Grenze International Journal of Engineering and Technology, January Issue

FPGA-based Implementation of Dark Channel Prior for Image Dehazing

Anita B. Agrawal[1], K. Richa[2], Karun Jain[3], Vasanth Gowda M K[4] and Vinukonda Sreekar[5]
[1-5]BITS Pilani, K. K. Birla, Goa Campus, Goa, India
Email: h20240130@goa.bits-pilani.ac.in, aagrawal@goa.bits-pilani.ac.in, f20221494@goa.bits-pilani.ac.in, h20240132@goa.bits-pilani.ac.in, h20240087@goa.bits-pilani.ac.in

***Abstract*— Image degradation due to atmospheric haze significantly impairs visual quality and the performance of computer vision systems. This paper presents a Verilog-based hardware system for image dehazing, primarily implementing the Dark Channel Prior (DCP) algorithm. The system architecture is designed for Field-Programmable Gate Array (FPGA) deployment, emphasizing a pipelined approach for high throughput. Key components include modules for dark channel calculation, atmospheric light estimation, transmission map estimation, and guided filter-based transmission map refinement. An optional processing path incorporating YCbCr colour space conversion is included for targeted brightness enhancement of the dehazed image.**

This paper details the theoretical underpinnings of the implemented algorithms, the specifics of the hardware architecture including Verilog module design, and a comprehensive performance evaluation. Results demonstrate the system's capability for efficient haze removal, with analysis of processing speed, FPGA resource utilization, power consumption, and quantitative dehazing quality.

***Index Terms*— Verilog, Image Processing, Dark Channel Prior (DCP), Haze Removal, FPGA.**

I. INTRODUCTION

In the Initial days, photography was done in analog mode. Images were captured on photo-films. With the advent of digital photography, earlier analog cameras have become obsolete. Images captured in digital cameras can be processed, enhanced, and improved by photo-editing software freely available. However, these targeted image capturing, enhancement are typically done in offline mode.

Nowadays, various applications like ECG monitoring, Surveillance Cameras, Intruder Detection and Advanced Driver Assistance Systems (ADAS) require real-time image capturing and processing of the targeted subject. These real-time applications as mentioned above, face problems due to the presence of atmospheric interference, which leads to the capture of hazy images. In these digital devices, an image is captured by reflected light from the target subject. Sometimes, these light rays get blocked or scattered due to obstructing atmospheric particles like dust, fog, smoke etc. This phenomenon makes images blurry. This effect is known as haze. The resulting image has less contrast and looks dull. In extreme conditions, identification of the target object may become difficult in such conditions. To obtain a clear visual and object identification, a process known as image dehazing is required.

Grenze ID: 01.GIJET.12.1.83_1

II. Literature Studies

Image dehazing has been extensively researched, with Dark Channel Prior (DCP) being a well-known method for giving good results. In most non-sky regions of haze-free images, it has been observed that at least one colour channel has very low intensity. This method helps in estimating haze thickness and recovering a high-quality dehazed image. However, it fails in cases where the scene contains bright, haze-like regions [1]. A fast edge-preserving image filter that uses a guidance image to get a clear, smooth image while preserving sharp edges has been introduced in this paper. It supports dehazing, matting, and up-sampling. However, it might produce light halo effects near high-textured areas [2]. J. Xiao et.al. [3] introduce a haze-removal method that uses sky-region segmentation before applying the Dark Channel Prior method to handle sky and non-sky areas differently. It improves the appearance of the sky, but it is extremely complex as it depends on manual tuning of gamma values. For dehazing, [4] implemented a faster and more efficient method for haze removal by modifying the traditional dark channel prior (DCP). It reduces visual artifacts near sharp edges by eliminating the extra refinement step, thus speeding up the processing. The main drawback of this method is that it can be affected by noise in the image, leading to less accurate haze removal. B. Zhang et. al. [5] implemented a real-time haze removal system on FPGA using the Dark Channel Prior (DCP). It introduced an efficient approximation method for estimating transmission and atmospheric light. It delivered fast and high-quality dehazed images but involved complex circuit design. Another approach by K. Tan et. al. [6] proposed a model-based method to enhance colour images taken in hazy conditions, especially from airborne cameras. By estimating atmospheric effects from the image, contrast and visibility range were restored more accurately. It performs well under different visibility levels, but it involves complex parameter estimation. M. Majoka et. al. [7] described an FPGA-based architecture for real-time image dehazing using the Dark Channel Prior and fast guided filtering. RGB images in hardware are processed by dividing them into patches. It was fast and hardware-friendly, but involved complex implementation. J.-P. Tarel et. al. [8] presented a fast algorithm to restore visibility in hazy images using a filtering approach that worked on both colour and grayscale images. While it produced good results and was highly efficient, it sometimes missed small details and needed adjustments in parameters like restoration strength and smoothness. One method involved a fast and effective single-image dehazing method using a colour attenuation prior. By learning from training data, it estimated scene depth and removed haze efficiently [9]. D. Li et. al. [10] discussed an improved haze image enhancement method which included refining the transmission map. Although image visibility was significantly improved, it performed poorly when used in real-time. X. Su et. al. [11] designed an FPGA-based computer vision system that achieved real-time image processing through dedicated interfaces and clock domains. However, the design was complex and required close interaction between hardware and software, making its implementation difficult. In 2020, A. Metkar et.al. described an edge detection processing algorithm to achieve faster and efficient performance. However, this method required careful design handling [12].

III. Proposed Method

Dark Channel Prior (DCP) method is a common method used for image dehazing. The primary concept behind this technique is that in most outdoor images, at least one colour channel (Red, Green or Blue) will have very low intensity in non-hazy areas. But in hazy images, it becomes brighter due to the presence of haze. So, by computing the dark channel we can estimate the amount of haze present and get a clearer image. The following steps are involved in this process:

A. JPG to BMP Conversion

Hardware Description Languages (HDLs) such as Verilog are not designed to directly process compressed image formats, as they can't handle high-level image data. To process the image, pixel values must be in an uncompressed form. Thus, a conversion step is performed using software where the input image is transformed into BMP format, and each raw pixel value is stored in memory, which can later be accessed during simulation. In our implementation, the input image was in JPG format, which was converted to BMP format using a Python program.

B. Dark Channel Estimation

Next Step involves computing the Dark Channel, where we start with separating each pixel into its R, G, B parts. The smallest value amongst the 3 colours is found out and this value tells us how dark the pixel is. This is done using the formula:

$$min\ (R, G, B) \tag{1}$$

For understanding how dark a pixel is, we look at a small 3x3 area of neighbouring pixels around it. Inside this block, the system computes the smallest value amongst all 9 pixels. This gives us the final dark channel image. Mathematically, it is represented as:

$$DarkChannel(x) = min_{y\in\Omega(x)} \quad (min(I^R(y), I^G(y), I^B(y))) \tag{2}$$

where DarkChannel(x) = The final dark channel value at pixel position x; x is the current pixel position being processed. This indicates how dark the pixel is within its neighbourhood. Ω(x) = A small window (e.g., 3×3) centered around pixel x. $I^R(y)$, $I^G(y)$, $I^B(y)$ = Red, Blue and Green Intensity values of pixel at position y within the window Ω(x).

C. Atmospheric Light Estimation

Next, the goal is to estimate the global atmospheric light, which will be used to recover the scene radiance. This is done by first tracking the brightest pixels in the dark channel image, as these are mostly affected by haze. So a buffer is used so as to keep the top 0.1% brightest dark channel values, and the maximum value from the buffer is chosen as the atmospheric light estimate.

D. Transmission Map

Transmission Map tells us how much light has reached the camera without being scattered by haze. It is calculated using formula:

$$t(x) = 1 - \omega \times \frac{I_{dark}(x)}{A} \tag{3}$$

where t(x) = amount of light that reaches camera at pixel x, ω = A constant to determine how much of haze can be kept in the image for a natural view, $I_{dark}(x)$ = Dark channel value at pixel x and A = Atmospheric Light.

E. Time Alignment

For image processing, we require two images: one coloured and the other, greyscale. These images need to be synchronized with each other because they come from different sources. In this process, we ensure that both these images are in synchronization and can be processed together.

F. RGB to YCbCr Conversion

This module converts RGB format to YCbCr format. YCbCr is a colour space used in image processing. Y represents the brightness of the image, Cb and Cr represent how much blue and red are present in relation to brightness. It's done as human vision is sensitive to brightness so, it gets easier to compress the image if we separate the brightness and colour information.

G. YCbCr to RGB Colour Conversion

To display output on screen, YCbCr format is required for storage and compression of video streams, the image is converted back to RGB. The final value should stay between the range 0-255 so as to get valid output on the screen.

H. Haze Removal

The main aim of this system is to remove haze from an image. It does this by adjusting the Red, Green and Blue colour channels and using the formula:

$$J(x) = \frac{I(x) - A}{t(x)} + A \tag{4}$$

where $J(x)$ = dehazed image, $I(x)$ = original hazy image, $t(x)$ = transmission map and A = Atmospheric Light. This is done for each colour channel separately.

IV. Simulation Results

We implemented the dehazing algorithm on an FPGA using Xilinx Vivado 2024.1. The proposed system was designed to take input images in BMP format for better compatibility and ease of processing. The input images in JPG format were initially converted to BMP format using Python programming and fed to the system. Using Verilog HDL, it was analyzed and processed to give the final desired output. The results obtained from the

simulation for image dehazing using the proposed algorithm were tested on 50 locally stored images. After processing, the output images showed significant improvement, with clearer and distinct edges. Previously hidden details became visible with brighter contrast.
Although the results were largely positive, it was observed in images containing sunlight or large sky regions, that these bright areas appeared darker after processing. This happened because the dehazing algorithm misinterpreted such regions to be hazy, leading to over-correction. This is one of the limitations of the Dark Channel Prior (DCP) method. Further improvements can be made to address this issue without compromising the image quality.

Figure 1. Experimental Results using the DCP method. The top row shows the input hazy images, while the bottom row shows the corresponding dehazed outputs

V. Applications

Image dehazing plays a very significant role in different areas of everyday life. From autonomous cars to surveillance drones, it finds diverse applications that rely on clear visual data. In autonomous driving systems, hazy weather conditions such as fog and smoke can significantly reduce the performance of cameras, affecting obstacle recognition, lane detection, and more. Dehazing algorithms help restore scene visibility, enabling safer navigation. Another significant application lies in the field of security and surveillance, where dehazing techniques are applied to images captured under adverse conditions, thereby reducing the effect caused by low visibility conditions, helping in object recognition and tracking. It also finds application in underwater imaging, where the visibility of the image is essential for marine research and exploration. Apart from this, there are many such real-world applications where clear and accurate images are essential.

VI. Conclusion

In this paper, we have successfully implemented image dehazing using the Dark Channel Prior (DCP) method on an FPGA-based hardware implementation, targeting the processing of locally stored images. While this algorithm has proven to be effective in removing haze and enhancing image clarity, with further optimization and improvements, the system can be expanded to handle real-time applications where quick response and high performance are critical.

References

[1] K. He, J. Sun and X. Tang, "Single Image Haze Removal Using Dark Channel Prior," IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 12, pp. 2341–2353, Dec. 2011.

[2] K. He, J. Sun and X. Tang, "Guided Image Filtering," IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 6, pp. 1397–1409, June 2013.

[3] J. Xiao, L. Zhu, Y. Zhang, E. Liu, and J. Lei, "Scene-aware image dehazing based on sky-segmented dark channel prior," IET Image Process., vol. 11, no. 12, pp. 1163–1171, Dec. 2017.

[4] S. Salazar-Colores, J.-M. Ramos-Arreguín, J.-C. Pedraza-Ortega, and J. Rodríguez-Reséndiz, “Efficient single image dehazing by modifying the dark channel prior,” EURASIP Journal on Image and Video Process., vol. 2019, no. 1, p. 66, May 2019.

[5] B. Zhang and J. Zhao, “Hardware Implementation for Real-Time Haze Removal,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 25, no. 3, pp. 1188-1192, Mar. 2017.

[6] K. Tan and J. P. Oakley, “Enhancement Of Colour Images In Poor Visibility Conditions,” Proc. IEEE Int. Conf. Image Processing (ICIP), vol. 3, pp. 561–564, Sept. 2000.

[7] M. Majoka and G. Raja, “FPGA based image de-hazing architecture for real-time applications,” in Proc. IEEE Int. Conf. Electron., Comput. Commun. Technol., 2018, pp. 1–6.

[8] J.-P. Tarel and N. Hautière, “Fast visibility restoration from a single colour or gray level image,” in Proc. IEEE Int. Conf. Comput. Vision, Sep. 2009, pp. 2201–2208.

[9] Q. Zhu, D. J. Mai, and L. Shao, “A fast single image haze removal algorithm using colour attenuation prior.” IEEE Trans. Image Process., vol. 24, no. 11, pp. 662-673, Nov. 2015.

[10] D. Li, J. Sun, H. Wang, H. Shi, W. Liu, and L. Wang, “Research on haze image enhancement based on dark channel prior algorithm in machine vision,” Journal of Environmental and Public Health, vol. 2022, pp. 1–12, Jul. 2022, doi: 10.1155/2022/3887426.

[11] X. Su and G. Zuo, “Computer artificial vision image processing system based on FPGA,” in Lecture Notes on Data Engineering and Communications Technologies, vol. 85, Springer Singapore, 2022, pp. 194–201.

[12] A. Metkar, M. Maroo, A. Sawant, V. Singh, and S. Mhatre, “Hardware implementation of image processing algorithms on FPGA,” SSRN Electronic Journal, 2020, doi: 10.2139/ssrn.3568140.