Cartoonize Me!

Ai-Linh Alten

University of California, Merced

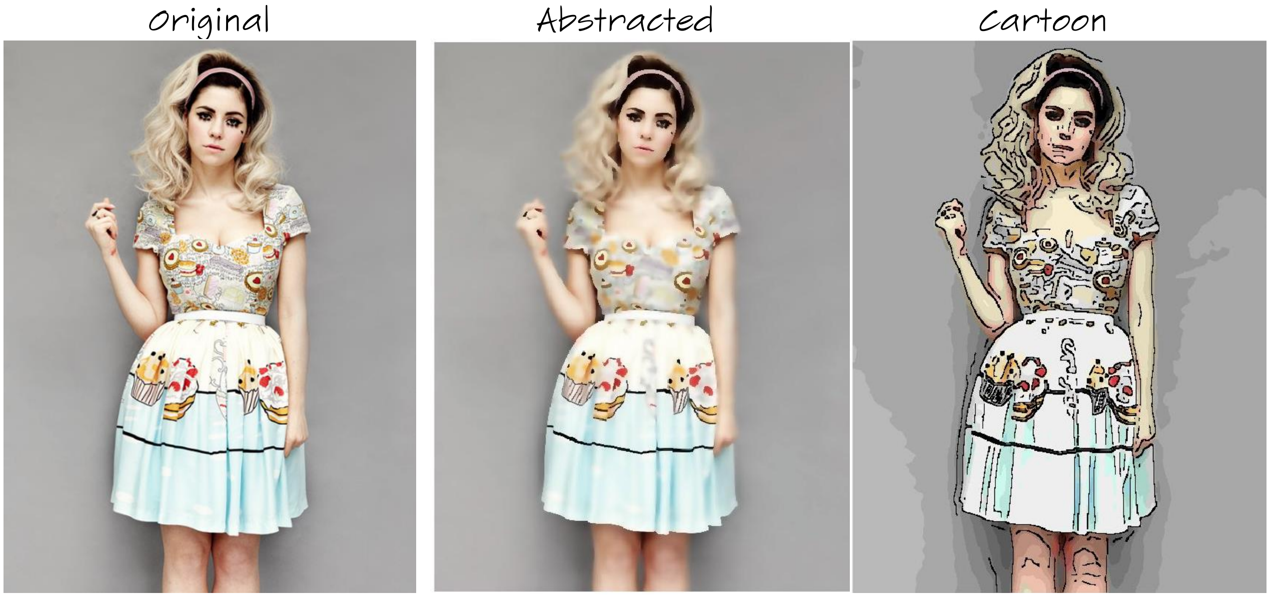

Figure 1: **Cartoon example.** *Original*: Marina and the Diamonds - Electra Heart. *Abstracted*: After several passes of bilateral filtering. *Cartoon*: Lab channels quantized to 16 bins. Overlayed with maximized Canny edge detector.

Abstract

Cartoonize Me! is a cartoonization filter consisting of three main steps. The first step to the cartoonization filter is a non-linear bilateral filter that is based on Gaussian distribution but consists of a weight that changes based on the displacement between image intensities and distance. The bilateral filter preserves edges but still reduces noise. The bilateral filter is done in the CIE-Lab space to emphasize contrast to make the image more visually salient. The second step to the cartoonization filter is to apply some sort of edge detector to enhance edges and make the image look like a drawing. The third step is a quantization effect to give the image more of a paint-like look.

1 Introduction

Although *Cartoonize Me!* seems like a straightforward filter, local contrast in an image can heavily affect the estimation of visual salience the bilateral filter computes. For example, images with very low contrast are more likely to abstract way too much and lose detail. A higher contrast image with a lot of detail would be further emphasized.

In this paper, I will explore the effects of the bilateral filter by making observations on its parameters to get the most cartoonized image. I will also explore the effects of the Canny edge detector on different parameters of the bilateral filter.

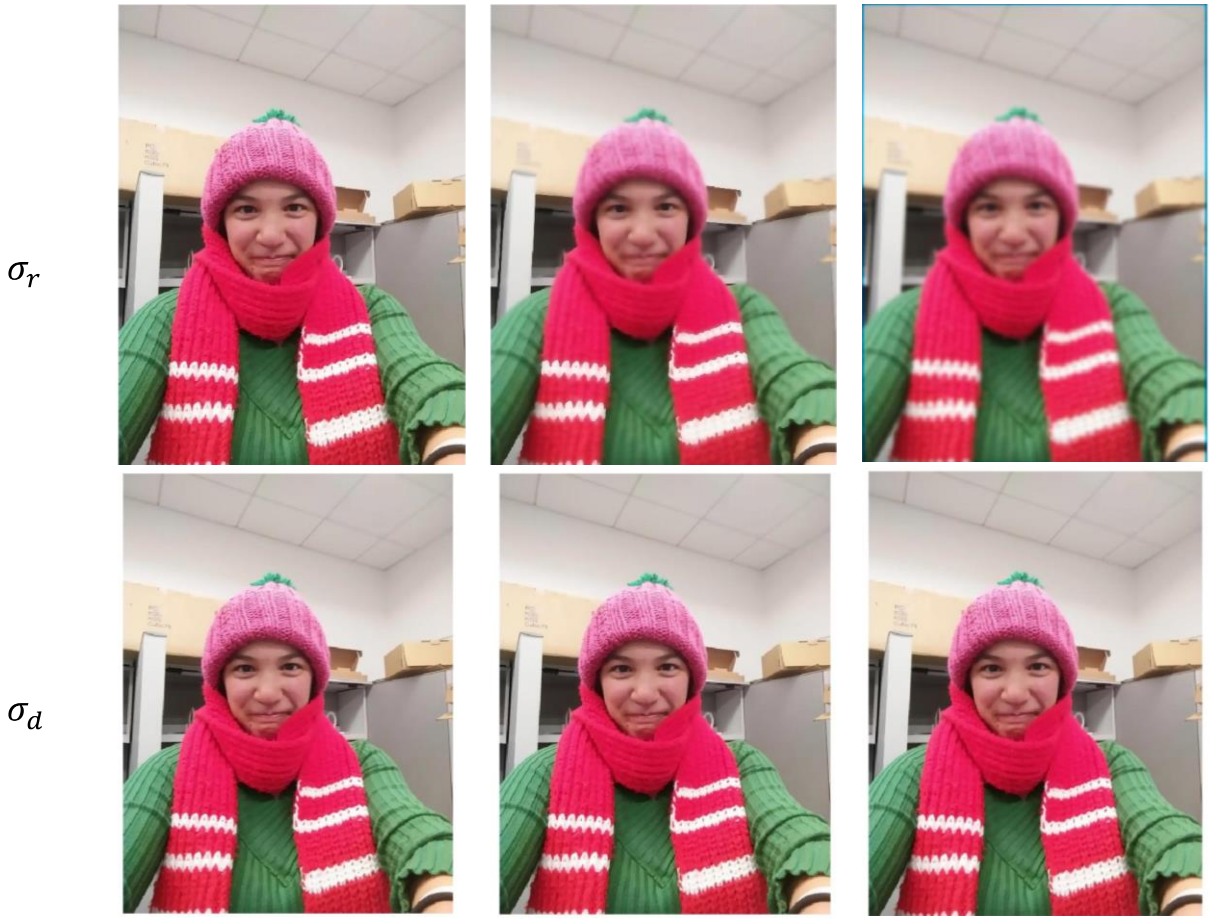

Figure 2: **Sigma samples.** Samples are tested with 3 passes of bilateral filter. *Range (top row)*: $\sigma_r = 0.02, 0.1, 0.8$ with σ_d = 3 from left to right on scale [0, 1]. Blur increases as it approaches 1. Edges become less preserved. *Domain (bottom row)*: σ_d = 3, 25, 1000 with $\sigma_r = 0.02$ from left to right. Changes in this parameter does not show significant change. Edges are all preserved.

2 Related Work

Just so you know that the cartoonization filter isn't anything new, I am supposed to let you know in this section that my research and findings is based off of someone else's research. This paper will closely relate to the research article on *Real-Time Video Abstraction* by Winnemöller, Olsen, and Gooch [1]. Their work focuses on performing the cartoonization filter in real-time which requires better optimized bilateral filter and edge detector to avoid disconnections between frames.

My work is based on their 3-step methodology on how to obtain a cartoon-like image. The first step being a bilateral filter, second step is a difference-of-Gaussian (DoG) edge detector, and an optional color quantization step on the abstracted image. However, their bilateral filter used an *extended nonlinear diffusion* filter which is an *anistropic diffusion* filter that helps amplify or subdue given contrast parts in the image. Mine is simplified to the bilateral filter given by the *Computer Vision: Algorithms and Applications* textbook by Szelski, Richard [2] and the MATLAB's Canny edge detector from the image processing toolbox.

In this report I try to explore the failure case Winnemöller, Olsen, and Gooch experienced when trying to abstract contrast-based images.

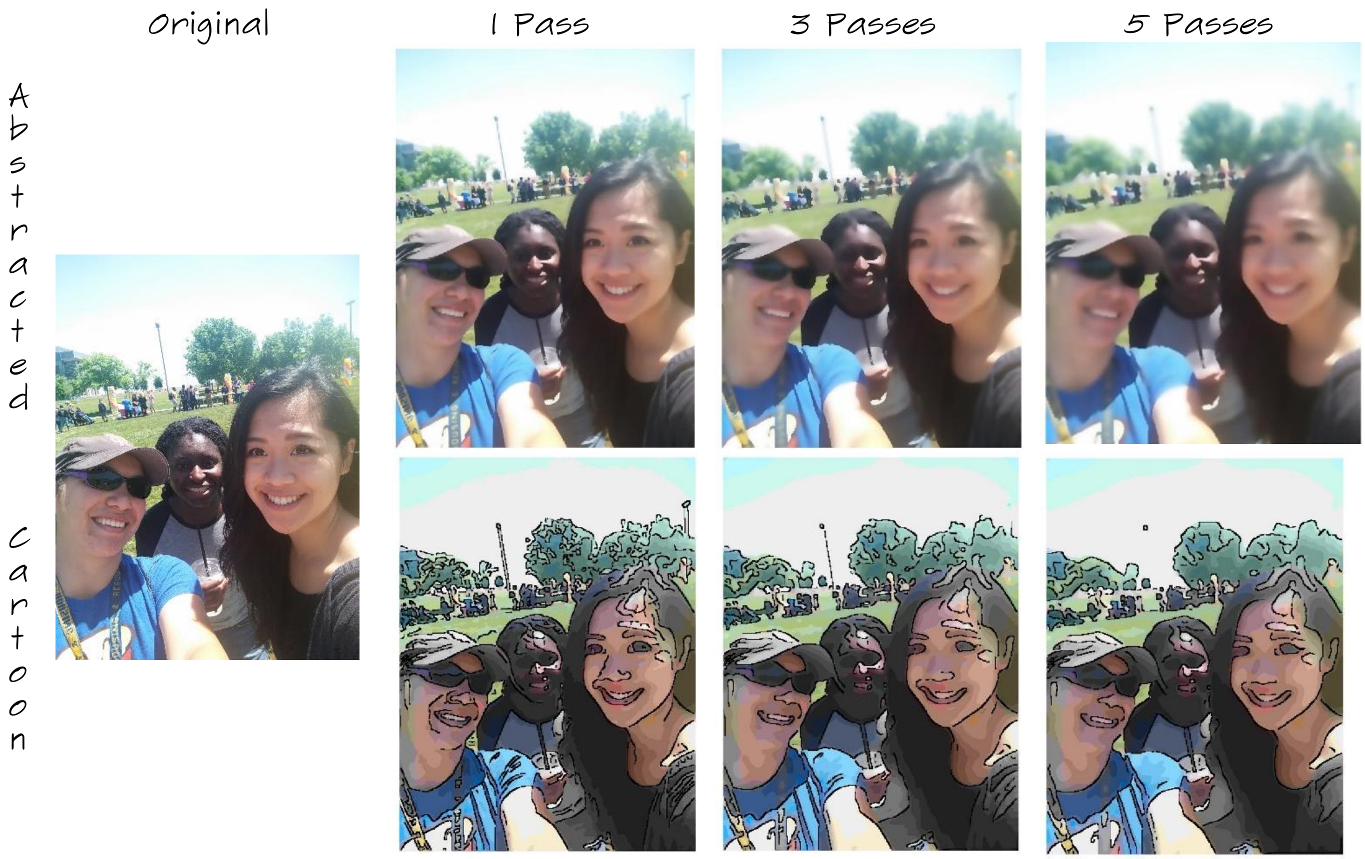

Figure 3: **Bilateral filtering & cartooning.** *Abstracted*: Bilateral filter undergoing 1, 3, and 5 passes. Blurring becomes apparent as number of passes increases. Edges appear to be preserved. *Cartoon*: images are overlayed with Canny edge detector for 1, 3, and 5 passes and images quantized to 16 bins. Less edges are detected as number of passes increase.

In their report, they experienced limitations dealing with images of super low contrast and very high contrast features [1].

3 Method

My goal is to find out the best parameters for abstracting the images with a bilateral filter on the CIE-Lab color space to get the desirable cartoon effect suggested by Winnemöller, Olsen, and Gooch [1]. The following steps were taken to find out the effects of the abstraction: (1) changing sigma constants on the range and domain kernel of the bilateral filter, (2) test how many passes of bilateral filter to obtain the best abstracted image, (3) find best parameters for high and low contrast images, (4) and observe effects of Canny edge detector from MATLAB IPT.

3.1 Bilateral Filter

In this step, I observed the effects of the bilateral filter given by Richard Szeliski [2]. Here I observe the effects of the following function:

$$g(i,j) = \frac{\sum_{k,l} f(k,l)w(i,j,k,l)}{\sum_{k,l} w(i,j,k,l)} \quad (1)$$

In this function, *(i,j)* is the pixel location and *(k,l)* are the neighboring pixels in the kernel. The output depends on the weighting function:

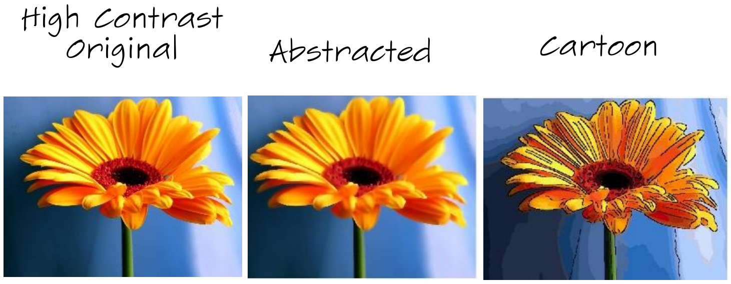

Figure 4: **High Contrast Example.** *High Contrast Original*: a cute sunflower. *Abstracted*: 3 passes of the bilateral filter. Not much difference between the first except disc florets were abstracted away. *Cartoon*: still manages to look like art.

$$w(i,j,k,l) = \exp\left(-\frac{(i-k)^2+(j-l)^2}{2\sigma_d^2} - \frac{\left\|f(i,j)-f(k,l)\right\|^2}{2\sigma_r^2}\right) \quad (2)$$

I tested the effects of the weighting function by changing the domain kernel's (Gaussian) sigma and range kernel's (intensity difference) sigma. σ_d is related to blur radius and σ_r is related to contrast [1]. I also chose sigma values before performing multiple passes of image to get desired cartoon-effect.

3.2 Images of High Contrast

In this step, I observed the effects of the bilateral filter with its chosen parameters on a high contrast image. I tested several numbers of passes to get a desirable look.

3.3 Images of Low Contrast

In this step, I observed the effects of the bilateral filter on a low contrast image. I tested several numbers of passes to get a desirable look.

3.4 Images of High & Low Contrast

In this step, I observed the effects of the bilateral filter on an image with varying contrast in the foreground and background.

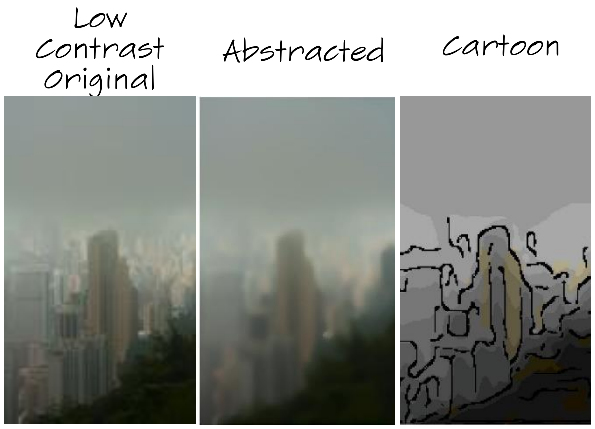

Figure 5: **Low Contrast Example.** *Low Contrast Original*: Hong Kong. *Abstracted*: 3 passes of the bilateral filter. Buildings are less visible. *Cartoon*: can't even make out any objects.

3.5 Edge Detection

In this step, I observed the effect of the Canny edge detector from MATLAB IPT on several passes of the bilateral filter. I also used *order statistic* filtering to bold edges to create a more hand-drawn look by making selecting the 4th order pixel of a 4-neighbor domain.

4 Results

In order to get the best cartoonized image, I had to observe the parameters for the bilateral filter and what effects it had on various types of images. In addition to this, I also observed the Canny Edge detector on how it responded to the several passes of the bilateral filter.

4.1 Effects of the Bilateral Filter

By changing the sigma values of the domain and range kernels of the bilateral filter, I was able to observe the effects the scalar had on the weight of this non-linear function. In Figure 2, there is noticieable change in range kernel. σ_r values

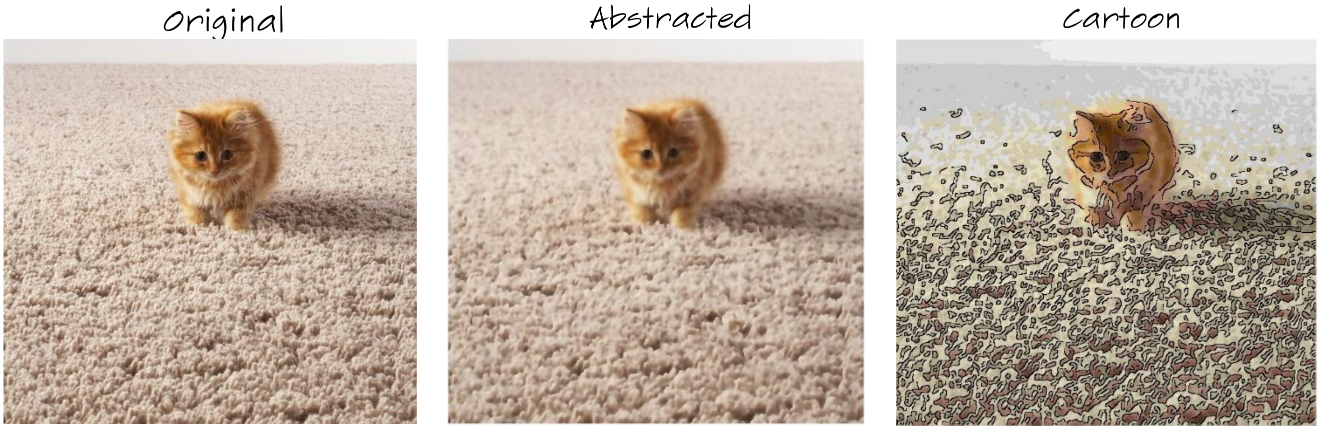

Figure 6: **Failure case.** A case with abstraction of high contrast background and low contrast foreground. *Original*: A cute fluffy orange tabby kitten. *Abstracted*: After 3 passes of bilateral filter. A lot of detail still shows in carpet as cute kitten is abstracted away. *Cartoon*: Edge detector picks up a lot of detail in carpet. Quantization also makes carpet pick up a lot of gray tones from color of the wall.

are on a scale from [0, 1]. As σ_r approaches 1, the contrast was less preserved and became further blurred. Figure 2 also shows the differences in σ_d but it does not show much sign of change as σ_d increased in value.

Number of passes of bilateral filter was also observed with bilateral filter. In Figure 3, abstraction shows that edges are still preserved but obvious signs of blurring occur as recursion of the bilateral filter goes through several passes.

4.2 High Contrast Image W/ Bilateral

Figure 4 shows an example of a high contrast image undergoing several passes of a bilateral filter. During tests this high contrast image was unable abstract the sunflower very much. Although the flower petals couldn't fully be abstracted, the flower still made out a cartoon look after the edge layer was applied and quantization was in effect.

4.3 Low Contrast Image W/ Bilateral

Figure 5 shows an example of a low contrast image undergoing several passes of a bilateral filter. During this test, many features were abstracted away making the buildings less visible in its abstracted version. During the quantization step the building couldn't even be recognized at all but the Canny edge detector was able to find some prominent edges.

4.4 Canny Edge Detector W/ Bilateral

In the bottom row of Figure 3 of the cartoonized selfie, examples of the effects of Canny edge detector show as it goes over several passes of the bilateral filter. Canny edge detector seems to detect fewer lines as the number of passes of the bilateral increases.

5 Discussion of Results

5.1 Happy Parameters

For the bilateral filter, I tested several parameters to get the most visually cartoonized picture for *Cartoonize Me!* Winnemöller, Olsen, and Gooch suggested keeping the parameters of the filter to be $\sigma_r = 4.25$ on a scale between (0, 255], $\sigma_d = 3$, number of passes before edge detection $n_e \in \{1, 2\}$, and number of passes before quantization step $n_b \in \{3, 4\}$ [1]. These parameters might be low for the real-time video abstraction but working on a single frame the number of passes can be increased as desired. I found to be satisfied with the following parameters: $\sigma_r \in \{0.5, 0.1\}$ on a scale of (0, 1], $\sigma_d = 3$ as there wasn't much significant change as this parameter was changed, $n_e = 3$, and $n_b \in \{2, 3\}$.

I preferred a more abstracted face when working on human subjects for my observation step because it could help smooth out details on the skin especially on facial features. This helped with the 16-bin quantization step to give the image paint-like effects.

The Canny edge detector in general was able to detect a lot of edges in first couple passes of the bilateral filter but on objects in the background like in Figure 3 it can be seen that some essential edges are lost as the bilateral filter goes through more passes. In the foreground more passes gives a much more desired effect on images like selfies since it outlines less detail on faces to give it a more hand-drawn look.

5.2 High vs. Low

Getting good results from the bilateral filter is very tricky in order to get the best cartoonized image. It had not much effect on the higher contrast image and the lower contrast image lost essential detail. *Cartoonize Me!* fails in these cases. Figure 6 shows an image of a kitten on carpet in which the higher contrast carpet was further emphasized by the bilateral filter whereas the kitten was successfully abstracted. It can be seen in the cartoon version of the abstracted image that a lot of edges were preserved in the carpet. During tests as the bilateral filter increased, less edges were detected in the carpet after 5 passes but not a significant amount. Not willing to have the patience to wait for about 100 or so passes, I can assume it would take too long to try to abstract the carpet away with a bilateral filter.

6 Suggestions and Conclusion

Limitations Bilateral filter seems to be limited on how easily it can abstract higher contrast images and lower contrast images. Higher contrast brings out too much detail and lower contrast loses too much detail. Bilateral filter may not be a good solution for these kinds of images. One could think that you can just lower contrast of an high contrast image and heighten the contrast of a low contrast image, but with images with varying contrast regions this solution isn't viable.

What one could do in the scenario of varying contrast regions, is to separate regions by some sort of region growing algorithm or watershed. Then take those region locations to come up with luminance histograms to determine if the region is of high or low contrast. Threshold to define the region of high or low contrast. Then, use a Gaussian kernel with a large standard deviation value on regions of super high contrast and a Gaussian kernel with a small standard deviation value on regions of super low contrast. Everything else can be abstracted with a bilateral filter. This solution is not tested by me but I think this could be a cool project for another undergraduate research assistant!

Execution time. Unfortunately execution time for MATLAB is very slow to recurse the bilateral filter for all 3 channels on the CIE-Lab space and as well as increase the number of passes. For example, a 1000x1000 image would take 1000^2 (# of pixels) x 3 (passes) x 3 (CIE-Lab channels) computations of the 3x3 kernel would take. This can take an excruciating amount of time! A good solution to this is to reduce the size of the image to avoid filtering a very large image.

Conclusion. Bilateral filter is not the only filter that can be used for abstraction. Bilateral filter can be difficult to optimize and it can be difficult to solve the failure case for high and low contrast images. Other abstraction methods by works from Li Xu can be nice alternatives to the cartoonization method: *L_0 Gradient Minimization* and *Structure Extraction from Texture via Relative Total Variation* are a couple of examples.

References

[1] WINNEMÖLLER , H., OLSEN, S. C., AND GOOCH, B. 2006. *Real-Time Video Abstraction.* ACM Trans. Graph. 25, 3, 1221–1226.

[2] SZELISKI, RICHARD. *Computer Vision: Algorithms and Applications*. Springer-Verlag, New York. 2010.

[3] *Quantization (image processing).* (n.d.). Retrieved April 5, 2016, from https://en.wikipedia.org/wiki/Quantization_(image_processing).