Artistic Rendering of Digital Images

Allison Card
Department of Electrical Engineering, Stanford University

Purpose

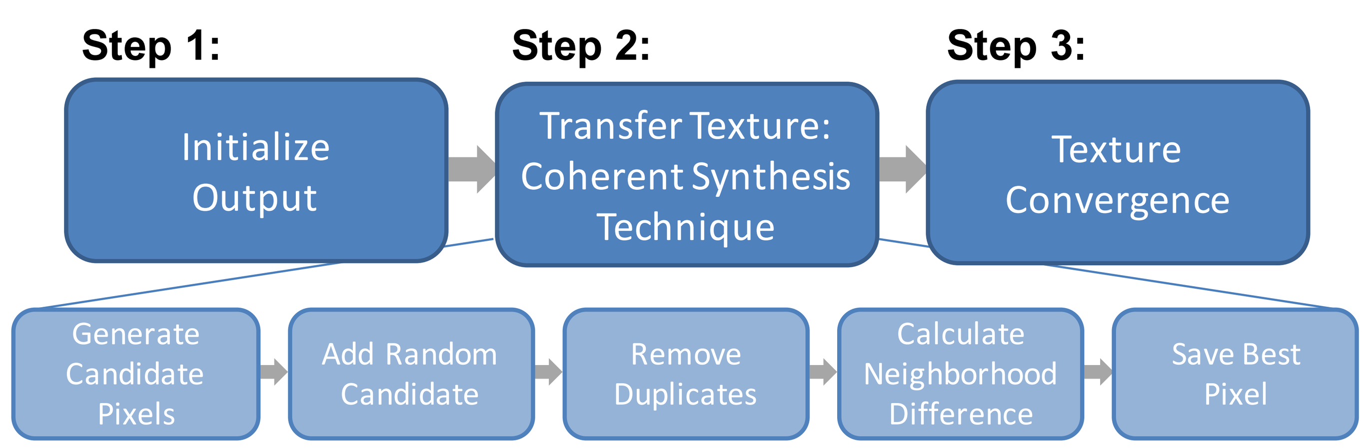
Artistic rendering is a subset of **non-photorealistic rendering** with the intention of using **artistic effects** in rendered images. This allows for a greater range of stylist presentations, dramatic effects, and ability to focus the viewer than in realistic rendering [1].

Texture transfer is a generalized method to render digital images in artistic styles. Texture transfer algorithms transfer the **source texture** to a **target image**. In the late 1990s and early 2000s, a number of texture transfer techniques were developed [2] [4] [5]. With some modifications, I chose to implemented Ashikhmin's fast texture transfer algorithm [2] and coherent synthesis technique [3].

Since **artistic style is subjective**, this algorithm has **four parameters** that allow a user to effect the style of the resulting image:

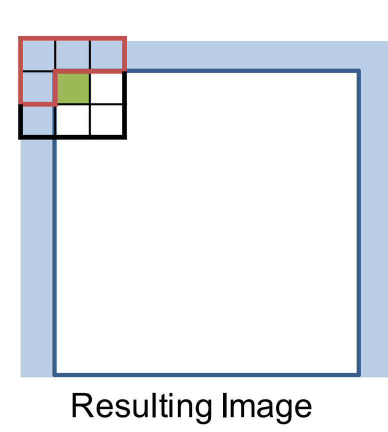
- Neighborhood size (n).
- Probability of adding a new pixel (p).
- Weight on average intensity difference between the source and target (w).
- Number of iterations (i).

Process Overview



Step 1: Initialize Output

An **L-shaped neighborhood** of **resulting pixels** is required for every **candidate pixel** in **Step 2**. To avoid harsh edges in the resulting image, a boarder of **half the neighborhood size** is created. Every pixel in the boarder is **randomly assigned** from the source image.



Step 2: Transfer Texture

For every pixel in the resulting image in **scanline order:**

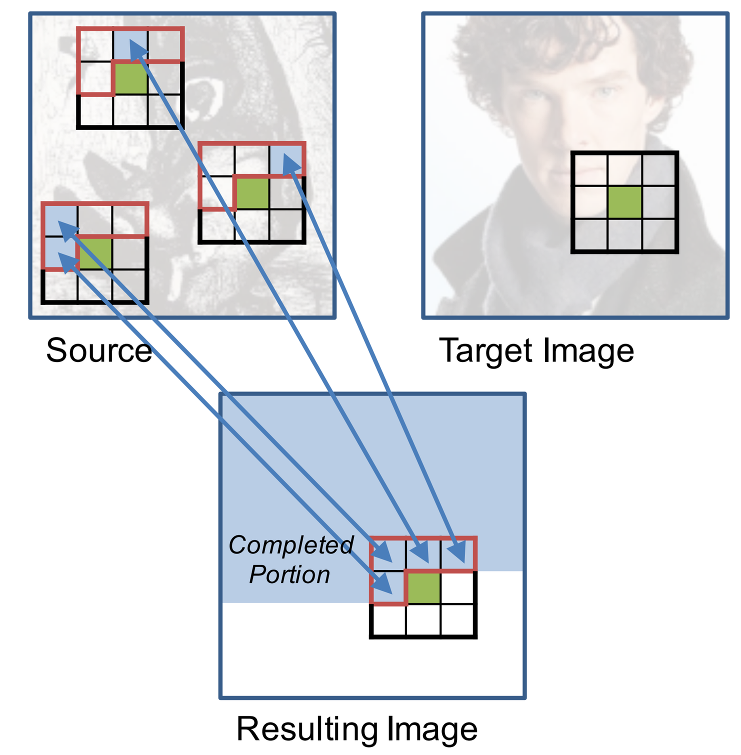
(a) Generate candidate pixels: For all pixels in the **L-shaped neighborhood** in the resulting image, find the neighboring **candidate pixel** in the source image.

(b) Add a random candidate: With probability *p* choose a random location in the source image as a **candidate pixel**.

(c) Remove duplicate candidates.

(d) Calculate neighborhood difference: For every **candidate pixel**, calculate the difference in average **neighborhood** intensity in the source and target image and the L2 distance between the **L-shaped neighborhoods** in the target and resulting images.

(e) Save candidate pixel: Find **candidate pixel** with the smallest neighborhood difference.



Reference [2], Image sources [1] [2].

$$Neighborhood\,Difference = w * (\overline{N_{source}} - \overline{N_{target}})^2 + (1/\#pixels_L)^2 * \mathrm{L}_2\,(N_{Lresult}, N_{Lsource})$$

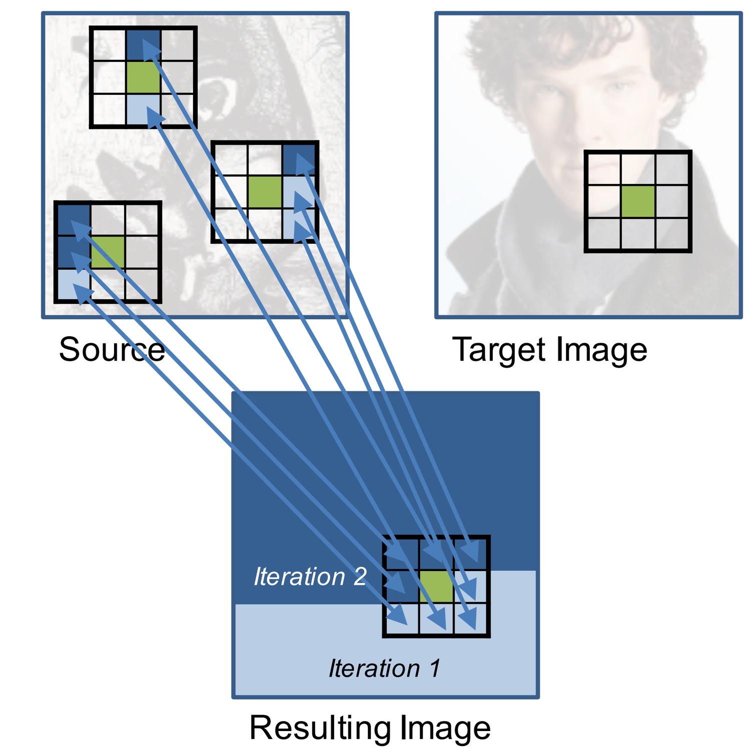
Step 3: Texture Convergence

One **iteration** may not be sufficient for the resulting image to converge (see (c) in experimental results). Multiple iterations can **reduce harsh edges**.

For iterations 2+: For every pixel in the resulting image in scanline order do the same as Step 2 except:

(a) Generate candidate pixels: For all pixels in the **upper L-shaped neighborhood** and **lower L-shaped neighborhood** in the resulting image, find the neighboring **candidate pixel** in the source image.

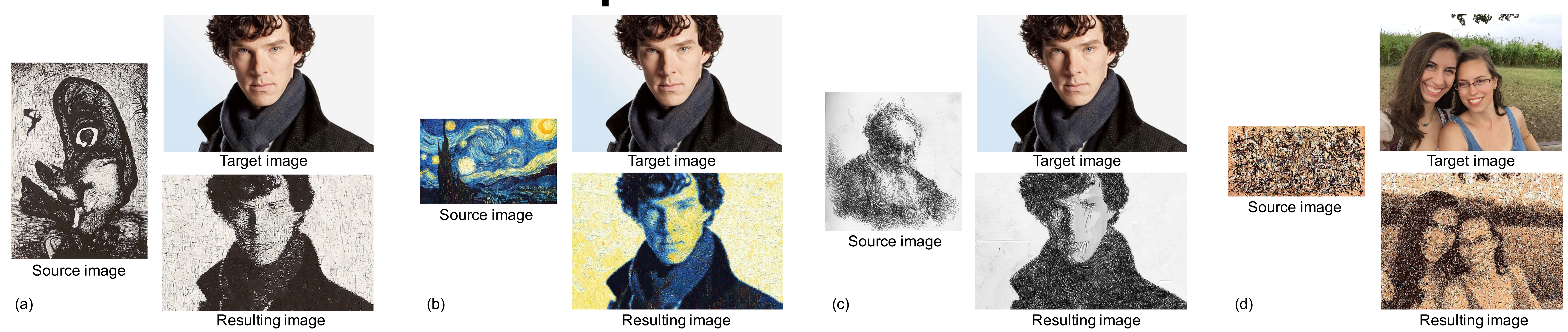
(d) Calculate neighborhood difference: For every **candidate pixel**, calculate the difference in average **neighborhood** intensity in the source and target image and the L2 distance between the **neighborhoods** in the target and resulting images.



Reference [2], Image sources [1] [2].

$$Neighborhood\,Difference = w * (\overline{N_{source}} - \overline{N_{target}})^2 + (1/\#pixels_N)^2 * \mathrm{L}_2\,(N_{result}, N_{source})$$

Experimental Results



(a) Parameters: n=5x5, p=0.2, w=1, i=1.
Image sources [1] [2].

- ➕ High frequency texture components (short lines).
- ➕ Similar foreground/background intensities and subject matter.
- ➕ Target lighting, composition.

(b) Parameters: n=5x5, p=0.2, w=1, i=5.
Image sources [2] [3].

- ➕ High frequency texture components (brush strokes).
- ➕ Target lighting, composition.
- ➖ Difference in foreground/background intensities and subject matter.

(c) Parameters: n=5x5, p=0.2, w=1, i=1.
Image sources [2] [4].

- ➕ Similar foreground/background intensities and subject matter.
- ➖ Smooth background components do not converge with high frequency ones.
- ➖ Long straight lines in source.

(d) Parameters: n=5x5, p=0.2, w=1, i=1.
Image sources [5] [6].

- ➕ High frequency texture components (paint splatters).
- ➖ Poor target lighting (strong directional light preferred).
- ➖ Busy background in target.

Image sources: [1] *Untitled* by Enrico Donati. [2] Benedict Cumberbatch. [3] *Starry Night* by Vincent van Gogh. [4] *After Rembrandt* by BrokenUmbrella. [5] *Autumn Rhythm no 30* by Jackson Pollock. [6] Unpublished.

References: [1] J. Romero and P. Machado, "Evolutionary search for the artistic rendering of photographs." [2] M. Ashikhmin, "Fast texture transfer." [3] M. Ashikhmin, "Synthesizing natural textures." [4] A. Hertzmann, C. Jacobs, N. Oliver, B. Curless, and D. Salesin, "Image analogies." [5] A. Efros and W. Freeman, "Image quilting for texture synthesis and transfer."