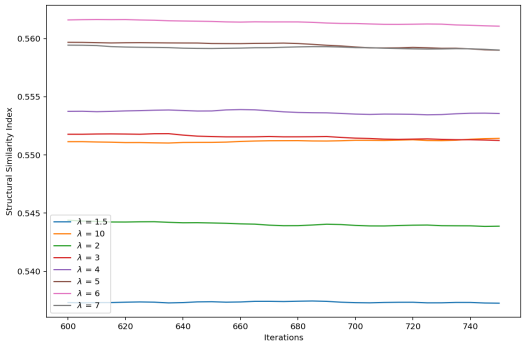
(c) Evolution of PSNR during gradient descent



(d) Evolution of PSNR during the end of gradient descent



(e) Evolution of SSIM during gradient descent



(f) Evolution of SSIM during the end of gradient descent

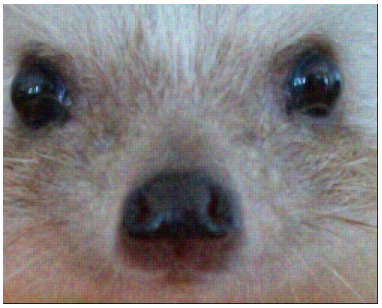
Figure 4: Results of the first experiment

The graphs in Figure 4 show that all the experiments converged to a minimum MSE after 750 iterations but $\lambda = 1$. This particular value shows worse results: its PSNR and SSIM are worse than the other values of λ. This particular value, which weights the data fidelity term and the regularization term evenly, seems to show the conflicting influence of the two terms. Apart from this term, the convergence speed of all λ values is similar.

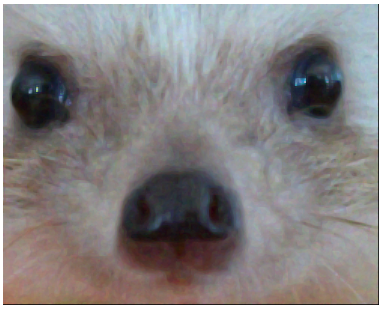
We notice on the zoom figures that the best λ value is $\lambda = 6$, with some good error metrics: $MSE =$, $PSNR = 26.8dB$, $SSIM = 0.56$. The similarity index is not very high, even if our image looks correct, because of the loss of details during reconstruction. For all future experiments, we fix the parameter $\lambda = 6$.

The pictures below are the original, and the reconstructions after 750 iterations with $\lambda = 1$ and $\lambda = 6$. A higher noise level is visible in the picture with $\lambda = 1$.

(a) Original HR image



(b) Reconstruction with $\lambda = 1$



(c) Reconstruction with $\lambda = 6$

Figure 5: Original HR image and reconstructions with $\lambda = 1$ and $\lambda = 6$

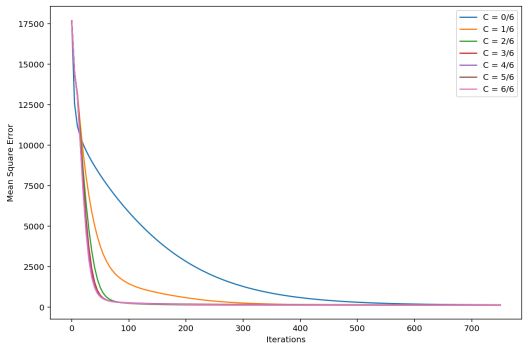
5.3 Comparing adaptive weighting schemes

Here, we test the reconstructed images quality for both weighting functions τ_1 and τ_2 for different values of C, and we compare the results to find the best configuration for our experiment.

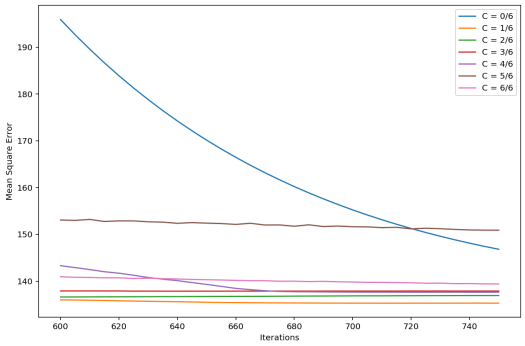
Similarly, we run 750 iterations of gradient descent with $\lambda = 6$, $\beta = 0.8$, $dt = 0.033$. We measured the distortion of the reconstruction using the same metrics as before, and the layout of our presentation is similar.

Both weighting schemes show similar evolution. In particular, the convergence speed of all configurations is roughly the same, and all solutions reach similar results. As we can see on the graphs on the right, the final convergence is not fully reached yet.

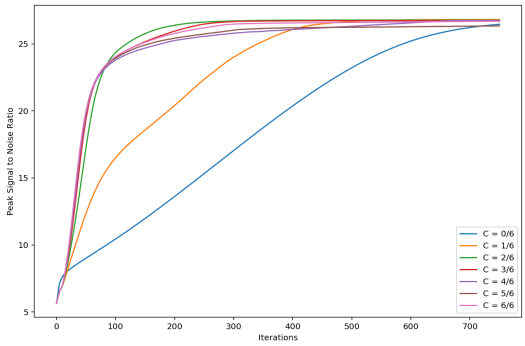
Comparing all final values, we find that the configuration $C = \frac{1}{6}$ produce both the lowest MSE and the highest SSIM in both schemes. The results are slightly better with τ_1 than with τ_2. In figure 8 are the pictures of the original ROI, the result of 200 iterations using $\tau_1(C = \frac{1}{6})$, and the result after 750 iterations using $\tau_1(C = \frac{1}{6})$.



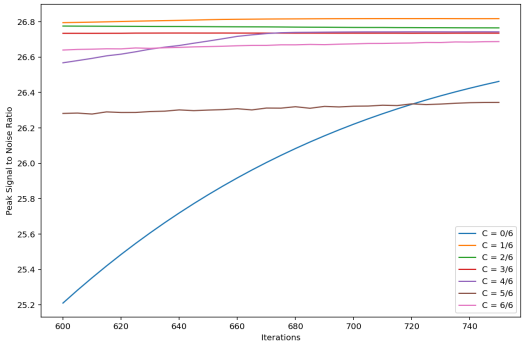
(a) Evolution of MSE during gradient descent



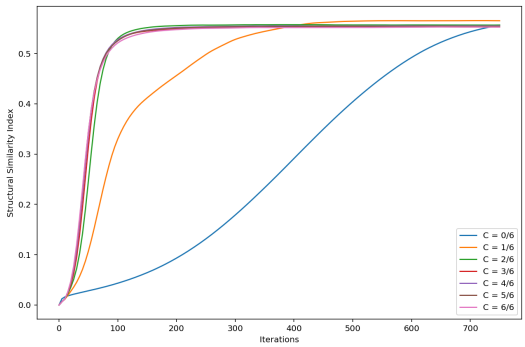
(b) Evolution of MSE during the end of gradient descent



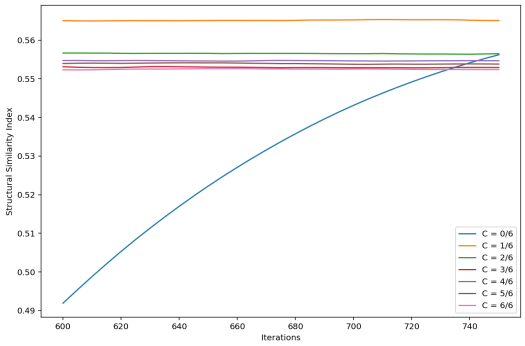
(c) Evolution of PSNR during gradient descent



(d) Evolution of PSNR during the end of gradient descent

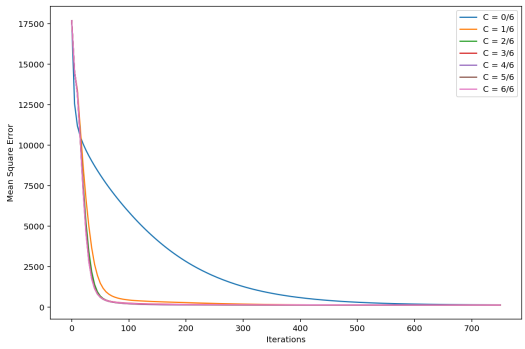


(e) Evolution of SSIM during gradient descent

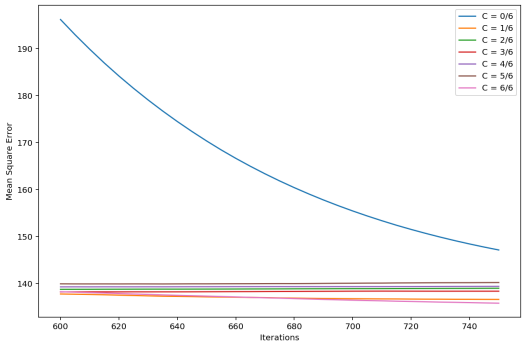


(f) Evolution of SSIM during the end of gradient descent

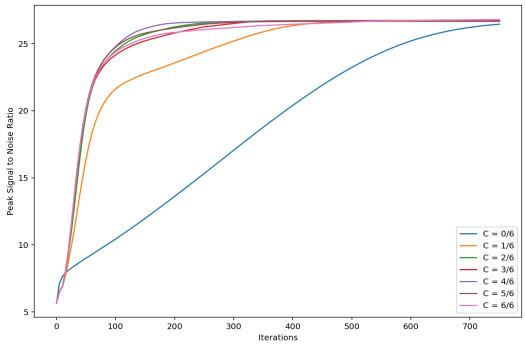
Figure 6: Results of the τ_1 experiment



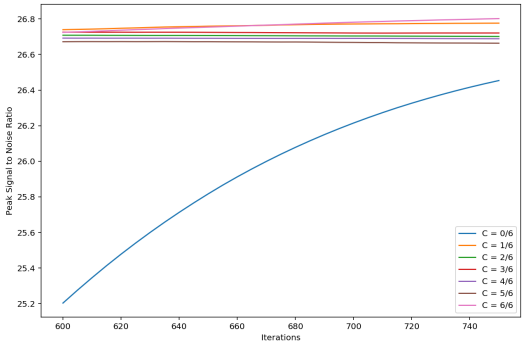
(a) Evolution of MSE during gradient descent



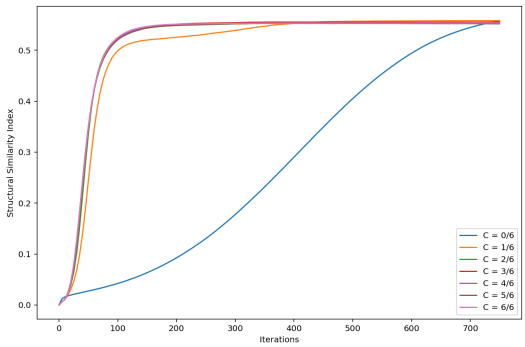
(b) Evolution of MSE during the end of gradient descent



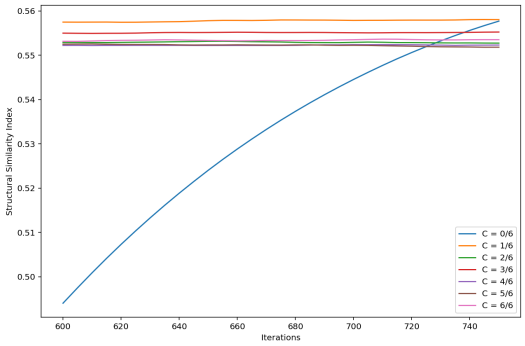
(c) Evolution of PSNR during gradient descent



(d) Evolution of PSNR during the end of gradient descent



(e) Evolution of SSIM during gradient descent

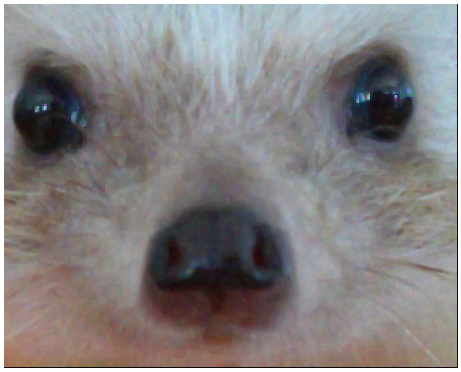


(f) Evolution of SSIM during the end of gradient descent

Figure 7: Results of the τ_2 experiment

(a) Original HR image

(b) Reconstruction with 200 iterations



(c) Reconstruction with 750 iterations

Figure 8: Original HR image and reconstructions with $\tau = \tau_1(C = \frac{1}{6})$ at 200 iterations and 750 iterations

6 Conclusion

In this project, we have designed and implemented a super resolution algorithm using variational gradient descent. We also estimated a good set of parameters to reach a satisfying super-resolution reconstruction of our original image. Our super-resolution algorithm uses both TV regularization for denoising and the heat equation for smoothing, and we compared the efficiency of two weighting schemes.

We did not succeed to reach a high similarity index because of the quantity of information that gets destroyed when the image is subsampled and distorted with noise. Furthermore, no shifting was added to the image before being subsampled : all LR images correspond to the same pixels in the reconstructed HR image. This might limit the impact of noise on the reconstruction, but also limits the total information available about the original image.

With more time, the following improvements could be put in place:

- add more iterations to the experience, to reach the minimum of the gradient descent for all experiments.
- add shifting to the LR images before sub-sampling them. This would include more information in the reconstructed HR image, and provide better results than the current solution.
- implement and try the fourth order PDE for smoothing, to see how best higher-order PDEs are at smoothing surfaces.
- test the current algorithm on harder regions of interest, such as the watermelon.

7 Annex

7.1 Annex 1: Gradient of norm of matrix

Let A be a matrix of size (m, n), X a vector $(n, 1)$ and B a vector of size $(m, 1)$. We will prove that $\nabla_X(||AX + B||^2) = 2A^T(AX + B)$. The dot product of two matrices A and B, given that their sizes match, is A^TB. We will use the definition of the gradient for this proof: the gradient of $||AX + B||^2$ according to X is the vector G such that, for every vector Y,

$$\lim_{h \to 0} \frac{||AX + B + hAY||^2 - ||Ax + B||^2}{h} = G$$

Let Y be another vector of size $(n, 1)$, and h a real. Note that

$$\begin{aligned} ||AX + B + hAY||^2 - ||Ax + B||^2 &= (AX + B + hAY)^T(AX + B + hAY) - (AX + B)^T(AX + B) \\ &= h(AX + B)^T(AY) + h(AY)^T(AX + B) + h^2(AY)^T(AY) \end{aligned}$$

So the gradient verifies the equalities:

$$\begin{aligned} (\nabla_X||AX + B||^2, Y) &= \lim_{h \to 0} \frac{h^2||AY|^2 + h(AX + B, AY)}{h} \\ &= 2(AX + B, AY) \\ &= 2(AY)^T(AX + B) \\ &= 2Y^T(A^T(AX + B)) \\ &= (2A^T(AX + B), Y) \end{aligned}$$

So finally, the gradient of the square norm of $AX + B$ according to X is

$$\nabla_X(||AX + B||^2) = 2A^T(AX + B)$$