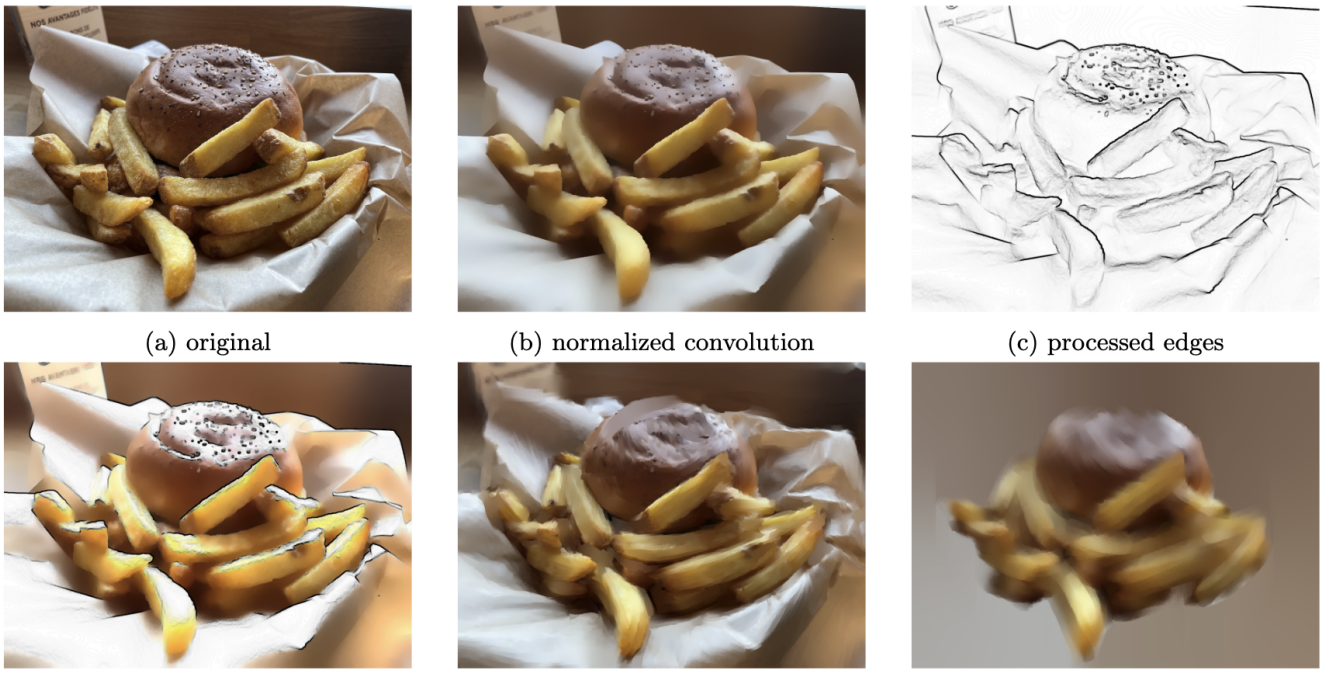
15463 Final Project - Image Stylization by Edge-Aware Processing

Gavin Zhu
feiyuz@andrew.cmu.edu



(a) original (b) normalized convolution (c) processed edges

(d) stylized (e) oil paint strokes (f) watercolor

Figure 1: Different stylistic transfer

ABSTRACT

Image stylization has been an open research question for more than a decade. Earlier work explores using edge-aware bilateral filtering for cartoonization, detail enhancements, etc. while more recent work utilizes data-driven methods for more complex image style-transfer. In this work, we revisit edge-aware image processing using domain transfer and demonstrate how it can be use for cartoonization and user-centric style transfer. Specifically, we recreated the cartoonization process in previous work and show how joint filtering can be used to introduce painting-like texture to a photo.

1 INTRODUCTION

Photos capture the world as it is, recording everything including the fine details and sometimes even noise. In many cases such as illustrations, a more simple stylistic image is more preferable. This requires a balance between keep the image realistic and also giving it an paint-like touch.

Earlier works in this direction has looked at removing the details of the image and amplifying the edges. This gives the image a cartoon-like feeling. More recent works leverage the power of machine learning and the internet-scale data to train statistical models that generates images of arbitrary style.

In this work we focus on more physically-sound approaches and explore how they can be used to generate stylistic images. The focus of this work is two-fold: 1) revisit domain-transfer edge-aware filtering, and use it to generate simple stylized images; 2) explore using bilateral filtering to give images a painting-like style.

Additionally, we also made the implementation of the two pipeline we developed publicly available, along with the auxiliary images we used to guide image style transfer.

2 RELATED WORK

2.1 Bilateral Filtering

Bilateral filtering works by averaging pixel values that are close in both spatial and intensity domains [7]. Doing so averages the image patches while preserving the edge information. Despite its wide applications, naive bilateral filtering is not linear nor shift-invariant, limiting its computational efficiency. Relatively recent approaches examines more efficient methods on edge-aware filtering using local Laplacian pyramids [4], domain-transfer approximation [1], and guided filters [2].

Final Project, Dec. 2023,
2023. https://github.com/zfy0314/image-stylization

In this work we re-implement the domain-transfer edge-aware filtering method for image stylization.

2.2 Style Transfer

Earlier works explored using bilateral filtering to perform style transfer such as cartoonization, detail enhancements, and simulated pencil drawings [1, 5]. More modern approaches demonstrated using data-driven methods including generative adversarial network [3] and stable diffusion [6]. Although the newer methods produce more astonishing results, their internal mechanisms and are not as clear as the filtering based methods.

In this work we aim to achieve style transfer with joint filtering, specifically giving images a paint-like feeling. Unlike previous cartoonization work that emphasizes on the existing edges of the image, in this work we introduce new edges to simulated the paint brushes. As our approach depends only on edge-aware filtering, it is simpler than data-driven methods, and the user can have more control of the process.

3 STYLIZATION

3.1 Procedure

We largely follow the procedures as described in [1] for the normalized convolution and the outline for image stylization.

Specifically, we first apply a normalized convolution to an image (I) such that it's visible edges are preserved while the more intricate ones are suppressed (I_{nc}). Then we extrat the major edges from the gradients of the filtered image by

(1) compute the per-pixel gradient magnitude of the filtered image
$$G = \left| \sqrt{\left(\frac{d}{dx} I_{\text{nc}}\right)^2 + \left(\frac{d}{dy} I_{\text{nc}}\right)^2} \right|$$
(2) scale the gradient by some constant to make the edges darker
(3) dilate the gradient image with a small box filter to make the edges thicker (more visible)
(4) apply Gaussian Blur to the dilated gradient image to soften the edges
(5) clip the values to appropriate range

Given the processed edges G', the final stylized image is computed by taking the per-pixel minimum between $\alpha \cdot I_{\text{nc}}$ and G', where α is a brightening constant to give more contrast between the original filtered image and added edges. For the sake of simplicity we take the minimum per RGB channel, but for better result one can convert the images to the XYZ color space and take the minimum on the luminance channel.

After going through the pipeline as described above, the edges are amplified by the dark lines and the area in between are filled with blurred, filtered image.

3.2 Results

Figure 2 shows the comparison between our stylization pipeline and the one from [1]. As the original paper didn't provide the details on how their edges are computed and integrated, only the final result is used for comparison.

We keep the parameters (σ_s, σ_r, and the number of iterations) the same as described in the original paper.

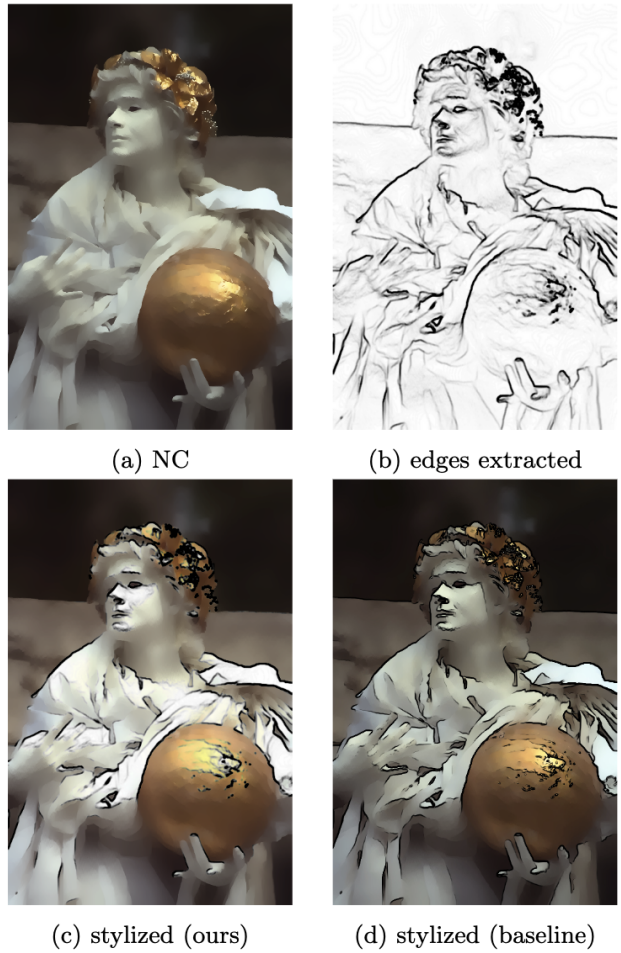


Figure 2: Stylization on the statue image

As our method brightens the base image, overall it gives a cleaner feel. The artifacts on the ball and the head ornaments are shared between both methods as the gradients in those areas are prominent. Our method shows fewer edges than the baseline (e.g., the baseline has a visible line on the face of the statue but not ours). Otherwise there is not much considerable difference between the two approaches.

More result is shown in Figure 1 (d) where the edges and NC filtered images are fused (Figure 1 (b) and (c)). Not every edge (e.g., the fries on the lower part of the image) is picked up because its curved shape leads to more gentle gradient changes.

The edges extracted using our pipeline also give a feeling of pencil-drawn line art as the boundaries are highlighted with thick black lines while sufficient details are illustrated with more intricate lines.

4 STYLE TRANSFER

Building on top of the normalized convolution as described in the previous section, in this section we explore how joint filtering can

Figure 3: Burger with vertical oil paint style transfer

Figure 4: Burger with image-aligned strokes and $\sigma_r = 0.3$

support style transfer. More formally, we want to give the realistic photo a paint-like feeling.

4.1 Procedure

We use the joint filtering and provide auxiliary stroke patterns as the joint image. The high-level idea is that the joint image will simulate the boundaries of each paint stroke. By filtering on the original image's color profile with edges of the joint image, it would look like the image is painted by different strokes.

More concretely, the joint image can either be an image-agnostic synthetic brush strokes (e.g., Figure 6 (a)), or customly made patterns that is specific to the image and the desire of the user (Figure 6 (b)). In the later case the strokes can be selective applied such that that only part of the image gets transferred (Figure 6 (c)).

In practice, the image-centric strokes are created by the built-in brushes in the Procreate software.

Figure 5: Burger with image-aligned strokes and $\sigma_r = 1.5$

4.2 Results

Figure 3 shows the result of transferring the burger image (Figure 1 (a)) to the vertical oil paint style (Figure 6 (a)). As the strokes are not aligned with the edges on the image, it feels like looking at the burger through a wet glass.

When the strokes are more aligned, the transferred image shows a more coherent style (e.g., Figure 1 (e) is transferred from Figure 6 (b)). This gives the image an oil painting-like touch, especially the imperfections such as the brush stroke on the top left of the burger. As the colors are blurred within each stroke boundaries, this simulates the consistent yet non-constant color of each brush stroke in real paintings. This is especially visible when looking at the fries and the texture of the brown paper on the upper right.

Figures 1 (f) and 6 (c) show that by changing the stroke type and region, we can put emphasis on the object of interest and entirely blur the background and give the object a more abstract feeling.

Figures 4 and 5 show the effect of varying σ_r. When σ_r is larger the strokes are more coarse and the image is more painting-like. Specifically, the sesame seeds on the burger are somewhat visible when σ_r is small but not at all when σ_r is large. Experiments shows that the difference between different σ_s is very subtle on the burger image.

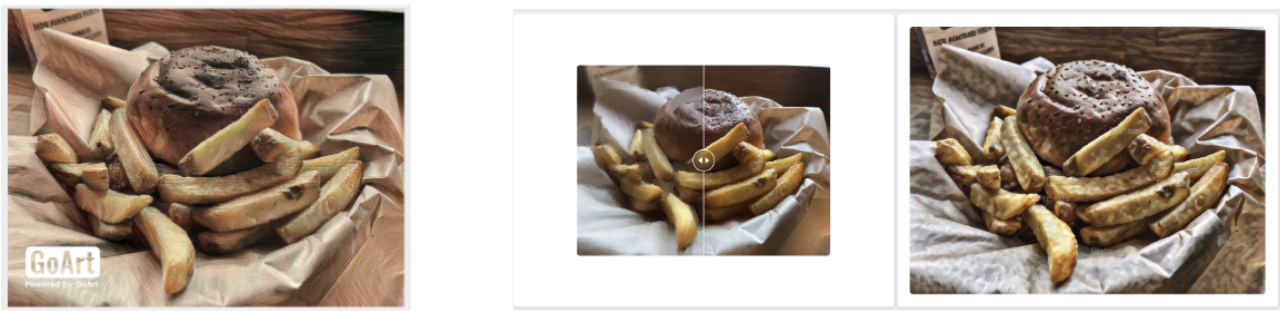
During experimenting with different brush types, we found that not all styles can be transferred. For example the spray paint brush or charcoal end up just averaging the image without having a visible texture. This is hypothesized to be their texture is too fine-grained unlike the lengthy lines that oil painting and watercolor have.

On comparison, Figure 7 shows the style transfer from two online platforms. They didn't disclose their exact algorithm but both of them claim to use data-driven machine learning methods. Figure 7 (a) is decently visually pleasing but only has uniform strokes across the peripheral of the image. The top of the burger is still too realistic. Figure 7 (b) failed to give it a stroke-like touch, but seems more like fusing the original image and the paint-stroke image more directly.

These comparison shows that our method has more flexibility (in terms of type of paint strokes, region of the transfer) and provides

(a) vertical oil paint (b) custom oil paint (c) custom watercolor

Figure 6: Different textures for style transfer



(a) goart.fotor.com (b) phosus.com/tools/image-style-transfer

Figure 7: Two online off-the-shelf image style transfer

paint-brush like feeling to the image with a more physically sound manner.

At the time of writing GPT4-vision model does not support style transfer, so we didn't experiment with using large foundation models to perform image style transfer.

5 CONCLUSION AND FUTURE WORK

In this work we re-implemented one of the the stylization processes from the previous work using domain-transfer edge-aware processing. Additionally, we explored using joint filtering to add texture to a given image by using joint filtering. Specifically we demonstrated that by using different types of stroke and having different placement, we can simulate different painting textures and emphasize on the foreground elements. We also briefly compared our technique with some off-the-shelf style transfer pipeline online.

A straightforward extension to the current style-transfer pipeline is to automatically generate the stroke patterns from the original gradients from the image instead of purely relying on the user. This can be potentially achieved by first computed the gradients of the original image, then estimate the contiguous segments using Hough lines transform, and finally placing the stroke patterns along the estimated line segments.

REFERENCES

[1] Eduardo SL Gastal and Manuel M Oliveira. 2011. Domain transform for edge-aware image and video processing. In *ACM SIGGRAPH 2011 papers*. 1–12.

[2] Kaiming He, Jian Sun, and Xiaoou Tang. 2012. Guided image filtering. *IEEE transactions on pattern analysis and machine intelligence* 35, 6 (2012), 1397–1409.

[3] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-Image Translation with Conditional Adversarial Networks. *CVPR* (2017).

[4] Sylvain Paris, Samuel W Hasinoff, and Jan Kautz. 2011. Local Laplacian filters: Edge-aware image processing with a Laplacian pyramid. *ACM Trans. Graph.* 30, 4 (2011), 68.

[5] Ramesh Raskar, Kar-Han Tan, Rogerio Feris, Jingyi Yu, and Matthew Turk. 2004. Non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging. *ACM transactions on graphics (TOG)* 23, 3 (2004), 679–688.

[6] Sloke Shrestha, Sundar Sripada V. S., and Asvin Venkataramanan. 2023. Style Transfer to Calvin and Hobbes comics using Stable Diffusion. arXiv:2312.03993 [cs.CV]

[7] Carlo Tomasi and Roberto Manduchi. 1998. Bilateral filtering for gray and color images. In *Sixth international conference on computer vision (IEEE Cat. No. 98CH36271)*. IEEE, 839–846.