

Image Processing Techniques Using LabVIEW

Anusha Nellutla

Assistant Professor, ECE Dept., IARE Hyderabad, Telangana, India

Abstract— Image processing is a methodology to perform some operations on an image, so as to get enhanced image or to extract some helpful information from it. It's a sort of signal process during which input is an image and output could also be image or characteristics/features related to that image. Nowadays, image process is among apace growing technologies. It forms core research area among engineering and technology disciplines too. We have innovatively developed and implemented image process based mostly application using labview ni vision module. using vision module image acquisition, color transformation, edge detection and morphological operation are performed.

I. INTRODUCTION

A. Laboratory Virtual Instrument engineering workbench (LabVIEW) is a system-design platform and development surroundings for a visible artificial language from National Instruments. LabVIEW is usually used for data acquisition, instrument control and industrial automation. LabVIEW simplifies hardware integration so you'll chop-chop acquire and visualize data sets from virtually any I/O device, whether by ni or a third-party. Combined with a graphical programming syntax that reduces programming time.

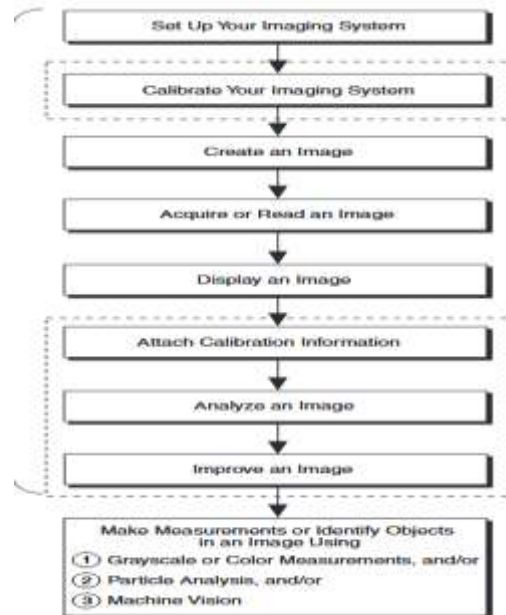
B. TYPES OF IMAGE PROCESSING AND TECHNIQUES

There are two forms of strategies used for image processing particularly, analogue and digital image process. Analogue image process is often used for the exhausting copies like printouts and pictures. Image analysts use numerous fundamentals of interpretation whereas victimisation these visual techniques. Digital image processing techniques facilitate in manipulation of the digital images by victimisation computers.

- The image processing principally deals with
- Image Acquisition
- Image Enhancement,
- Image Segmentation,
- Feature Extraction,
- Image Classification Etc.

C. Algorithm

General steps for designing a vision application



II. IMAGE PROCESSING TECHNIQUES

Image Acquisition

The first stage of any vision system is that the image acquisition stage. once the image has been obtained, varied ways of process are often applied to the image to perform the numerous completely different vision tasks needed nowadays. However, if the image has not been nonheritable satisfactorily then the meant tasks might not be realizable, even with the help of some type of image enhancement. Most image processing programs are designed to start out by loading an image from disk. Igor provides some facilities for acquiring images directly from a camera or from a video source.

Edge Detection

The definition of edges is critical native changes in intensity among an image. Edges tend to occur at the boundary between two completely different regions among an image. The goal of edge detection, then, is to supply a limning of a scene by tracing the boundaries /edges of the objects among a picture. Edges are often caused by two completely different situations; geometric events corresponding to an object or surface boundary, or non-geometric events corresponding to reflections of light and shadows among the image.

Canny Edge Detection

Canny edge detection algorithm is also known as the optimal

edge detector. Canny's intentions were to enhance the many edge detectors in the image.

- The first criterion should have low error rate and filter out unwanted information while the useful information preserve.
- The second criterion is to keep the lower variation as possible between the original image and the processed image.
- Third criterion removes multiple responses to an edge.

Based on these criteria, the canny edge detector first smoothes the image to eliminate noise. It then finds the image gradient to highlight regions with high spatial derivatives. The algorithm then tracks along these regions and suppresses any pixel that is not at the maximum using non-maximum suppression. The gradient array is now further reduced by hysteresis to remove streaking and thinning the edges.

Morphological Image Processing

When applied to vision, morphology (from the Greek word morph, meaning shape or form) refers to the alteration of an image using computer routines. A more-common use for the term is the changing of an image in graduated steps (similar to the morphing of an image of a baby to an adult), although this is a particular type of morphology.

Simple Morphology Theory

Morphological operations are generally neighborhood based (the new value for the pixel under inspection (PUI) is determined from the values of its neighboring pixels).The two main types of binary morphological operations are erosion and dilation (other morphological operations exist, but are generally combinations of the two).Morphological operations are very similar to filters, except the kernels used are dependent on the original value of the PUI.

Dilation

Similar to the biological condition when referring to the human iris, dilation refers to the spatial expansion of an object with the potential of increasing its size, filling holes and connecting neighboring objects. Consider an image with the following elements:

NW	N	NE
W	x	E
SW	S	SE

Simple dilation occurs when the following procedure is applied: if the PUI is 1, then it retains its value; if it is 0, it becomes the logical OR of its cardinal (North, South, East and West) neighbors. An example is shown below. Note that any

PUIs with an original value of 1 are retained, small holes are filled and boundaries are retained.

$$x_{New} = \begin{cases} 1 & x_{Old} = 1, \exists(N, E, S, W) = 1 \\ 0 & otherwise \end{cases}$$



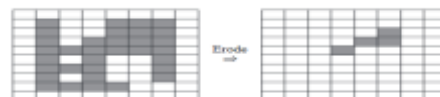
Erosion

Similar to a small island in the middle of a river during water flow, erosion wears away the edges of image features, decreasing its size, and potentially opening upholes in the object. Consider again the previous image with the following elements'

NW	N	NE
W	x	E
SW	S	SE

Simple erosion occurs when the following is applied: if the PUI is 0, then it retains its value; if it is 1 and all of its cardinal neighbors are also 1, then set the new pixel value to 1 (otherwise set it to 0).

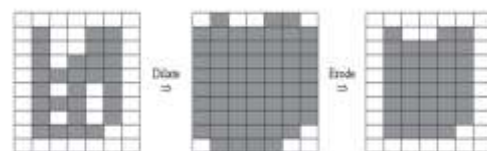
$$x_{New} = \begin{cases} 0 & x_{Old} = 0 \\ 1 & x_{Old} = 1 \text{ and } \wedge(N, E, S, W) = 1 \end{cases}$$



Closing

To close small holes and other background features in an image, all you need to do is first dilate the source image, and then erode it. Although performing two seemingly opposite morphs on the source image may suggest that it will result in the destination image being the same as the source, this is not the case. Original image data is lost on executing the first operation, and hence the original image cannot be recovered.

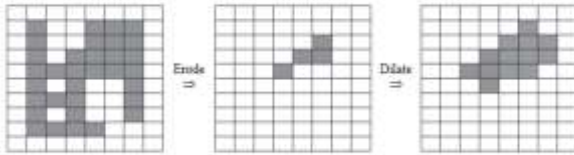
$$I_{Closed} = Erode\{Dilate(I_{Source})\}$$



Opening

Opening an image (expanding holes and other background features in an image) requires the opposite sequence to closing first erode the source image, and then dilate it.

$$I_{\text{Opened}} = \text{Dilate}\{\text{Erode}(I_{\text{Source}})\}$$



Colour Transformation

Color can be described by its red (R), green (G) and blue (B) coordinates (the well-known RGB system), or by some its linear transformation as XYZ, CMY, YUV, IQ, among others. The CIE adopted systems CIELAB and CIELUV, in which, to a good approximation, equal changes in the coordinates result in equal changes in perception of the color. Nevertheless, sometimes it is useful to describe the colors in an image by some type of cylindrical-like coordinate system, it means by its hue, saturation and some value representing brightness. If the RGB coordinates are in the interval from 0 to 1, each color can be represented by the point in the cube in the RGB space. Let us imagine the attitude of the cube, where the body diagonal linking "black" vertex and "white" vertex is vertical. Then the height of each point in the cube corresponds to the brightness of the color, the angle or azimuth corresponds to the hue and the relative distance from the vertical diagonal corresponds to the saturation of the color.

III. SOFTWARE

A. Dataflow Programming

The artificial language utilized in LabVIEW, conjointly brought up as G, may be a dataflow artificial language. Execution is set by the structure of a graphical diagram (the LV-source code) on that the software engineer connects completely different function-nodes by drawing wires. These wires propagate variables and any node will execute as shortly as all its computer file become offered. Since this can be the case for multiple nodes at the same time, G is inherently capable of parallel execution. Multi-processing and multi-threading hardware is mechanically exploited by the intrinsic hardware, that multiplexes multiple OS threads over the nodes prepared for execution.

About IMAQ Vision Module

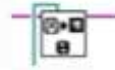
IMAQ Vision for LabVIEW—a a part of the Vision Development Module—is a library of LabVIEW VIs that you just will use to develop machine vision and scientific imaging applications. The Vision Development Module conjointly

includes constant imaging functions for LabWindows and different C development environments, moreover as ActiveX controls for Visual Basic. Vision Assistant, another Vision Development Module wares, permits you to image your application strategy quickly while not having to try and do any programming. to boot, metal offers Vision Builder AI: configurable machine vision package that you just will use to image, benchmark, and deploy applications



Vision Modules

Use the Image show management to show a picture on the LabVIEW front panel; you'll access the Image show management by right-clicking the front panel and choosing Vision.



Use the IMAQ browse File VI (Vision Utilities »Files) to open and skim knowledge from a file keep on your laptop into the image reference. you'll browse from image files keep in an exceedingly common place format—such as BMP, TIFF, JPEG, PNG, and AIPD—or a nonstandard format you specify. all told cases, the package mechanically converts the pixels it reads into the kind of image you pass in.

Use the IMAQ produce VI (Vision Utilities »Image Management) to form a picture reference. after you produce a picture, specify one among the subsequent image knowledge types:

IMAQ Replace Color Plane VI (Vision Utilities »Color Utilities), you can reassemble a color image from a set of three 8-bit or 16-bit images, where each image becomes one of the three primary components. 32-bit and 64-bit color images break down into their three primary components.



High pass filters emphasize details, such as edges, object boundaries, or cracks. These details represent sharp transitions in intensity value. You can define your own high pass filter with IMAQ Convolute or IMAQ Nth Order or use the IMAQ Edge Detection VI or IMAQ Canny Edge Detection VI (Image Processing »Filters). IMAQ Edge Detection allows you to find edges in an image using predefined edge detection kernels, such as the Sobel, Prewitt, and Roberts kernels.



Use the IMAQ Gray Morphology VI (Image Processing »Morphology) to perform Erosion, Dilation, Opening, Closing



Morphology—A group of VIs that perform morphological operations on an image. Some of these VIs perform basic morphological operations, such as dilation and erosion, on grayscale and binary images. Other VIs improve the quality of binary images by filling holes in particles, removing particles that touch the image border, removing small particles, and removing unwanted particles based on different shape characteristics of the particle. Another set of VIs in this sub palette separate touching particles, find the skeleton of particles, and detect circular particles.



Color Processing—A group of VIs that analyze and process color images. Use these VIs to compute the histogram of color images; apply lookup tables to color images; change the brightness, contrast, and gamma information associated with a color image; and threshold a color image. Some of these VIs also compare the color information in different images or different regions in an image using a color matching process.



Use the IMAQ Read Image and Vision Info VI (Vision Utilities» Files) to open an image file containing additional information, such as calibration information, template information for pattern matching, or overlay information. Performing Machine Vision Tasks, for information about pattern matching templates and overlays.



You also can use the IMAQ Get File Info VI (Vision Utilities» Files) to retrieve image properties—image size, pixel depth, recommended image type, and calibration units—without actually reading all the image data.



Use IMAQ AVI Open and IMAQ AVI Read Frame to open and read data from an AVI file stored on your computer into the image reference. IMAQ Vision automatically converts the pixels it reads into the type of image you pass in.



Use the IMAQ Array To Image VI (Vision Utilities» Pixel Manipulation) to convert a 2D array to an image. You also can use the IMAQ Image To Array VI (Vision

Utilities» Pixel Manipulation) to convert an image to a LabVIEW 2D array



You can use a color palette to display grayscale images by applying a color palette to the window. You can use the IMAQ Get Palette VI (Vision Utilities» Display) to obtain predefined color palettes. For example, if you need to display a binary image—image containing particle regions with pixel values of 1 and a background region with pixel values of 0—apply the predefined binary palette.



Color Utilities—A group of VIs that access data from color images. Use these VIs to extract different color planes from an image, replace the planes of a color image with new data, convert a color image to and from a 2D array, read and set pixel values in a color image, and convert pixel values from one color space to another.



High pass filters emphasize details, such as edges, object boundaries, or cracks. These details represent sharp transitions in intensity value. You can define your own high pass filter with IMAQ Convolve or IMAQ Nth Order or use the IMAQ Edge Detection VI or IMAQ Canny Edge Detection VI (Image Processing» Filters). IMAQ Edge Detection allows you to find edges in an image using predefined edge detection kernels, such as the Sobel, Prewitt, and Roberts kernels.



At the end of your application, dispose of each image that you created using the IMAQ Dispose VI (Vision Utilities» Image Management).

B. Input and Output Combinations

Image Analysis

The following connector pane applies only to VIs that analyze an image and therefore do not modify either the size or contents of the image. Examples of these types of operations include particle analysis and histogram calculations.

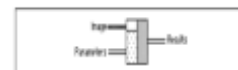
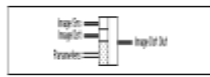


Image Processing

The following connector pane applies to VIs that process an image.



This connector is the most common type in NI Vision. The **Image Src** input receives the image to process. The **Image Dst** input can receive either another image or the original, depending on your goals. If two different images are connected to the two inputs, the original **Image Src** image is not modified. As shown in the following diagrams, if the **Image Dst** and **Image Src** inputs receive the same image, or if nothing is connected to **Image Dst**, the processed image is placed into the original image, and the original image data is lost.



The **Image Dst** input is the image that receives the processing results. Depending on the functionality of the VI, this image can be either the same or a different image type as that of the source image. The VI descriptions in the NI Vision for LabVIEW VI Reference Help include the type of image that can be connected to the Image inputs. The image connected to **Image Dst** is resized to the source image size.

IV. FINAL OUTPUT

Image Acquisition

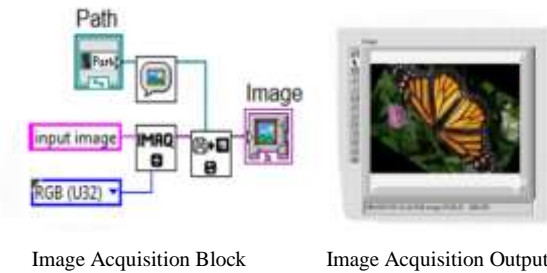
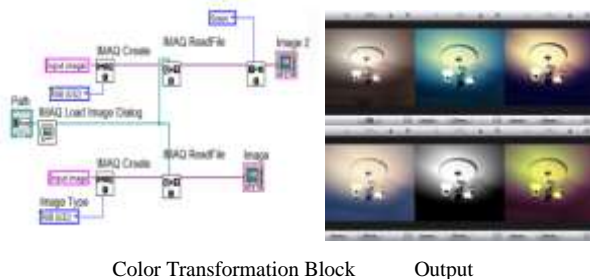


Image Acquisition Block

Image Acquisition Output

Color Transformation

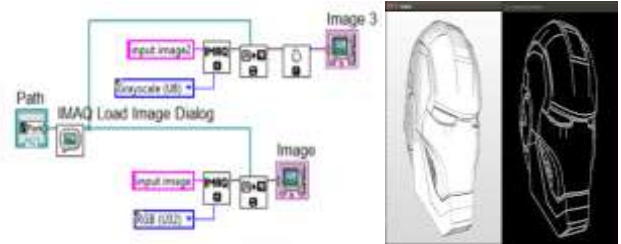


Color Transformation Block

Output

The above block is used to perform different color transformation operations like saturation, hue, HIS, etc on the loaded input image

Edge Detection

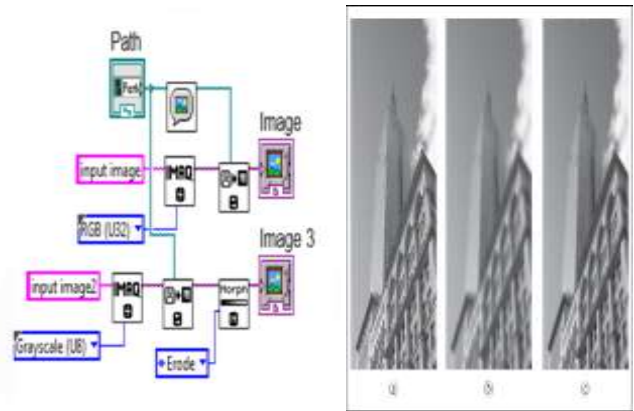


Edge Detection Block

Output

The above block is used to perform edge detection operations such as sobel edge detection, canny edge detection..etc on the loaded input image

Morphological Operation

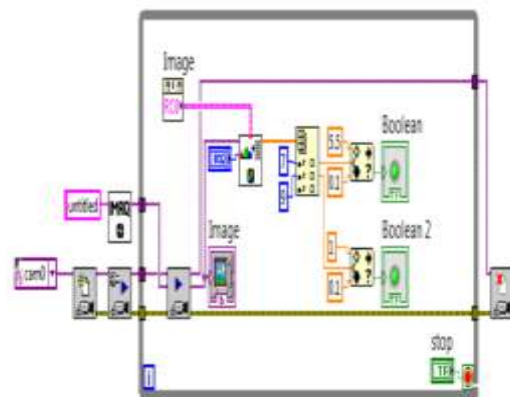


Morphology Block

Output

Block Description :The block is used to perform different morphological operations such as erosion, dilation, closing and opening.

Real Time Color Matching





Real Time Color Matching For Image Output

Block Description : This block is used to detect the color in real time i.e, initially a color is given as input in the block. If the camera detects the same color which is given as input then the led will be on.

V. CONCLUSION

The image transformation leave to grasp however the output image is generated from Associate in Nursing input image with utilization of timely and sorted operations are performed.

It is enforced by interconnecting all the operations and during a single frame by victimisation loop functions. This whole operation is used for a lot of operations. The versatility provided by the software system LabView and also the capability of IMAQ tool case increase the likelihood to boost the utilization of digital image process in any application space.

REFERENCES

- [1]. *IMAQ Vision Concepts Manual*, 2002, National Instruments
- [2]. *LabVIEW Advanced Programming Techniques*,
- [3]. Bitter, Mohiuddin, and Nawrocki, 2001, CRC Press LLC
- [4]. *LabVIEW Graphical Programming (2nd ed.)*,
- [5]. Gary W. Johnson, 1997, McGraw Hill
- [6]. *LabVIEW Power Programming*, Gary W. Johnson, 1998, McGraw Hill
- [7]. *National Instruments Developer Zone*, <http://zone.ni.com>
- [8]. *National Instruments LabVIEW Zone*,
- [9]. <http://www.ni.com/devzone/lvzone>
- [10]. *Shape Analysis and Classification*, Luciano da Fontoura Costa and Roberto
- [11]. Marcondes Cesar Jr., 2001, CRC Press LLC
- [12]. *The Engineering Handbook*, Richard C. Dorf, 2003, CRC Press LLC
- [13]. *The Image Processing Handbook (4th ed.)*, John C. Russ, 2002, CRC Press