

Image Processing Operations for 3d Image

Mrs. A. Padmapriya, S.Vigneshnarthi

Department of Computer Science & Engineering, Alagappa University, Karaikudi – 630 003, India

Abstract- Image processing and pattern recognition has become a powerful technique in many areas. This includes Engineering, Computer Science, Statistics, Information Science, Physics, Chemistry and Medicine. Anyone who wants to extract data from image or visual project, image processing is required. Most of the image processing software's are able to process two dimensional images alone. This paper implements three basic image processing operations namely enhancement, blur and segmentation. Each of the operation can be implemented using a variety of algorithms. The algorithms are implemented using and their performance are compared in this paper. The proposed work produces better result of 3D images also.

Index Terms- Image processing, Enhancement, Blur, Segmentation, 3D Image

I. INTRODUCTION

Image Processing

Image processing [3] is a form of signal processing for which the input is an image, such as photographs; the output of image processing can be either an image or a set of characteristics or parameters related to the image. Most image-processing techniques involve treating the image as a two-dimensional signal and applying standard signal-processing techniques to it. This is done through the development and implementation of processing means necessary to operate on the image. Processing image using a digital computer provides the greatest flexibility and power for general image processing application, since the programming of a computer can be changed easily which allows operation to be modified quickly. Interest in image processing technique dates back to early 1920's when digitized pictures of world news events were first transmitted by submarine cable between New York and London.[4]. However, application of digital image processing concepts did not become widespread until the middle 1960's, when third-generation digital computers began to offer the speed and storage capabilities required for practical implementation of image processing algorithms. Since then, this area has experienced vigorous growth and has been subjected of study and research in various fields. Image processing and computer vision practitioners tend to concentrate on a particular area of specialization [5] research interests as "texture", "surface mapping", "video tracking", and the like. Nevertheless, there is a strong need to appreciate the spectrum and hierarchy of processing levels.

The paper is organized as follows: Section II briefly review the literature behind the proposed work. Section III elaborates the proposed work. Section IV discusses the implemented of the proposed work. Section V concludes the findings.

II. LITERATURE SURVEY

Motivation for a digital image processing course [1] arises mainly from two areas of applications, namely, a) an increase in the pictorial information available for human interpretation, and b) image processing for automatic and autonomous machine control. This is because vision is the most important human sense in terms of the amount of information it conveys and because of a good visualization is very important for the correct information converged in an image. Many image processing circuits require dedicated software to perform their tasks. These packages usually are highly priced and are not easily modified by the final user. In this paper the basis for the image processing is MATLAB [3], a software package now available almost anywhere for other uses and that is used as the engine for the image processing experiments. The software written for the image processing experiments is available from the users at no cost to interested users and instructors. Image processing applications include many topics, among which we can mention remote sensing, ultrasound images, meteorology, astronomy, inspection, radar, seismology, radiology, autonomous navigation, recognition, etc.

III. PROPOSED WORK

This paper deals with the implementation of three basic image processing operations namely image enhancement, segmentation and blur.

A. Image Enhancement

Image Enhancement of image means processing an image so that the results are more suitable for a particular application. Processing an image so that the result is more suitable for a particular application. (sharpening or deblurring an out of focus image, highlighting edges, improving image contrast, or brightening an image, removing noise)

Enhancing Multispectral Color Composite Images

Enhance multispectral radiance or reflectance data to create an image that is suitable for visual interpretation. Another way to enhance the true color composite is to use a decorrelation stretch, which enhances color separation across highly correlated channels. Use decorrstretch[6] to perform the decorrelation stretch. Again, surface features have become much more clearly visible, but in a different way. The spectral differences across the scene have been exaggerated.

Enhance Color Images

Contrast enhancement of color images is typically done by transforming an image to a color space that has image intensity as one of its components. One such color space is $L^*a^*b^*$. Use

color transform functions to convert the image from RGB to $L^*a^*b^*$ color space, and then work on the luminosity layer 'L*' of the image. Manipulating luminosity affects the intensity of the pixels, while preserving the original colors. Using the default settings, compare the effectiveness of the following three techniques:

- a) *Imadjust*
- b) *Histeq*
- c) *Adaphisteq*

Imadjust

It increases the contrast of the image by mapping the values of the input intensity image to new values such that, by default, 1% of the data is saturated at low and high intensities of the input data.

Histeq

It performs histogram [8] equalization. It enhances the contrast of images by transforming the values in an intensity image so that the histogram of the output image approximately matches a specified histogram (uniform distribution by default).

Adaphisteq

It performs contrast-limited adaptive histogram equalization. Unlike histeq, it operates on small data regions (tiles) rather than the entire image. Each tile's contrast is enhanced so that the histogram of each output region approximately matches the specified histogram (uniform distribution by default). The contrast enhancement [7] can be limited in order to avoid amplifying the noise which might be present in the image. Notice that imadjust had little effect on the image of the tire, but it caused a drastic change in the case of pout. Plotting the histograms of pout.tif and tire.tif reveals that most of the pixels in the first image are concentrated in the center of the histogram, while in the case of tire.tif, the values are already spread out between the minimum of 0 and maximum of 255 thus preventing imadjust from being effective in adjusting the contrast of the image.

Histogram equalization, on the other hand, substantially changes both images. Many of the previously hidden features are exposed, especially the debris particles on the tire. Unfortunately, at the same time, the enhancement over-saturates several areas of both images. Notice how the center of the tire, part of the child's face, and the jacket became washed out. Concentrating on the image of the tire, it would be preferable for the center of the wheel to stay at about the same brightness while enhancing the contrast in other areas of the image. In order for that to happen, a different transformation would have to be applied to different portions of the image. The Contrast-Limited Adaptive Histogram Equalization technique, implemented in adaphisteq, can accomplish this. The algorithm analyzes portions of the image and computes the appropriate transformations. A limit on the level of contrast enhancement can also be set, thus preventing the over-saturation caused by the basic histogram equalization method of histeq. This is the most sophisticated technique in this demonstration.

B. Segmentation

Segmentation is the process of partitioning a digital image into multiple segments. The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics.

An object can be easily detected in an image if the object has sufficient contrast from the background. We use edge detection and basic morphology tools to detect a prostate cancer cell.

The steps involves are below:

Step 1: Detect Entire Cell

Step 2: Fill Gaps

Step 3: Dilate the Image

Step 4: Fill Interior Gaps

Step 5: Remove Connected Objects on Border

Step 6: Smooth the Object

Step 1: Detect Entire Cell

Two cells are present in this image, but only one cell can be seen in its entirety. We will detect this cell. Another word for object detection is *segmentation*. The object to be segmented differs greatly in contrast from the background image. Changes in contrast can be detected by operators that calculate the gradient of an image. One way to calculate the gradient of an image is the Sobel operator, which creates a binary mask using a user-specified threshold value. We determine a threshold value using the graythresh function. To create the binary gradient mask, we use the edge function.

Step 2: Fill Gaps

The binary gradient mask shows lines of high contrast in the image. These lines do not quite delineate the outline of the object of interest. Compared to the original image, you can see gaps in the lines surrounding the object in the gradient mask. These linear gaps will disappear if the Sobel image is dilated using linear structuring elements.

Step 3: Dilate the Image

The binary gradient mask is dilated using the vertical structuring element followed by the horizontal structuring element.

Step 4: Fill Interior Gaps

The dilated gradient mask shows the outline of the cell quite nicely, but there are still holes in the interior of the cell.

Step 5: Remove Connected Objects on Border

The cell of interest has been successfully segmented, but it is not the only object that has been found. Any objects that are connected to the border of the image can be removed.

Step 6: Smooth the Object

Finally, in order to make the segmented object look natural, we smooth the object by eroding the image twice with a diamond structuring element. An alternate method for displaying the segmented object would be to place an outline around the segmented cell.

C. Blur

Blurring an image usually makes the image unfocused. It increased by increasing the pixel radius

Blur Deconvolution

The Blur Deconvolution Algorithm can be used effectively when no information about the distortion (blurring and noise) is known. The algorithm restores the image and the point-spread function (PSF) [9] simultaneously. What happens is that what should be seen as a sharp point gets smeared out, usually into a disc shape. In image terms this means that each pixel in the source image gets spread over and mixed into surrounding pixels. Another way to look at this is that each pixel in the destination image is made up out of a mixture of surrounding pixels from the source image

IV. EXPERIMENT STUDY

Image can be processed using any software programming language such as C language Matrix-X. Visual Basic. Java program and MATLAB program. Image applications frequently need to interactively manipulate an image by continuously changing a single parameter and multi-parameter. This paper briefly elaborates the image process operations for 3D images. Three major operation selected for implementation are enhancement, segmentation and blur.

A. Enhancement

In the papers [2,6], the Enhancement technique is used for 2D colour images. Enhance multispectral radiance or reflectance data to create an image that is suitable for visual interpretation. Another way to enhance the true color composite is to use a decorrelation stretch, which enhances color separation across highly correlated channels. Use decorrstretch to perform the decorrelation stretch. Again, surface features have become much more clearly visible, but in a different way. The spectral differences across the scene have been exaggerated.

In this paper, 3D images are used for Enhancement technique and three functions imadjust, histeq, and adapthisteq are used for contrast enhancement.



ORIGINAL IMAGE



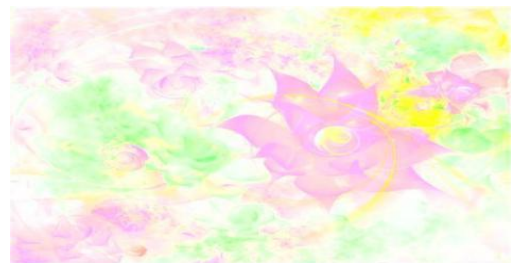
IMADJUST



IMADJUST



HISTEQ



ADAPTHISTEQ



Image Composite after Decorrelation Stretch

B. Segmentation

In the papers [1-4] segmentation can be implemented for 2D method.

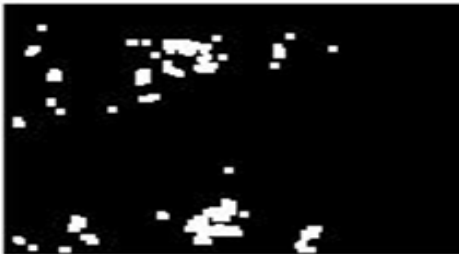
In this proposed work, segmentation implemented for 3D method.



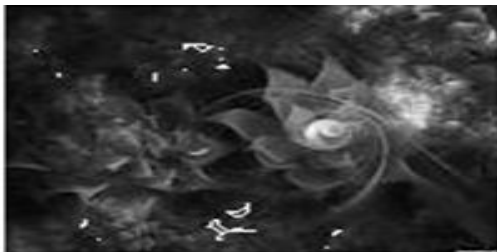
BINARY GRADIENT MASK



BINARY IMAGE FILLED WITH HOLES



CLEARED BODER IMAGE



ORIGINAL IMAGE

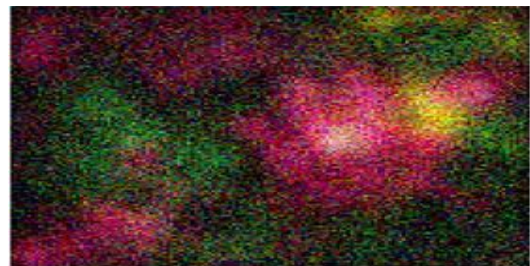


SEGMENTED IMAGE

B. Blur

In the paper [9] Blind Blur Deconvolution Algorithm for 2D method can be used effectively when no information about the distortion (blurring and noise) is known. The algorithm restores the image and the simple blur algorithm simultaneously. What happens is that what should be seen as a sharp point gets smeared out, usually into a disc shape. In image terms this means that each pixel in the source image gets spread over and mixed into surrounding pixels. Another way to look at this is that each pixel in the destination image is made up out of a mixture of surrounding pixels from the source image.

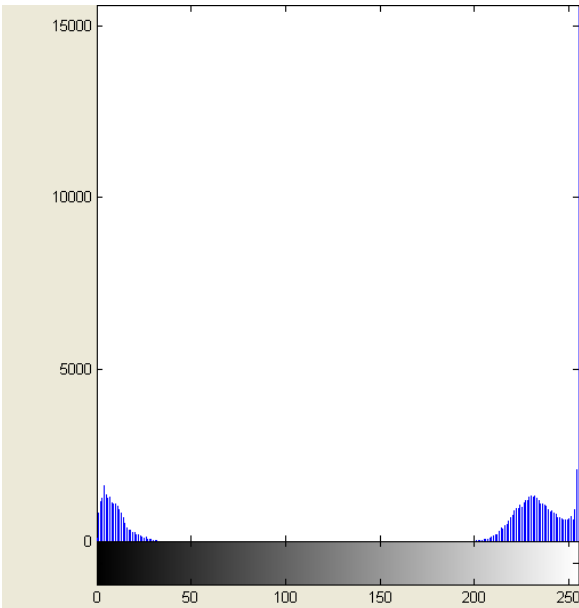
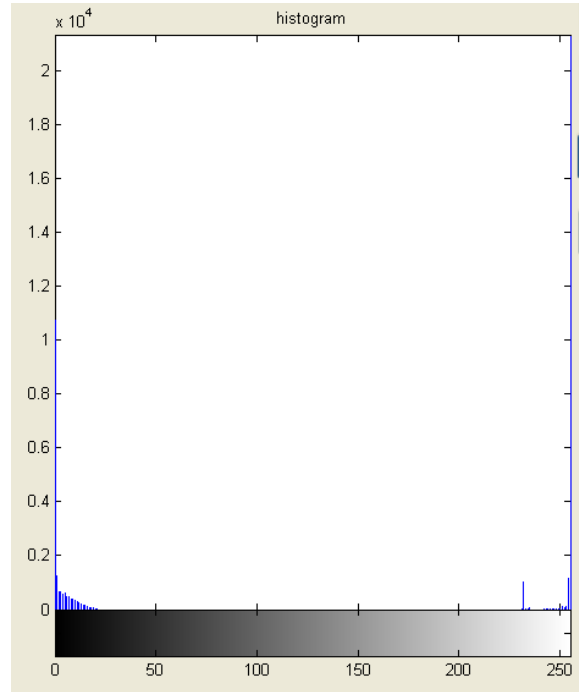
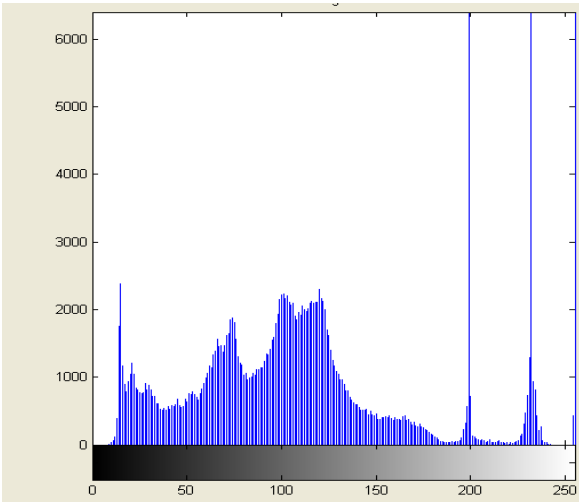
In this paper, Blur Deconvolution Algorithm for 3D method can be used effectively when no information about the distortion (blurring and noise) is known. The algorithm restores the image and the point-spread function (PSF) simultaneously. In image terms this means that each pixel in the source image gets spread over and mixed into surrounding pixels. Another way to look at this is that each pixel in the destination image is made up out of a mixture of surrounding pixels from the source image.



BLURRED AND NOISY IMAGE

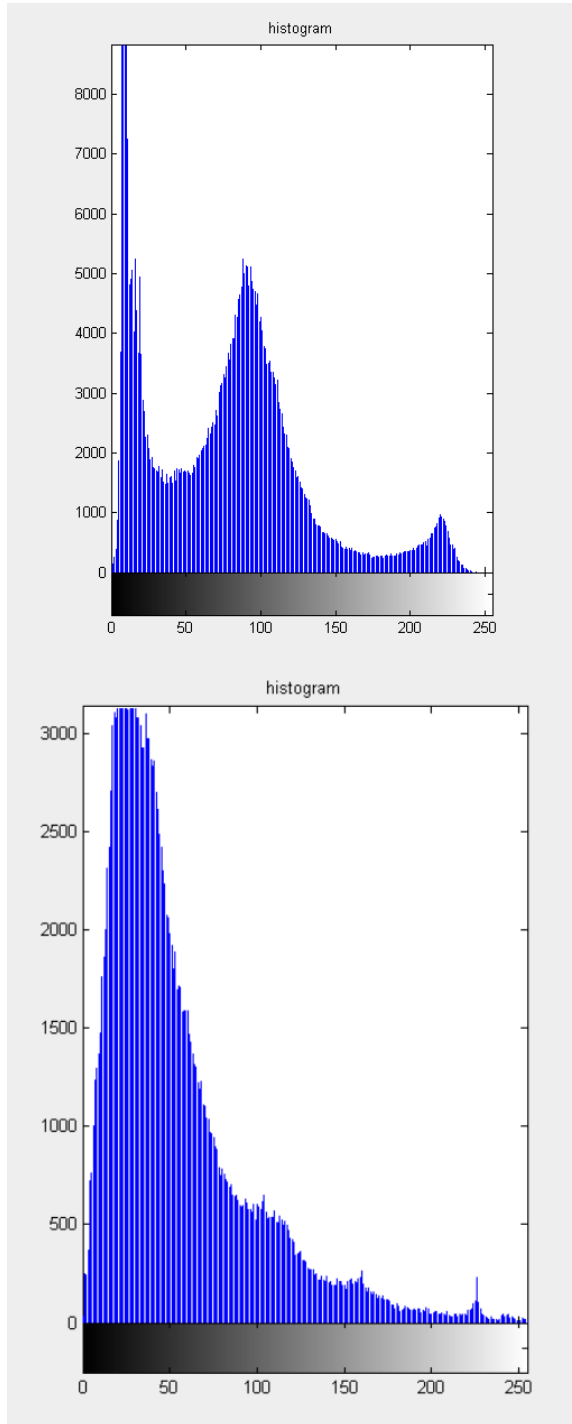
Comparison histogram for enhancement for 2d and 3d images:

Comparison histogram for segmentation for 2d and 3d images:



Comparison histogram for segmentation for 2d and 3d images:

Comparison histogram for blur for 2d and 3d images:



V. CONCLUSION

This paper briefly elaborates the image process operations for 3D images. The three major operations selected for implementation are enhancement segmentation and blur. This paper briefly the software used for implementation is matlab. The outputs of the algorithms are compared. Previously there operations are done used 2D images only. The proposed work processes 3D images.

REFERENCES

- [1] ABDUL HALIM BIN BABA. image processing learning tool-edge detection bachelor degree. university of technology malaysia 1996.
- [2] FIONN MURTAGH. Image Processing data analysis. The multi-scale approach. University of Ulster.
- [3] Fundamentals of image processing, hany.farid@dartmouth.edu .(<http://www.cs.dartmouth.edu/~farid>)
- [4] JEAN-LUC STARCK, centre d' etudes de Saclay, Fionn Murtagh. Image processing and data analysis the multiscale approach. University of Ulster.
- [5] LOUIS J.GALBIATI, JR. machine vision and digital image processing fundamentals. Prentice-Hall International Editions.
- [6] S. K. MITRA AND H. LI, "a new class of nonlinear filters for image enhancement," in Proc. IEEE int. Conf. Acoustics, Speech, Signal Processing, Toronto, Ont., Canada, may 14-17, 1991, pp. 2525-2528.
- [7] G. RAMPONI, N. STROBEL, S. K. MITRA, AND T. YU, "nonlinear unsharp masking methods for image contrast enhancement," J. Electron. Image, vol. 5, pp. 353-366, July 1996.
- [8] G. RAMPONI, "A cubic unsharp masking technique for contrast enhancement," Signal Process., vol. 67, pp. 211-222, June 1998.
- [9] Y. H. LEE AND S. Y. PARK, "A study of convex/concave edges and edge enhancing operators based on the laplacian," IEEE Trans. Circuits Syst., vol. 37, pp. 940-946, July 1990.

AUTHORS

First Author – Mrs. A. Padmapriya, Assistant Professor, Department of Computer Science & Engineering, Alagappa University, Karaikudi – 630 003, Email : mailtopadhu@yahoo.co.in

Mrs.A. Padmapriya is with Department of Computer Science and Engineering, Alagappa University, Karaikudi as Assistant professor. She is having 9 years of teaching experience and 5 years of research experience. She has published 5 research Papers in National/International Journals and presented 9 papers in International/National Conferences. She research area includes Computer Networks and Data Mining. She is a reviewer for IEEE Conferences and Symposion and various International Journals.

Second Author – S.Vigneshnarthi, M. Phil Scholar, Department of Computer Science & Engineering, Alagappa University, Karaikudi – 630 003, Email : narthi86@gmail.com

S.Vigneshnarthi received her B.sc degree in Computer Science from Alagappa University, Karaikudi, B.Ed degree Computer Science from Tamilnadu Teacher Education University, Chennai and M.sc in Computer Science from Alagappa University, Karaikudi. Currently she is pursuing her M.Phil in Computer Science under Alagappa University, Karaikudi.