Academic Journal of Nawroz University (AJNU), Vol.14, No.1, 2025
 e-ISSN: 2520-789X
https://doi.org/10.25007/ajnu.v14n1a249

Cartoonize an Image Using Bilateral Filtering

Pawan Shivan Othman [1], Rasheed Rebar Ihsan [2], and Reving Masoud Abdulhakeem [1]

1 Department of Computer Science, College of Science, Nawroz University, Duhok, KRG – Iraq

2 Department of Computer Engineering, College of Engineering, Nawroz University, Duhok, KRG – Iraq

ABSTRACT: This paper introduces a fast method for turning images into cartoons in real-time. We use bilateral filtering, a strong smoothing technique that preserves important edges. Cartoonization is a popular image transformation that gives photos a cartoon-like look while keeping key details and edges. Our method uses bilateral filtering for quick performance, making it suitable for live video and mobile applications. We have a few main steps: first, we apply bilateral filtering to the image to smooth small details while keeping larger structures and edges. Then, we detect edges using methods like Canny edge detection. These edges are crucial for defining cartoon-like contours. To create the cartoon effect, we make a stylized color palette based on the image's color values and use it to limit the colors to a set of tones, like traditional cartoons. Finally, we overlay the cartoon edges and color palette on the original image, producing a real-time cartoon result. Our experiments show that our method is effective.

Keywords: Digital Image processing, Filtering, Edge preserving, Cartoonization, Bilateral Filter.

1. Introduction

Rotating real-life images into cartoon-style representations, known as Cartoonization, has become popular in entertainment, art, and image enhancement. An efficient method for achieving real-time Cartoonization is using bilateral filtering, an effective image processing technique. This study outlines a real-time Cartoonization approach that employs bilateral filtering. Bilateral filtering is a non-linear image filtering method that considers both spatial proximity and intensity similarity between pixels. This makes it suitable for Cartoonization as it enhances edges and stylizes images while preserving crucial details. The Cartoonization process begins by applying bilateral filtering to a conventional photograph or video frame, smoothing the image while preserving edges. This step simplifies textures and emphasizes important features, aligning with the clean and well-defined lines seen in typical cartoon-style images. After bilateral filtering, the image undergoes color quantization, reducing the number of distinct colors to achieve a posturized appearance similar to cartoon color palettes.

Techniques like k-means clustering may be used for color grouping. Subsequently, edges are extracted using edge detection algorithms like the Canny edge detector, emphasizing prominent contours and creating cartoon-style black outlines. To achieve the final cartoonized effect, the image is overlaid with the simplified color palette and edge information. The original pixel values are replaced with colors from the quantized palette, and edges are superimposed using a black outline technique, resulting in a distinctive cartoon-style appearance. Efficient implementations and optimizations of bilateral filtering algorithms, utilizing parallel processing on modern GPUs or dedicated hardware accelerators, enable seamless and interactive Cartoonization. This makes the approach suitable for real-time video streaming, interactive image editing, and various applications where a playful or stylized visual effect is desired. In conclusion, real-time Cartoonization using bilateral filtering is an effective way to transform photographs or video frames into cartoon-style representations while preserving important details and edges. This approach has diverse applications in fields such as entertainment, video conferencing, and beyond, offering a creative and practical solution.

2. Literature review

(Received: November 17, 2024; Accepted: March 24, 2025; Online: July 15, 2025)

The general concept of real-time image Cartoonization techniques, primarily based on bilateral filtering, is discussed along with key developments in this field up to a certain point in time. Real-time image Cartoonization involves the instantaneous or near-instantaneous transformation of regular images or videos into cartoon-like representations. This process typically employs various image processing algorithms, with bilateral filtering being a crucial component for achieving the desired cartoon effect. Several noteworthy works and points related to real-time image Cartoonization techniques using bilateral filtering are highlighted:

Bilateral Filtering for Cartoon Style Rendering (Tomasi and Manduchi, 1998): This groundbreaking work introduced bilateral filtering as a method for Cartoonization. It demonstrated the ability to control stylization effects by adjusting filter parameters, laying the foundation for subsequent research in this domain.

Real-Time Cartoon Animation (Paris et al., 2007): This research focused on optimizing the bilateral filter for real-time Cartoonization and showcased applications in interactive media and video games. This marked a significant advancement in the field.

Fast and Stable Cartoonization Using K-means Clustering (Li et al., 2013): This study proposed combining bilateral filtering with K-means clustering to enhance efficiency and stability in Cartoonization. It addressed limitations of earlier approaches, such as sensitivity to parameter settings.

Deep Learning-Based Cartoonization (Wang et al., 2020): Recent developments in deep learning have been applied to Cartoonization, with some researchers using convolutional neural networks (CNNs) to automatically learn cartoon-style transformations. This approach has shown impressive results in terms of visual quality and flexibility.

In summary, bilateral filtering has played a pivotal role in the evolution of image Cartoonization, serving as a powerful and versatile tool for creating cartoon-like representations. Over time, research in this field has progressed, leading to improved algorithms and real-time applications. The integration of deep learning approaches has further expanded the possibilities for Cartoonization, holding promise for enhancing the visual appeal of images and videos across various domains

3. The Main Concept Behind the Idea

The bilateral filter, employed in image processing and computer vision, is a non-linear smoothing filter designed to preserve edges while reducing noise in images. Its mathematical foundation involves weighted averaging, considering both spatial distance and intensity similarity. The key components of the bilateral filter encompass the spatial and intensity domains. In the spatial domain, a Gaussian function defines a neighborhood around each pixel, diminishing the impact of pixels distant from the center. Simultaneously, in the intensity domain, the filter evaluates the likeness in pixel intensities, assigning weight values based on intensity differences. A Gaussian function is instrumental in computing these weights, mitigating the influence of pixels with markedly distinct intensities. The filter equation involves the weighted average of pixel values within the spatial and intensity neighborhoods, providing the filtered pixel value. The formula for the bilateral filter at a pixel (x, y) is commonly expressed as:

$$I'(x,y) = \frac{1}{w(x,y)} \Sigma_{i=-k}^{k} \Sigma_{j=-k}^{k} w(\mathrm{i}, j, x, y) . I(x+\mathrm{i}, y+j) \qquad (1)$$

Where:

$I'(x,y)$ is the filtered pixel value at (x, y) , $w(x,y)$ is the normalization factor (sum of the $w(\mathrm{i}, j, x, y)$ is the weight for the pixel at (x + i, y + j), considering both spatial and intensity similarities $I(x+\mathrm{i}, y+j)$ is the intensity value at (x + i, y + j).

Weight calculation the weight, denoted as w (i, j, x, y), is determined by multiplying two Gaussian functions—one accounting for spatial distance and the other for intensity difference. These functions are characterized by distinct standard deviations (σ_s for spatial and σ_r for intensity)

$$w(\mathrm{i}, j, x, y) = e^{-\frac{i^2+j^2}{2\sigma_s^2}} \cdot \quad e^{-\frac{||I(x+i,y+j)-I(x,y)||^2}{\sigma_r^2}} \qquad (2)$$

The initial component of the product accounts for the spatial separation, diminishing the influence of pixels located distantly from (x, y). Simultaneously, the second component addresses the likeness in intensity, diminishing the influence of pixels exhibiting significantly disparate intensities. By employing this filter on each pixel within the image, you effectively achieve image smoothing while retaining edges and intricate details. The smoothing strength and edge preservation attributes of the filter are determined by the values of σ_s and σ_r. Opting for smaller values results in heightened edge preservation but less pronounced smoothing, whereas larger values yield more substantial smoothing albeit with a potential for greater edge blurring. The selection of these parameters is contingent on the specific application and the intended outcome.

4. Proposed Methodology

In the proposed analysis of noisy RGB images, an advanced bilateral filter serves as a preprocessing method to prepare the image for the subsequent detection step. The detection method employed is the Canny edge detector, which identifies image features based on grayscale levels, as elaborated in [13]. Figure 1 illustrates the block diagram outlining the methodology of the proposed model. The techniques for conversion and filtering closely align with those described in [14].

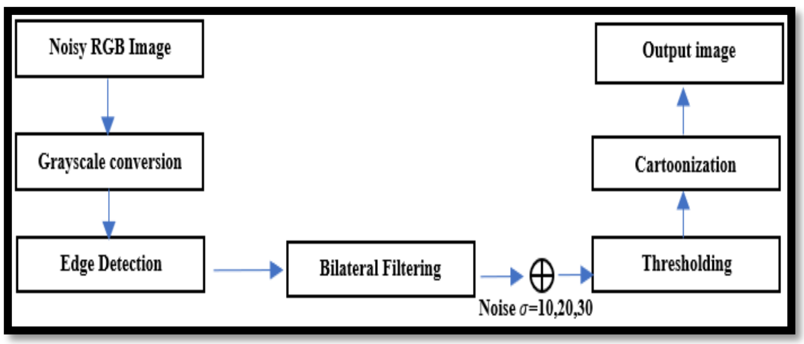

Figure 1: General diagram of the proposed method

To achieve a cartoonized effect using bilateral filtering in MATLAB, follow these steps:

Load the Image: Utilize the imread function to load the desired image for Cartoonization into MATLAB. Convert to Grayscale: Use the rgb2gray function to simplify the image by removing color information. Apply Bilateral

Filtering for Smoothing: Utilize the bfilter2 function from the MATLAB Image Processing Toolbox to apply bilateral filtering for smoothing. Adjust the sigma_r and sigma_s parameters to control the smoothing effect.

Edge Detection: Employ an edge detection algorithm, such as the Laplacian filter, to identify edges in the image.

Increase Edge Contrast: Adjust the contrast of the detected edges using imadjust to make them more pronounced and stylized. Thresholding with Histogram Analysis: Apply thresholding to the edge image, considering the image histogram to determine a suitable threshold for different images.

Combine Smoothed Image Thresholded Edges: Merge the smoothed image with the Thresholded edges to emphasize edges and create a cartoon-like appearance. In this study, we multiply the grayscale image by (1 – binary Edges) to retain non-edge regions and multiply by 255 * binary Edges to set edge regions to white (255).

Display the Cartoonized Image with Histogram Analysis: Showcase the cartoonized image alongside the computed threshold. These steps collectively form a comprehensive guide for achieving a cartoon-like effect through bilateral filtering in MATLAB.

5. Results

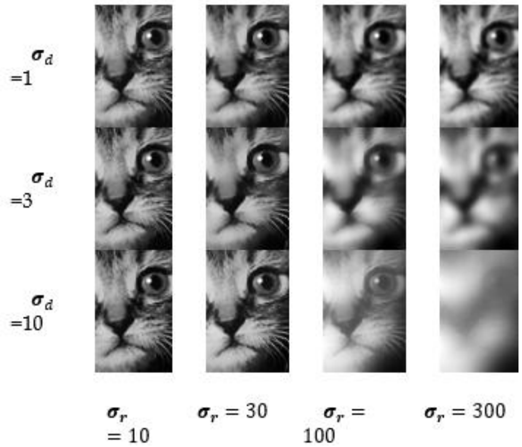

Figure 2: A detail from figure 5 (a) processed with bilateral filters with various range and domain parameter values.

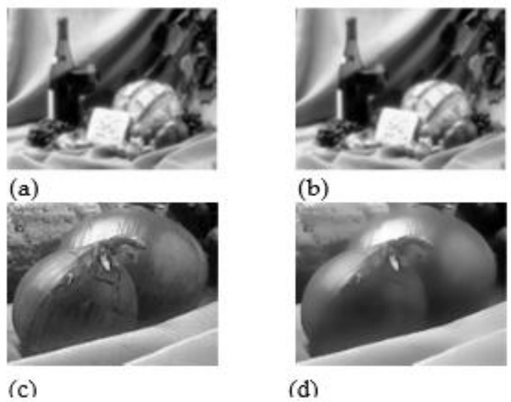

Figure 3: A picture before (a) and after (b) bilateral filtering. (c, d) are details from (a, b).

(a)

(b)

Figure *4:* A picture before (a) and after (b) bilateral filtering

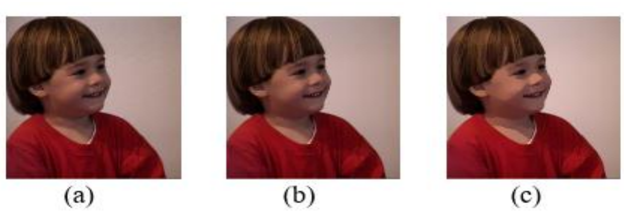

Figure 5*:* [left] (a) A detail from a picture with a red jacket against a blue sky. The thin, pink line in

(a) is spread and blurred by ordinary lowpass filtering (b). Separate bilateral filtering (c) of the red, green, blue components sharpens the pink band, but does not shrink

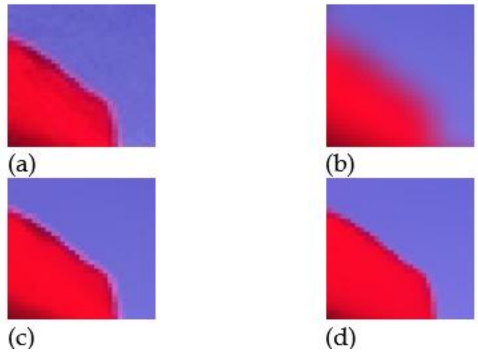

Figure 6*:* [above] (a) A color image, and its bilaterally smoothed versions after one (b) and five (c) iterations.

6. Conclusion

This study presented an efficient and real-time approach for image Cartoonization using bilateral filtering, a powerful edge-preserving smoothing technique. By leveraging bilateral filtering to reduce image noise while maintaining sharp edges, followed by edge detection and color quantization, the proposed methodology successfully converts standard photographs into stylized cartoon-like visuals. The method balances visual simplification with structural integrity, making it ideal for applications in digital entertainment, mobile graphics, and artistic rendering.

Experimental results demonstrated that bilateral filtering significantly enhances the visual quality of cartoonized images, especially when optimized with appropriate spatial and range parameters. Moreover, the integration of the Canny edge detector and adaptive thresholding contributes to clear and expressive contour detection, further reinforcing the cartoon effect. The technique's compatibility with MATLAB allows for practical implementation and rapid prototyping, while its computational efficiency renders it suitable for real-time processing scenarios.

Future work may explore GPU-accelerated bilateral filtering, integration with deep learning frameworks for automatic parameter tuning, and application in dynamic video streams. Overall, this work contributes a valuable solution to the field of image stylization and demonstrates the effectiveness of bilateral filtering as a core component for cartoon image generation.

7. References

[1] Saka, M. P., Hasan.ebi, O., and Geem, Z. W. (2016). "Metaheuristics in structural optimization and discussions on harmony search algorithm." Swarm and Evolutionary Computation, 28, 88–97.
[2] S.rensen, K. (2013). "Metaheuristics: The metaphor exposed." International Transaction in Operational Research, 22(1), 3–18.
[3] Dokeroglu, T.; Sevinc, E.; Kucukyilmaz, T.; Cosar, A. A survey on new generation metaheuristic algorithms. Comput. Ind. Eng. 2019, 137, 106040.
[4] Kondamadugula, S.; Naidu, S.R. Accelerated evolutionary algorithms with parameter importance-based population initialization for variation-aware analog yield optimization. In Proceedings of the 2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS), Abu Dhabi, United Arab Emirates, 16–19 October 2016.
[5] Battiti R, Brunato M, Mascia F (2008) Reactive search and intelligent optimization. Springer, Berlin
[6] Eiben A, van der Hauw J (1997) Solving 3-sat with adaptive genetic algorithms. In: Proceedings of the fourth IEEE conference on evolutionary computation. IEEE Press, pp 81–86
[7] Eiben AE, Smith JE (2003) Introduction to evolutionary computing. Springer, Berlin
[8] Gent IP, Walsh T (1993) Towards an understanding of hill-climbing procedures for SAT. Proceedings of AAAI- 93:28–33
[9] Glover F, Laguna M (1993) Tabu search. In: Reeves C (ed) Modern heuristics techniques for combinatorial problems. Blackwell Scientific Publishing, Oxford, pp 70–141
[10] Larranaga P, Lozano JA (2001) Estimation of distribution algorithms. A new tool for evolutionary computation. Kluwer Academic Publishers, Boston
[11] Marchiori E, Rossi C (1999) A flipping genetic algorithm for hard 3-sat problems. Proceedings of the Genetic and Evolutionary Computation Conference 1:393–400