CS 534
COMPUTER VISION
ASSIGNMENT 2
SUBMITTED BY: SANCHIT THAKUR (st976)

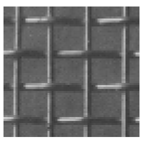
PART 1: EFROS-LEUNG ALGORITHM:

TEXTURE SYNTHESIS:

The Efros-Leung paper and pseudocode provided was studied and a code was written to perform texture synthesis on the 5 input images provided. For each input image, 3 different window sizes of 5,9 and 11 were applied, and the output obtained is shown below:

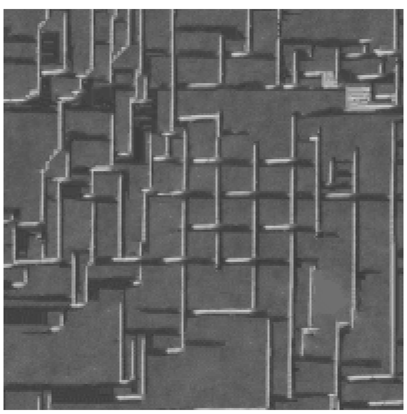
T1.gif:

Input image:

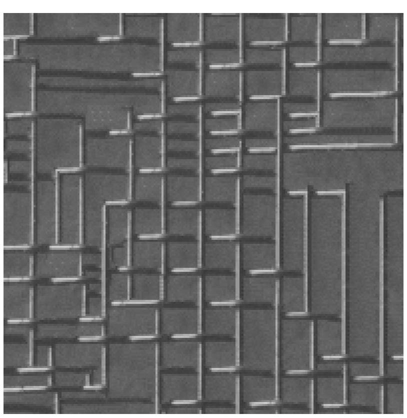


Output image:

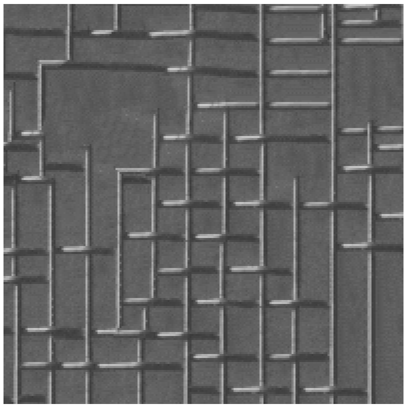
- Window size = 5



- Window size = 9

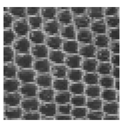


- Window size = 11



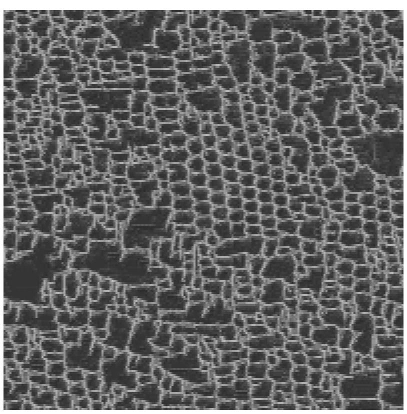
T2.gif:

Input image:

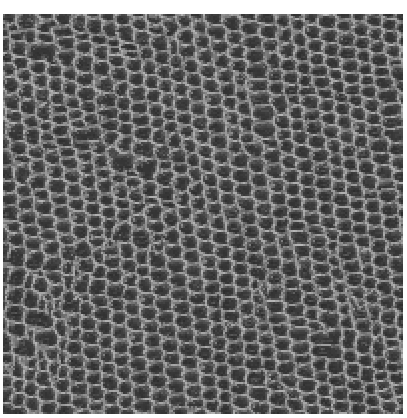


Output image:

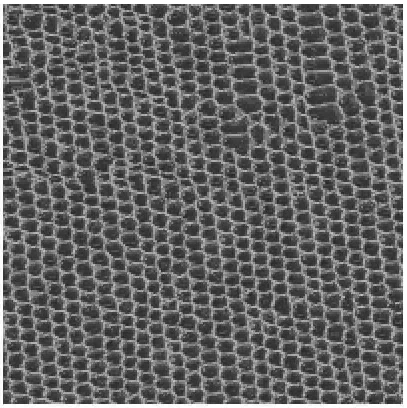
- Window size = 5



- Window size = 9

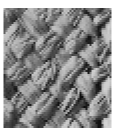


- Window size = 11



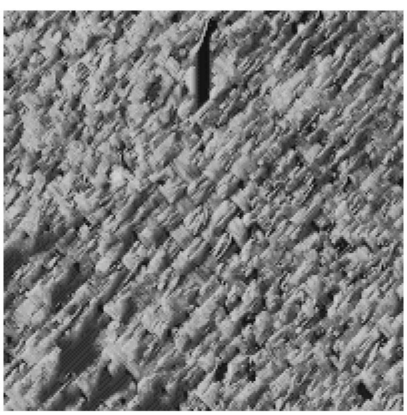
<u>T3.gif:</u>

Input image:

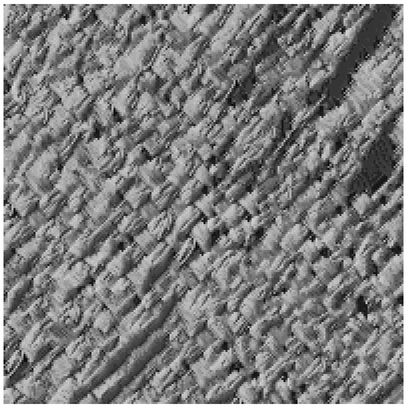


Output image:

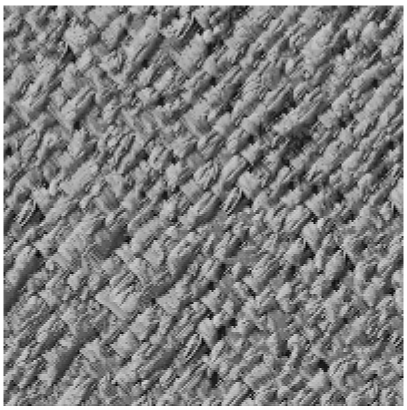
- Window size = 5



- Window size = 9

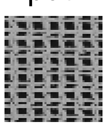


- Window size = 11



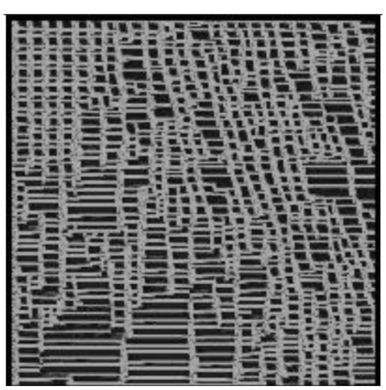
T4.gif:

Input image:

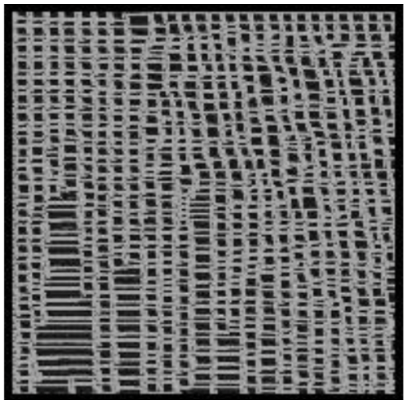


Output image:

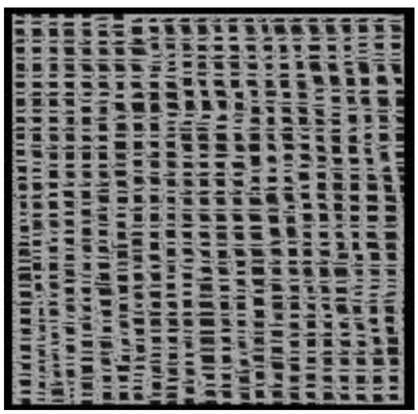
- Window size = 5



- Window size = 9

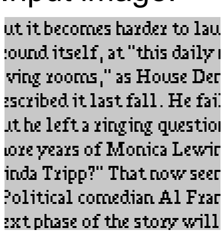


- Window size = 11



T5.gif:

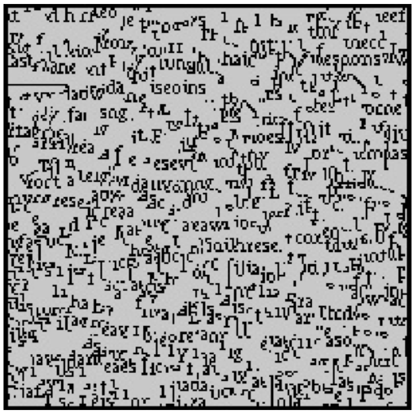
Input image:



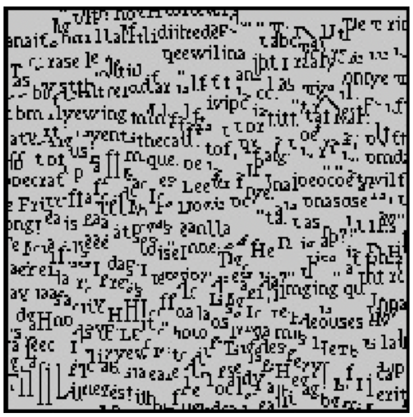
ut it becomes harder to lau
:ound itself, at "this daily r
ving rooms," as House Der
escribed it last fall. He fai
ut he left a ringing questio
ore years of Monica Lewin
inda Tripp?" That now seen
Political comedian Al Fran
ext phase of the story will

Output image:

- Window size = 5



- Window size = 9



- Window size = 11

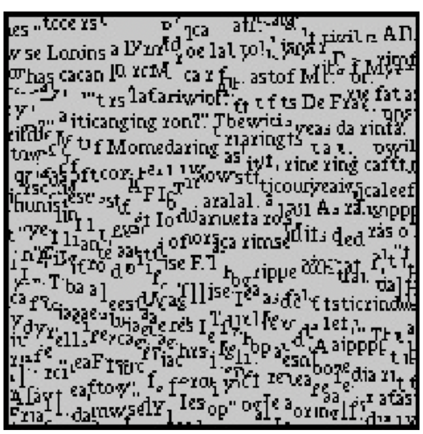


IMAGE INPAINTING:

The implementation used to carry out the texture synthesis described above was modified to now also perform image inpainting. Here, the 2 test input images contained gaps or black regions in them, and the algorithm fills in these regions. Again, window sizes of 5,9 and 11 were utilized. The outputs are shown below:

Test_im1.bmp:

Input image:

Output image:

- Window size = 5

- Window size = 9

- Window size = 11

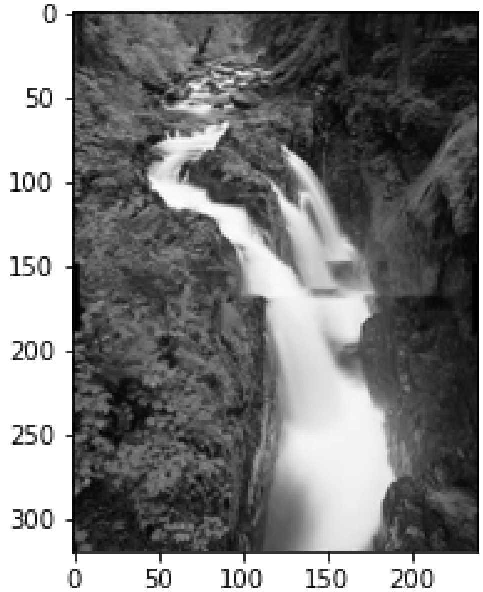
Test_im2.bmp:

Input image:

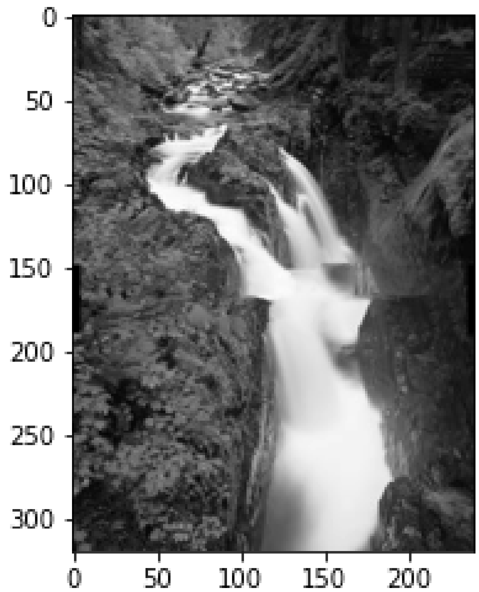
Output image:

- Window size = 5

- Window size = 9



- Window size = 11



As we can see from the above results, the window size of 11 gives the best output for the given task and input image.

OBJECT REMOVAL:

In this task, the same algorithm was used for the purpose of object removal. The input image was given and 3 objects were to be removed individually: man, sign and sunlight. The results for this task are shown below:

Test_im1.bmp:

Input image:

Output image:

- Man removed

- Sign and pole removed

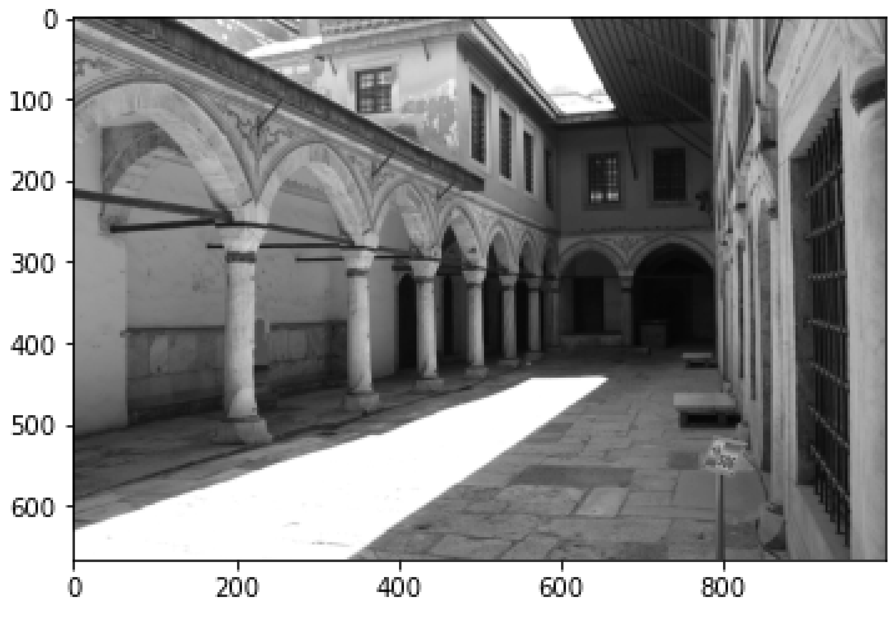
- Oversaturated area removed

PART 2: CRIMINISI ALGORITHM:

For the Criminisi algorithm, the paper provided was studied and an implementation of the algorithm was written. The aim of this algorithm is to again perform object removal. The same input image was provided and the same 3 objects were to be removed. For the sunlight removal, the algorithm took an exceptionally long time to execute, hence a portion of the object was masked and removed.

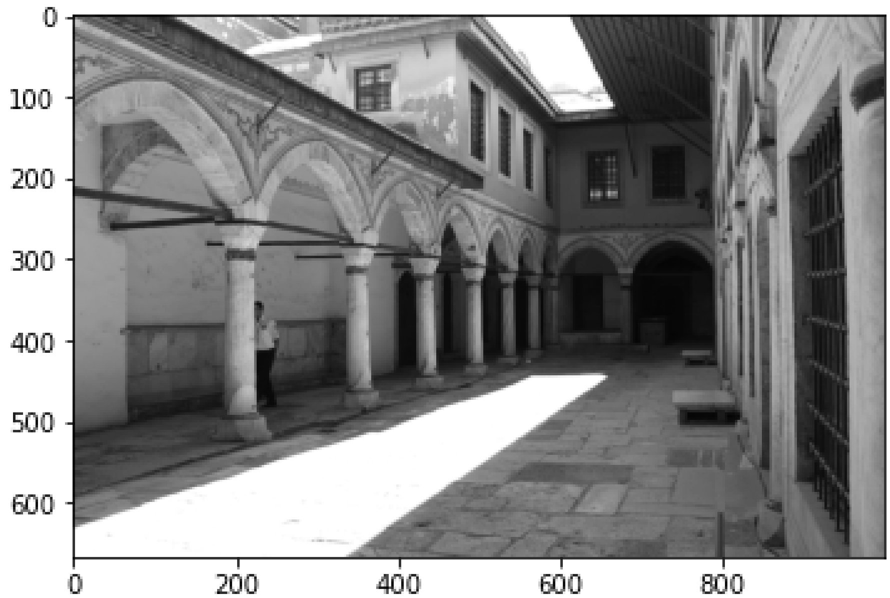
- Man removed

  Mask:



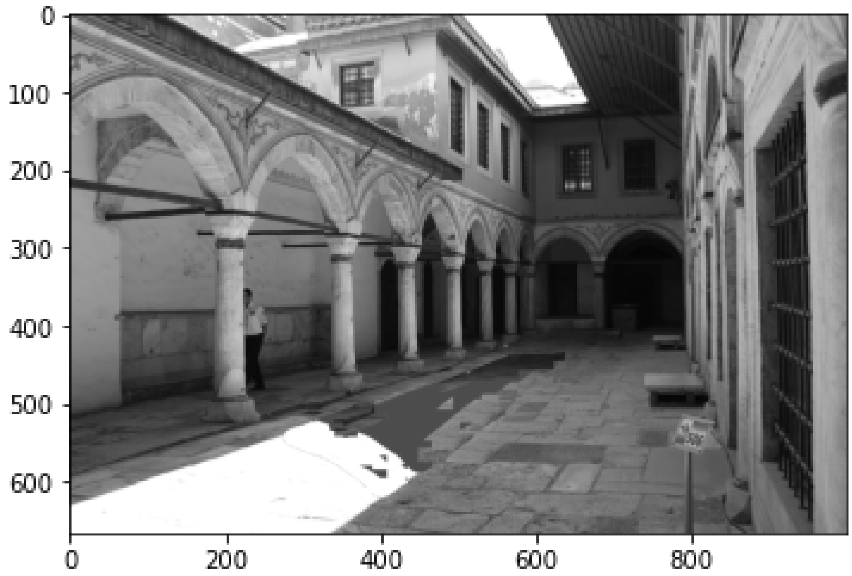
- Sign and pole removed

Mask:



- Oversaturated area removed

  Mask:



As we can see, the Criminisi algorithm runs much faster than the Efros Leung approach and also produces better results, therefore it is more efficient and thus more preferable.

PART 3: LAMA ALGORITHM:

For this last part, the paper and implementation of the Lama algorithm was provided beforehand. The algorithm was run for the same input image, and the task was to remove the same 3 objects as above. The results obtained are shown below:

- Man removed

  Mask:

- Sign and pole removed

  Mask:

- Oversaturated area removed

  Mask:

mask img img * mask

inpainting result

As we can see from the above results, it is evident that the Lama algorithm works much better than both the Criminisi and the Efros Leung algorithms. The following observations were made:

- The time taken to generate the output images in the case of Lama algorithm was substantially less compared to the other 2 approaches.
- Also, as was explained, the sunlight removal in Criminisi algorithm took multiple hours to generate the output, due to which the mask had to be reduced to obtain the output in an acceptable amount of time. However, there was no such issue in the Lama approach and the entire sunlight was removed, producing the output in mere seconds.

- Finally, as is evident, the quality of the output image after object removal is much better for the Lama approach compared to the other 2 approaches.

PART 4: FOURIER CONVOLUTION:

We know that convolutions are everywhere in data analysis. For decades, they've been used in signal and image processing. More recently, they became an important ingredient in modern neural networks.

Mathematically, convolutions are expressed as:

$$\text{(Discrete)} \quad (f * g)_i = \sum_{j=-\infty}^{\infty} f_{i-j}\, g_j \;=\; h_i$$

$$\text{(Continuous)} \quad (f * g)\,(x) = \int_{-\infty}^{\infty} f(x-z)g(z)\,dz \;=\; h(x)$$

Also, we know that the Fourier transform is a mathematical transform which decomposes functions depending on space or time into functions depending on spatial or temporal frequency. The Fourier transform of a function is a complex-valued function representing the complex sinusoids that constitute the original function.

In the context of object removal, when the input image is fed to the Fourier transform function, it assumes the image to be a space domain where the space is the grid of image pixels. Hence, the function will map the image from a spatial domain to a spatial frequency domain.

In mathematics, the convolution theorem states that under suitable conditions the Fourier transform of a convolution of two functions (or signals) is the pointwise product of their Fourier transforms. More generally, convolution in one domain (e.g., time domain) equals point-wise multiplication in the other domain (e.g., frequency domain). Mathematically, the Convolution Theorem can be stated as:

$$\mathcal{F}\left[f * g\right] = \mathcal{F}\left[f\right] \cdot \mathcal{F}\left[g\right]$$

where the continuous Fourier transform is (up to a normalization constant):

$$\mathcal{F}\left[f\left(x\right)\right] = \int_{-\infty}^{\infty} f\left(x\right) e^{ikx} dx \ = \ F\left(k\right)$$

Keeping this in mind, when Fourier transform is applied to our input image, it first transforms the image and the kernel from the spatial domain to the frequency domain and then applies convolution on the image and kernel in the frequency domain. Then, we use the inverse Fourier transform to find a mapping back to the spatial domain.

Hence, we see how the Fourier transform allows us to separate the function from their addition using its behavior of conversion between time domain and frequency domain. Because convolution corresponds to the Hadamard product in the Fourier domain, and given the efficiency of the Fourier transform, this method requires far fewer computational operations than convolution. Hence, it ends up being computationally less expensive. Therefore, the Fourier Convolution is the preferred approach in removing large objects.