An educational software package for digital image processing

Juan Manuel Ramírez[1], Pilar Gómez-Gil[2], Luis G. Guerrero[1], David Baez[1] & Ernesto Dorantes[1]

[1]*Departamento de Ingeniería Electrónica, Universidad de las Américas Puebla, México.*
[2]*Departamento de Sistemas Computacionales, Universidad de las Américas Puebla, México.*

Abstract

Due to the increasing number of applications of digital image processing in the last years, we have included a course on this topic in the list of courses offered by the Electrical Engineering Department. This course has been taught at graduate level, however, undergraduate students at senior level are allowed to take it as an optional course. An educational software package has been developed to be used as a complement of the theory revised in the lectures. This package was implemented in Visual Basic*, which is a visual programming language to create applications in a Windows* environment. The image to be processed can be generated using the editor included, or can be imported using images in a *bmp* format. The images consist of a 64×64 pixels array with 16 gray levels. This size was considered enough to perform, with educational purpose, most of the operations in the spatial and frequency domains in a reasonable amount of time. Operations available include manipulation in the space and frequency domains, such as 2-D convolution, gray level transformations, linear filtering, histogram equalization, morphological filtering, and others. Examples show how the package can be useful as a lecture aid and as a lab assistance tool. The image to be processed can be generated using the editor included, or can be imported using images in a bmp format. A graphical interface allows the user to activate available operations through a menu selection.
* Visual Basic and Windows of Microsoft, Inc.

1 Introduction

In the last several years there has been a strong development of software to be used in every field of engineering. In the specific case of digital image processing, there is a great variety of professional software, which can be extremely useful for industrial applications as well as for research purposes, but expensive for academic purposes. The educational software package presented in this article, has been developed to be used as a complement of the theory revised in the course Digital Image Processing, taught at the Electrical Engineering Department of *Universidad de las Américas*. This package was implemented in Visual Basic*, which is a visual programming language to create applications in a Windows environment.

2 General description of the software package

This program was conceived with the idea of providing students in this course, with a software tool which could be easy to install, learn, and use in any personal IBM compatible computer. Typically, a digital image in the different formats used by professional programs occupies a great amount of memory, and that was something to avoid if the student should store and print several samples to accomplish scholar assignments. In this package, the images consist of a 64×64 pixels array with 16 gray levels. This size was considered enough to perform, with an educational purpose, most of the operations in the spatial and frequency domains, without losing any signal processing concept. The image to be processed can be generated using the editor included, or can be imported using images in a *bmp* format.

   A graphical interface allows the user to activate different operations through a menu selection. The software starts by presenting a blank main screen with the different options described as follows:

FILE: Basic commands to handle files: open, close, save, exit.

EDITION: Examine buffer, Edit the selected image, create a new image, copy, paste. Access to the utility program to get a window from any bmp image.

SPACE: Processing in the space domain: Logic-arithmetic operations, gray level transformations, 2-D convolution.

FREQUENCY: Processing in the frequency domain: Fourier power spectrum, ideal linear filters: low-pass, high-pass, band-pass, and reject-band.

ENHANCEMENT: Histogram equalization, sharpening, pseudo color.

NON-LINEAR FILTERS: Median, homomorphic, and morphological filters. Erotion, dilation, opening, and closing operations needed to realize morphological filters.

2.1 Edition

Several ways to get a source image are provided. In the first case, an image editor to enter directly the value of each pixel is included. This editor is especially useful when the user wants to explore some operations with simple

geometrical forms. Furthermore, an image previously saved can be brought to the work area in order to edit some desired pixels. The option *examine buffer* can be used any time during the process in order to visualize the gray level map of any obtained image. The additional tool *IMAGEN_PDI* has been included in order to generate an image from any available BitMap. This option is self-explanatory when it is accessed.

2.2 Spatial domain

In this category we have included point operations which modify the gray scale of an image in a pixel-to-pixel basis, logic-arithmetic operations between two or more images, and two-dimensional convolution. In the last case, the user is allowed to define the size and coefficients of the convolution mask.

2.2.1 Logic-arithmetic operations
In this category, arithmetic and logic operations involving two or more images can be performed. The result is automatically adjusted in such a way that the dynamic range of the gray levels is not exceeded.

2.2.2 Gray scale transformations
In the same category of spatial domain, a gray-level transformation represented by Eq. (1) can be performed.

$$s = T(r) \tag{1}$$

r : gray level in the original image
s : gray level in the transformed image.

This is a pixel to pixel mapping, in which the desired transformation is specified graphically. The images in Figure 1 are the result of applying an inversion in the gray levels, which has the effect of turning the input image in its corresponding negative. The third image in the figure was obtained after a binarization.

2.2.3 Two-dimensional convolution
The last operation included in this category is a two dimensional convolution defined by Eq. (2). In a linear system, the output is given by the convolution of the input function with the unit impulse response [1,2].

$$g(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) h(x - m, y - n) \ . \tag{2}$$

In digital image processing, the impulse response is defined as a convolution mask or kernel with a size typically much less than the size of the input image. In this package the size of the convolution mask ranks from 3×3 to 11×11.
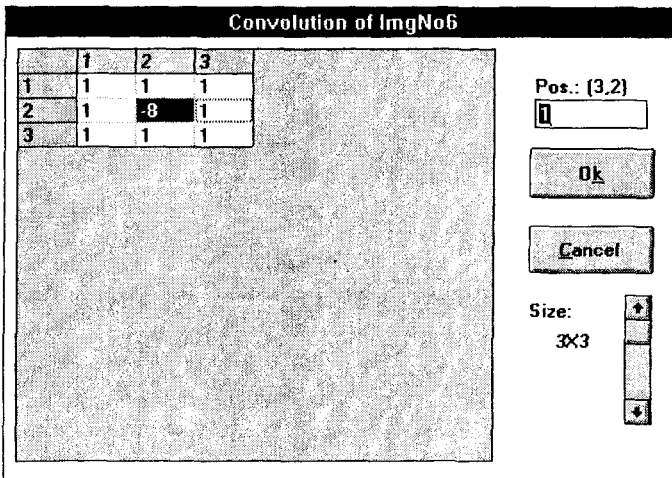
Figure 1: Gray level transformations.

Figure 2: Convolution; definition of the kernel.

In Figure 2, the convolution kernel of the Laplacian operator is shown with the coefficients entered in the corresponding position. As an example, the Laplacian operator is used on the image 'circles' to perform edge detection, with the result enhanced through a standard binarization. The results obtained are shown in Figure 3.
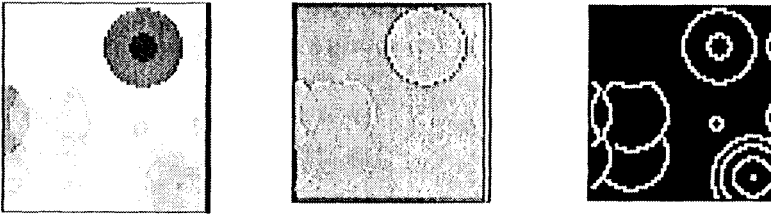
Figure 3: Original image 'circles', Laplacian of 'circles', and binarization of Laplacian of 'circles'.

2.3 Frequency domain

In this category, the operations corresponding to ideal linear filtering are included. The mathematical tool which links the space and frequency domains is the two dimensional Fourier Transform.

2.3.4 Discrete Fourier Transform

Since a digital image is represented by a two-dimensional sampled function, the Discrete Fourier Transform (DFT) is used. The DFT is defined as [2,3]:

$$F(u,v) = \frac{1}{N^2} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \, e^{-\frac{j2\pi}{N}(ux+vy)} , \qquad (3)$$

where (x,y) and (u,v) are coordinates in the spatial and frequency domains, respectively. Figure 4 shows an image $f(x,y)$ in the spatial domain, and its corresponding magnitude spectrum $|F(u,v)|$.
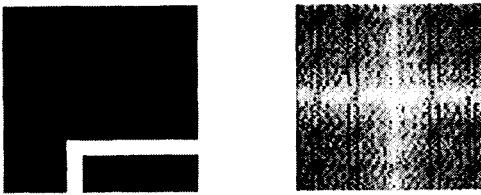

Figure 4: Image 'angle' in the spatial domain and its Fourier Transform.

2.3.5 Linear filtering

A linear filter consists basically of a selective discrimination of frequency spectral components, thus, an ideal low-pass filter keeps only the Fourier coefficients inside a circle of some specific radius, multiplying by zero the remaining ones. The ideal high pass filter is defined just in the opposite way, i.e., rejecting the frequency components inside a circle and passing the rest; the ratio of the circle defines the cutoff frequency of the filter. Two circles forming a ring

are required in order to implement the band pass and band reject ideal filters, passing or rejecting spectral components as required. Figure 5 shows the results obtained after applying ideal low pass filters with two different cutoff frequencies.

Figure 5: Low-pass filtering of the image `house`.

2.4 Enhancement

The goal of enhancement techniques is to process a given image so that the result is more suitable than the original image for a specific application. Under this category we have the operation of histogram equalization, enhancement, and pseudo-color.

2.4.1 Histogram equalization

An histogram provides statistical information about the use of the range of gray levels in a digital image, e.g., the histogram of a dark image is concentrated in the lower gray levels. Although visual evaluation of image quality is a subjective process, it is apparent that a human perceives better the definition of an image when the range of gray levels is completely covered, i.e., when the corresponding histogram tends to be flat. The process of applying a gray level transformation to some image in order to optimize the use of the dynamic range is called histogram equalization [1,2]. This is achieved when the cumulative distribution function, given by Eq. (4), is used as the transformation function.

$$s = T(r) = \int_0^r p_r(w)\, dw \ . \tag{4}$$

Figure 6 shows the image `lena-x` and its corresponding histogram. The pdf is obtained directly from the original histogram. A simple operation is applied in order to obtain the cdf which is used as the gray level transformation function. This function can be visualized in a table form. The new histogram as well as the equalized image is shown in Figure 7.
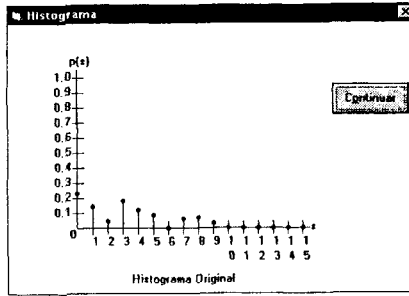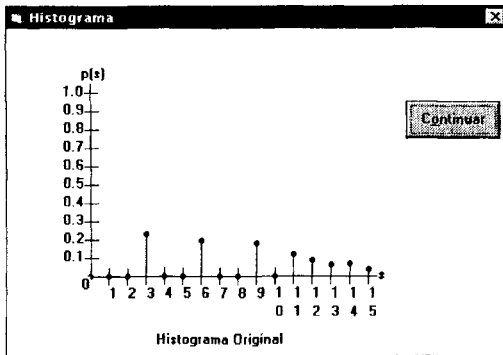
Figure 6: Histogram of the image `lena-x`.

Figure 7: Histogram equalization of the image `lena-x`.

2.4.2 Pseudo color

The motivation for using color in image processing is provided by the fact that the human eye can discern better color shade and intensities than gray level in a monochrome image [4]. This is especially important in those applications in which a visual examination is crucial, as is the case with medical images and other type of images. The software package has an option to assign a color to each gray level by entering the corresponding trichromatic coefficients. Two default color assignments are included in order to provide an initial choice to the user, which can be modified as needed.

2.5 Non-linear operations

Many applications of digital image processing techniques are based on non-linear operations. Under this category we have in the package the use of morphological, median, and homomorphic filters.

2.5.1 Morphological filters

Morphological filters are implemented by defining a structurant element B which operates on the required image A. This structurant element is defined in a similar

way to the convolution mask previously described, however, the operations between pixels are performed according to the definitions of mathematical morphology [4]. Erotion and dilation are defined by Eqs. (5) and (6), respectively:

$$C = A \ominus B = \bigcap_{b \in B} A_b \tag{5}$$

$$C = A \oplus B = \bigcup_{b \in B} A_b . \tag{6}$$

In these definitions, A_b denotes the translation of the image A by the sub-element b. The techniques included in the package are: erosion, dilation, opening, and closing filters. Figure 8 shows the results obtained when a binary image is filtered with a structurant element given by a 3×3 and a 5×5 square respectively:

Figure 8: A noisy image and the results after applying morphological opening filter.

2.5.2 Median filtering

In a median filter, the value of each pixel is replaced by the median of the gray levels in a neighborhood of that pixel. This method is particularly effective to remove impulsive noise, and when edge sharpness is important to preserve. The size of the region in which the statistical analysis is performed, is defined by the user.

3 Conclusions

An educational software created with the idea to provide a support in the teaching of digital image processing has been presented. This package was implemented in Visual Basic* to be used in a Windows* environment, which makes it easy to learn and manipulate by the students from the first attempt. The size of the program as well as the images processed with it, make this software easily handled by the students. The theory involved in each topic of the course has been reinforced with the help of this software package. This package is available on request to educators as a shareware package.