Artistic Rendering of Digital Images

Team Members :

Jyoti Saroha (MT17019)
Vishal Gulia (MT17069)
Ronak Kumar (2015080)

Introduction

- Texture Transfer (Subset of Non Photorealistic Rendering)
- Generate Sketch like effects in normal images
- Appears like drawings or some artistic piece rather than raw camera images.

Literature / Past Works

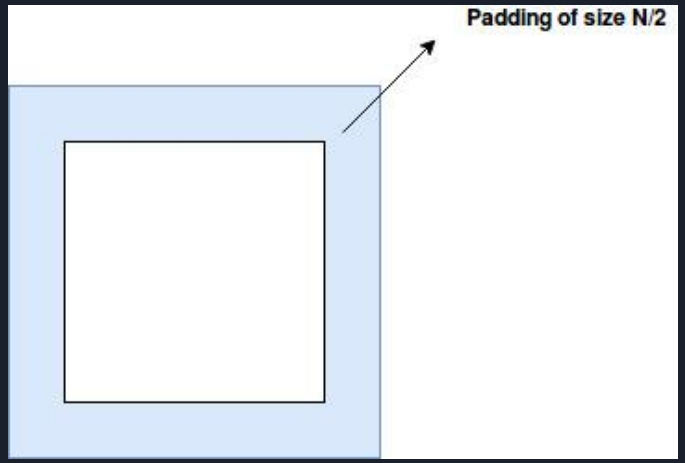
- Efros / Freeman et.al. were the first scholars implementing Texture Transfer to generate drawing kind effects in normal digital images.
- Used Image Quilting technique for stitching similar pixel clusters into single image
- Ashminkin proposed Fast Texture Transfer algorithm which uses coherent synthesis technique
- Shah et.al. used pixel and patch texture synthesis for texture transfer.

Dataset / Training

- Our code uses numerical computation using Fast Texture Transfer Algorithm.
- Hence, we don’t require any dataset.

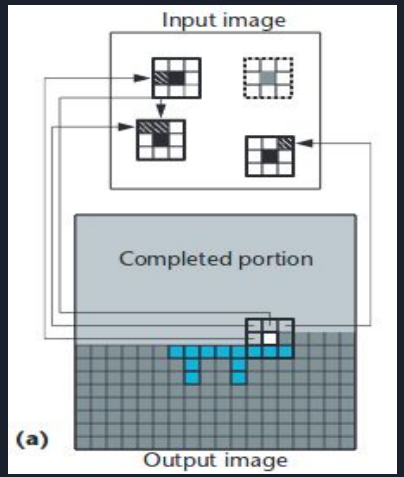
Methodology
Step 1:

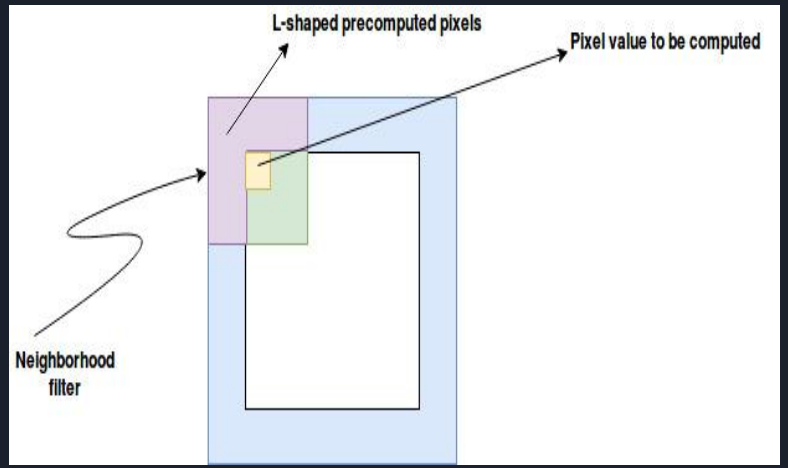
- Initializing Output Image
  - With size of target image
  - Padding of size N/2 for removing harsh edges



Methodology
Step 2:

- Generating Candidate Pixels
  - Consider an L-shaped neighbourhood
  - Compute pixel values in the shape with some probability





Methodology
Step 3:

- Calculating neighbourhood distance
  - Neighborhood distance = W*(Isource – Itarget) ^ 2 + (1/#pixels)^2*Dist(NLresultant, NLsource)
- Store specified resultant pixels into the resultant image

Evaluation

Source image (crayon.jpg)	
Target Image (korean.jpg)	

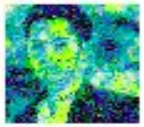
p = 0.05 m = 1 iterations = 1 n = neighbourhood size	 n=3	 n=5	 n=7	 n=9	 n=15
n = 5 w = 1 iterations = 1 p = probability	 p=0.05	 p=0.1	 p=0.2	 p=0.4	 p=0.5
n = 5 p = 0.2 iterations = 1 w = weight	 w=0.5	 w=1	 w=2	 w=3	 w=5
n = 5 p = 0.2 w = 1 iterations = iterations	 i=1	 i=2	 i=3	 i=4	 i=5

Table 1 - comparison of results by varying parameters

 Source Image (t1.jpg)	 Output of Fast Texture Transfer algorithm
 Target Image (girl3.jpg)	 Output of our algorithm (originality=0.25)

Table 2 - comparison of results from FTT and proposed algorithm

Artistic rendering is a subjective process. For this, this algorithm have five adjustable parameters. These can be adjusted for different results according to the user.

1) **neighborhood (neighborhood)**
   This parameter controls the pixels to be considered for the output image. It is observed from Table 1, 3x3 has too many sharp edges. While 15x15 has completely smoothed edges.

2) **probability (p)**
   This parameter determines the probability of adding the pixel in the final output. This affects the smoothness of the image. It is observed in Table 1, 0.2 gives better results. A low probability of 0.05 leads to sharp edges, while a high probability of 0.5 smooth out edges and loses low frequency components.

3) **weight (w)**
   In formula above, w signifies the importance to the average intensity. w=1 has got better results, as low weight is blurring the image. While higher weights are leading to losing low frequency components.

4) **iterations (iterations)**
   This depends largely on the perception of the viewer. It also depends on the source texture. More course textures converges on higher values of w.

5) **originality**
   This defines how much similarity the user want in the output image to the target image. A value of 0.25 seems to work well, as it conserves the properties of the target image in the output image.

Conclusion

In this project we have successfully implemented the texture transfer algorithm with some modifications. With this we were able to get some better results, which are discussed above. Though this is a simple algorithm, but it is sufficient to give considerably good artistic effects to the target image based on the source texture. More complex and computationally costly high algorithms can also be used.

This project can be extended by applying to other fields of image processing and domains. Obviously, improving the quality of the output result also.

References

[1] Efros, Alexei A., and William T. Freeman. "Image quilting for texture synthesis and transfer." Proceedings of the 28th annual conference on Computer graphics and interactive techniques. ACM, 2001.
[2] Hertzmann, Aaron, et al. "Image analogies." *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001.
[3] Ashikhmin, M., 2003. Fast texture transfer. IEEE Computer Graphics and Applications, (4), pp.38-43.
[4] Shah, Ankit, Parth Bhatt, and Kirit J. Modi. "Image enhancement techniques by texture synthesis." *Int. J. Emerg. Technol. Adv. Eng* 2 (2012): 97-101.