IMA 206

Focus Stacking

Prepared by:

Renata Porciuncula Baptista
Roland AKIKI

Supervisor:

Yann Gousseau

2017-2018

Summary

A. Introduction

Multifocus Image fusion is the process of combining information of two or more images of a scene with the aim of getting a superior image with uniform focus and sharp content.

When one scene contains objects at different distances from the optical lens, the camera can be focused on each object one after the other, creating a set of pictures. Then, using an image fusion technique, an image with better focus across all area can be generated.

First, we consider that the images are aligned. The biggest challenge is therefore to estimate the sharpness of a pixel. This is done using various mathematical measures which are going to be tested and compared.

Later we deal with images which are not aligned. Thus, the challenge is to be able to align the images before applying the multifocus algorithm. This project was coded in Matlab.

B. Aligned images

1. Problem formulation:

We start by developing an algorithm on aligned images. The problem is therefore formulated as : Given a set of 2-D images $I_1(r,c), I_2(r,c), \ldots, I_N(r,c)$, which have been acquired using different imaging setting and aligned well, the goal of multi-focus image fusion is to integrate the information content of the individual images into a single fused image f(r,c):

f(r,c)= $\dfrac{\sum_{n=1}^{N} w_n(r,c) I_n(r,c)}{\sum_{n=1}^{N} w_n(r,c)}$ where $w_n(r,c)$ is the weight to the pixel at position (r,c) in the n-th image. This weight is no other than a sharpness criterion for pixel (r,c).

2. Sharpness criterion:

The existing sharpness criterions listed in the article [1] were tested. Below is an extract of the article with a reminder of the different mathematical formulas of each criterion

- Variance [12]. For an $M \times N$ block of the image, its variance is defined as [12]

$$S_{VAR} = \frac{1}{M \times N} \sum_r \sum_c (I(r,c) - \mu)^2, \tag{3}$$

where μ is the mean intensity value of the image block and it is defined as $\mu = \frac{1}{M \times N} \sum_r \sum_c (I(r,c))$.

- Energy of image gradient [12]. For an $M \times N$ block of the image, it is measured as [12]

$$S_{EG} = \sum_r \sum_c \left(I_r^2 + I_c^2\right), \tag{4}$$

where I_r and I_c represent image gradients at the row and column directions, respectively. They are usually defined as $I_r = I(r+1,c) - I(r,c)$ and $I_c = I(r,c+1) - I(r,c)$.

- Tenenbaum [12]. For an $M \times N$ block of the image, it is measured as [12]

$$S_{TNG} = \sum_r \sum_c (\nabla I(r,c))^2, \tag{5}$$

where $\nabla I(r,c) = \sqrt{I_r^2 + I_c^2}$, in which I_r and I_c are gradients (obtained using Sobel operators) along the row and column directions, respectively.

- Energy of Laplacian [12]. For an $M \times N$ block of the image, it is measured as [12]

$$S_{EL} = \sum_r \sum_c \left(\nabla^2 I(r,c)\right)^2, \tag{6}$$

where $\nabla^2 I(r,c)$ represents image gradient obtained by Laplacian operator $[-1,-4,-1;-4,20,-4;-1,-4,-1]$.

- Sum-modified-Laplacian [12]. It differs from the usual Laplacian operator in that the absolute values of the partial second derivatives are summed instead of their actual values. That is, it can be mathematically expressed as [12]

$$S_{SML} = \sum_r \sum_c \left(\nabla^2 I(r,c)\right)^2, \tag{7}$$

where $\nabla^2 I(r,c) = |2I(r,c) - I(r+1,c) - I(r-1,c)| + |2I(r,c) - I(r,c+1) - I(r,c-1)|$.

- Frequency selective weighted median filter [14]. It measures the sharpness of the image as

$$S_{FSWM} = \sum_r \sum_c \left(I_r^2 + I_c^2\right), \tag{8}$$

where

$$\begin{aligned} I_r = {} & \operatorname{med}\{I(r-1,c), I(r,c), I(r+1,c)\} - \tfrac{1}{2}\operatorname{med}\{I(r-3,c), I(r-2,c), I(r-1,c)\} \\ & - \tfrac{1}{2}\operatorname{med}\{I(r+1,c), I(r+2,c), I(r+3,c)\}, \\ I_c = {} & \operatorname{med}\{I(r,c-1), I(r,c), I(r,c+1)\} - \tfrac{1}{2}\operatorname{med}\{I(r,c-3), I(r,c-2), I(r,c-1)\} \\ & - \tfrac{1}{2}\operatorname{med}\{I(r,c+1), I(r,c+2), I(r,c+3)\}. \end{aligned}$$

We also implemented the method explained in article[1]. First construct the matrix C:

$$C = \begin{pmatrix} \sum_W I_r^2(r,c) & \sum_W I_r(r,c) I_c(r,c) \\ \sum_W I_r(r,c) I_c(r,c) & \sum_W I_c^2(r,c) \end{pmatrix},$$

Then obtain the eigenvalues λ_1 and λ_2 with the associated eigenvectors v1 and v2.

Compute A(r,c)= $\lambda_1 - \lambda_2$ and

where θ(r, c) is the phase information at the position (r, c) determined by the principle eigenvector v1 associated with the largest eigenvalue λ_1 , $\bar{\theta}$ is the average of phases of the neighboring positions.

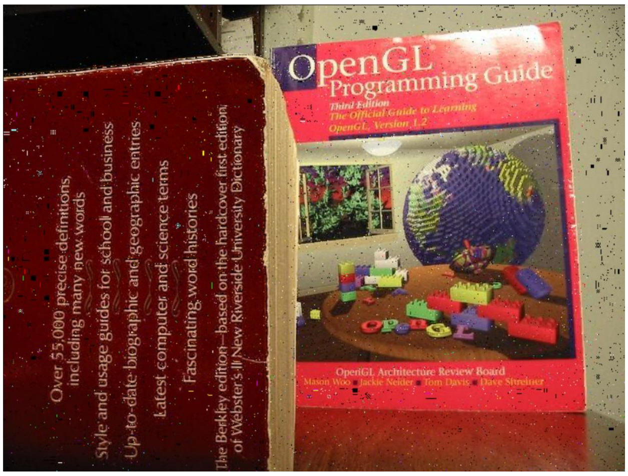
$$S_{BSC} = A^{\alpha}(r,c)P^{\beta}(r,c).$$

S_{BSC} is the sharpness criterion. We fixed α to 1 and β to 0.5 as recommended by the article.

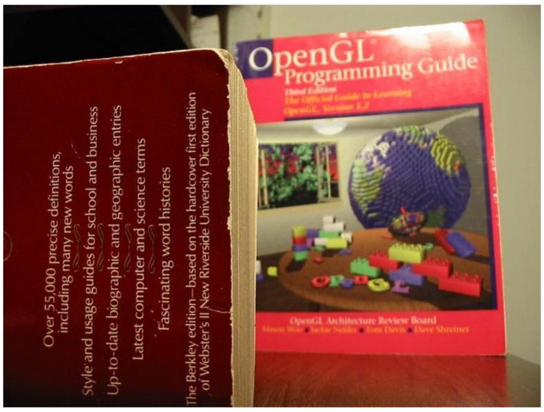
3. Results:

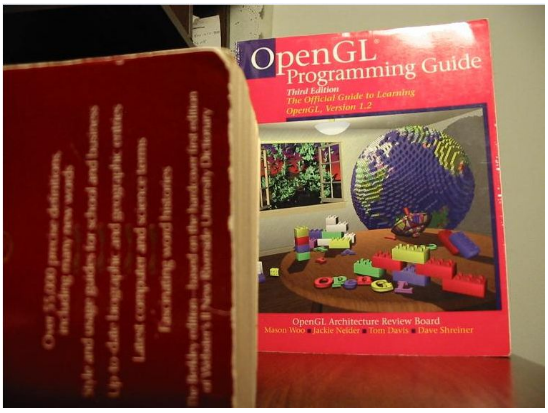
The results with the different measures on the different images are all satisfying. Some are slightly better then others, and some take more time to compute. Since the way used to evaluate the results is purely visual, we can't give an accurate estimation of the overall performance. For that we would need to compare to the ground truth. However, what we can do is compare locally, and search for defects in the output images.

First of all, it is worthy to mention that we encountered early on a bug in our implementation were we would get output images filled with NaN values. This was due to the fact that for some pixels, mostly in constant/uniform parts of the image, the sharpness measure is equal to 0 in all images. Thus when we normalize this, we divide by the sum of weights which is 0. This results in NaN weights and thus NaN values. We solve this by replacing the NaN normalized weights by 1/NumberOfImages. A similar problem was found with weight too small, when normalize, we found Inf. The solution was similar, replace this weight by zero, falling upon the later problem already solved.

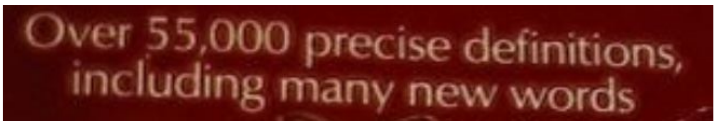


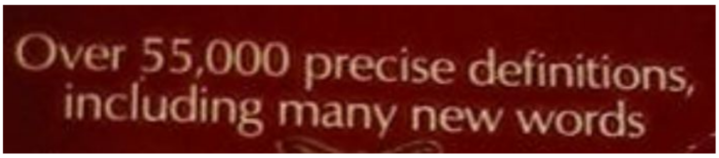
Now let us examine the effect of applying the different methods on edges. This effect is clear when it comes to text. This is why we choose the example of the book image to illustrate this idea, where we start off with the 2 images below to do the fusion.





After applying the fusion with various algorithms, we examine the text on the book. Below is the first sentence obtained with : Sobel gradient (top) , FSWM filter (bottom)





We notice that along the border of the letters, for the top image, we have a certain blend between the brown color of the background and the white color of the letters.

The same goes for all other sharpness criterions. However, this phenomenon is less apparent in the bottom image.

One way to interpret that is that the weight of a pixel is computed as a result of mathematical operations involving the neighboring pixels (in a certain window). This means that the weights of pixels on the letter borders are calculating using values of the pixels belonging to the background(brown) and to the letters(white). Whereas for the FSWM filter, the weight is computed using median filtering which has been proven to behave well along the contours. This is why we might be getting a slightly better result.

Another issue is the window size. It is often needed to compute the sharpness measure by using the pixel values inside a neighbourhood window. The size of this window is not known and it is a parameter that we should fix. However, it is not difficult to imagine that if there were an optimal window size, this value would depend on the scale of the image. For example, a 3x4 window on a certain image would be the same as a 6x8 window on the zoomed (x2) image. It is also not difficult to imagine why this window size would depend on the content of the image. For uniform/constant regions, it is better to have a large window (more observations under the hypothesis of spatial redundancy will yield a more accurate, less noisy result). For contours, it is better to have a small window. Therefore, an approach with an adaptive window size could be adopted. Testing this theory by relying only on the visual perception was proven to be difficult, which means that the window size if finally not so important to our perception and a 3x3 or 5x5 window would suffice. Choosing a small window size will also yield a faster computation time.

Now let us take a look on the computation time while keeping in mind that the values obtained depend heavily on our implementation. They also depend on the machine used to run the code.

	Laplacian	variance	gradient	SML	FSWM	article
book(sec)	0.2	0.21	0.17	0.14	0.23	4.65
Bug (sec)	2.47	2.79	3.6	2.64	3.99	95.87

The first conclusion is an obvious one: the computation time increases with respect to the number of images : book/2 input images bug/13 input images. Another conclusion is obtained by inspecting the chart. It is obvious that SML is a fast method for both images, and that the method of the article is a really slow one for both images.

C. Image registration

1. Problem formulation:

The second group of images that we would like to focus is the one of images that are not aligned since this is a problem because we can't suppose anymore that the same pixel position in the differents images correspond to the same object. Therefore, to tackle this problem, we need to align the images. This procedure is divided in three steps: first, find the correspondences between the images. Second, find the homography that represents this transform. Finally, apply this transform in the other images.
We try to adapt the code used to construct panoramas in the tp [2] to our application.
First, we are going to choose a reference image to which we are going to align all others. We choose the image with the middle index.
Then we take the other images one by one and apply this for image *I* :

- Find the correspondences between *I* and the reference using SIFT algorithm
- Find the homography that best describes these correspondences using RANSAC
- Compute the bounding box of the transform of image *I*

In the end, we take as a bounding box the area where all bounding boxes overlap, and this is the biggest difference with the panorama algorithm(where we take the union and not intersection of the areas of bounding boxes) . The reason we do this is that for the multi -focus algorithm, we are only interested in the region where we have pixels for all the images(the region where all warped images overlap).

Finally, since we have the bounding box, we can warp all images to the reference image within the bounding box.

2. Results:

For using the algorithm described above, we make a supposition about the images and the manner that they were captured. For instance, for the model of homography to be correct, we assumed that the only variations between the scenes is either a rotation around the optical center of the camera or that the image is planar. When those criterias are not satisfied, we may find some artifacts in the aligned images.

First, let's observe some partial results (after the alignment operation):

Before the alignment:

After:

As explained above, we used the center image as reference. Looking at the image in the right, we can see that the transformation applied rotated the image, but not enough to perfect alignment. This effect is a result of the the blur in the images which results in the SIFT algorithm finding pairs without the precision needed, even though SIFT is supposed to be invariant to blur(Gaussian pyramid) . A way to reduce this effect is by changing the threshold in RANSAC algorithm, that is, choosing more or less pairs to determine the homography.

The problem is that there's no easy way to decide this threshold, once that due to the blurring compare the image alignment is not obvious. Consequence of this problem is some artifacts in the final image, as we'll see below.

Another problem is when we have moving objects, let's analyze the scene below.

Before the alignment:

After the alignment:

Final result:

In this case, we can see the ghost effect in the pedestrian next to street pole, that's because he's not in the same position in the differents images. The bottle also suffers with the problem of the pair of SIFT not being in the same place, which gives this effect.
The problem of moving objects is a recurrent one in multi-image algorithms. The solution could be identical to the one adopted in HDR: theory of optical flow, patch methods, motion measure….

D. Conclusion

At the beginning of the project, our goal was to implement an algorithm able to focus images aligned and not aligned, and that was accomplished. First, we implemented an algorithm with six different sharpness criteria. The quality of results depends on the tuning parameters, the images and if the images fulfil or not the mathematical assumptions behind the techniques applied.

In a second moment, we tackle the problem of images which are not aligned. To do so, we adapt an algorithm to build panoramas given in a laboratory session, to align the images. The main difference between the panorama and the focus stacking algorithm is that we only keep the intersection of warped images. This algorithm used the SIFT to find the matches between the images, and the Ransac to find the homography that led to this transformation.

A problem with that approach is that not all images fulfil the criteria (image planar, rotation around the center optical of the camera, no movements). The results can be improved by taking the movement into account and using the theory of optical flow, patch methods, motion measure

Bibliographical references

1. Multi-focus image fusion using a bilateral gradient-based sharpness criterion, Jing Tian a,∗, Li Chen a, Lihong Ma b, Weiyu Yu c
2. Subject website: notes and lab session, accessed in 30/04/2018.