Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2014) 22: 166 – 175
© TÜBİTAK
doi:10.3906/elk-1203-129

A web laboratory using MATLAB Builder NE for computer image processing

Sezgin KAÇAR[1], Cüneyt BAYILMIŞ[2], Özdemir ÇETİN[1,*]
[1]Department of Electrical-Electronic Engineering, Faculty of Technology, Sakarya University, Sakarya, Turkey
[2]Department of Computer Engineering, Faculty of Technology, Sakarya University, Sakarya, Turkey

Received: 29.03.2012 • **Accepted:** 26.07.2012 • **Published Online:** 20.12.2013 • **Printed:** 20.01.2014

Abstract: Many various instruments are needed for computer image processing. These instruments are used for image restoration, analysis, etc. This paper presents a novel web laboratory based on MATLAB Builder NE with Web Figure, called ImageWeb, to facilitate computer image processing without needed technical information for users. The developed ImageWeb provides fundamental principles of image processing, including basic, compression, edge detection, filtering, optimization, and histogram processes, and so on. In addition, the usage of MATLAB Builder NE with MATLAB Web Figure provides high flexibility with visualization means such as zoom, rotate, and 3-dimensional view. Users can easily access ImageWeb using only a web browser, without requiring any special programs over the Internet and without time and location restraints.

Key words: Web laboratory, computer image processing, MATLAB Builder NE, MATLAB Web Figure

1. Introduction

Nowadays, considering the wide use of digital images, the importance and popularity of computer image processing has increased for the manipulation and rearrangement of digital images. To this end, computer image processing requires the usage of many different instruments, such as those for rotating, resizing, cropping, zoom, edge detection, compression, filtering, histogram processing, and optimization. Computer image processing tools present the usage of these instruments without any requirement of technical information and background. However, computer image processing is used in many application areas (medicine, education, engineering, etc.). Therefore, computer image processing tools play a very important role in assisting the understanding of digital image processing techniques.

There are many works, both academic and commercial, related to computer image processing tools in the literature. Ayala et al. [1] developed a Windows-based tool for teaching image processing techniques. This tool requires extra downloading and installation. CVIPtools, an educational website for computer image processing, was presented in [2]. An online image analysis tool for science education was developed in [3]. The web-based educational tools are an easy and fast solution for training and learning. Another website for online image processing was presented in [4]. This website includes some basic image processing tools, such as stereo vision, segmentation, characteristic lines, textures and gradients, color separations, and histogram manipulations, for users to experiment with on their own images. Moreover, a Java-based online laboratory for image and signal processing was designed in [5]. Mahmoudi et al. designed a web-based interactive 2D/3D medical image processing software by combining JavaScript with AJAX technology [6]. One of the popular

[*]Correspondence: ocetin@sakarya.edu.tr

online image processing tools is the Pixlr tool. This tool presents much ability for users to manipulate their special images over the Internet, like Photoshop. The Pixlr tool is a commercial tool and its aim is not education, unlike in this work [7].

Another commercial online tool is the Online Image Resizer (OIR). Users can resize and crop their images, and apply Text-Over-Image or basic effects, such as grayscale or sepia, using OIR [8]. Discussing further tools is possible, like those in [9,10]; however, they only pertain to image refinement, not education.

On the other hand, MATLAB Builder NE is a new toolbox for programmers who want to benefit from the .Net and MATLAB platforms together. This has a very simple and flexible usage that provides for using of advanced analysis and visualization features of MATLAB in a .Net application. More specifically, MATLAB Web Figure is very useful as a part of MATLAB Builder NE for graphical presentation in web-based .Net applications. For that reason, MATLAB Builder NE and MATLAB Web Figure can be implemented in various fields. For example, Kaçar et al. used MATLAB Builder NE and MATLAB Web Figure to remove the need for monitoring wireless sensor networks [11], and Bayilmis [12] used them in a web-based educational interface about digital communication. Moreover, the tools were useful for web-based biomedical applications [13,14]. In another work, a Windows-based .Net application for system analysis was realized with MATLAB Builder NE [15].

In this paper, a web laboratory for image processing, named ImageWeb, is aimed to be designed as a web interface/site with MATLAB Builder NE and MATLAB Web Figure based on the ASP.Net platform. Different from other web-based image processing tools, ImageWeb, presented in this paper, enables the use of the high analysis and graphical features of the MATLAB platform for users without any license payment. In addition, ImageWeb has more actual, flexible, and dynamic structures than other tools, because .Net technology is used for the design of ImageWeb. In Section 2, the features of the web laboratory are presented and the theoretical basics of the work are described in Section 3. Section 4 shows the usage of the designed web interface. In Section 5, the advantages and disadvantages of the web laboratory are mentioned.

2. Web laboratory (ImageWeb) features

In order to facilitate the learning and teaching of computer image processing, a web laboratory should have several features, such as using freeware, not requiring extra program installation, being user friendly, not requiring programming skills and technical information, and so on. Moreover, users can employ it without time and location restraints. However, an image processing tool must present fundamental instruments and features including basic processes (rotating, resizing, etc.), filtering, and histogram and compression processes. The developed ImageWeb has the features mentioned above. Figure 1 shows the basic menus of the developed ImageWeb and their subprocesses.

3. Computational basics of ImageWeb

3.1. Basic processes

Fundamentally, an image can be digitally shown as a matrix. Hence, all of the processes for images are matrix processes. The most popular basic processes are zooming and shrinking, which are associated with sampling and quantization, respectively. Zooming consists of 2 parts: creating a new pixel map and the assignment of gray levels for newly mapped pixels. The shrinking process is also done in a similar way. In the shrinking process, row-column deletion is achieved for creating the new pixel map, differently from zooming. Another basic process of image processing is rotation. A common definition can be given for the transforms as below:

$$\begin{array}{l} \dot{x} = a_0 x + a_1 y + a_2 \\ \dot{y} = b_0 x + b_1 y + b_2 \end{array}, \tag{1}$$

where $(\dot{x}, \dot{y})$ is mapped as (x, y), and a_0, a_1, a_2, b_0, b_1, and b_2 are real-valued coefficients. This definition can also be shown in matrix form:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ 1 \end{bmatrix} = \begin{bmatrix} a_0 & a_1 & a_1 \\ b_0 & b_1 & b_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}. \tag{2}$$

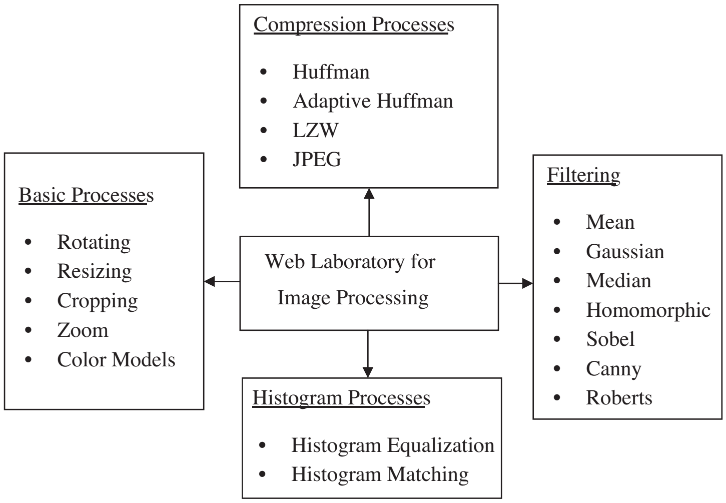


Figure 1. Features of the developed ImageWeb.

3.2. Compression processes

Image compression is reducing the size of a graphic file while keeping the quality. This process allows the storage of more images in memory. It also reduces the sharing time over the Internet.

In the image compression process, data redundancy is an essential issue. Defining it in a mathematical form, for instance, n_1 and n_2 indicate the number of information-carrying units in 2 sets. These sets also have the same information. Hence, this is called the relative data redundancy R_D of the first data set:

$$R_D = 1 - \frac{1}{C_R}, \tag{3}$$

where C_R is called the compression ratio:

$$C_R = \frac{n_1}{n_2}. \tag{4}$$

Here, if $R_D = 0$, then the first representation of the information does not have any redundant data [16].

Basically, image compression can be performed in 2 main ways: lossless compression and lossy compression, such as RAR-PNG or JPEG, respectively. Lossless compression makes it possible to have an exact original image from the compressed image. When it is important that the compressed image and the original image be the same, lossless compression is useful. Opposite to lossless compression, in lossy compression, the compressed image may be different from the original image, but they are almost same. A basic image compression process can be given like that in Figure 2.

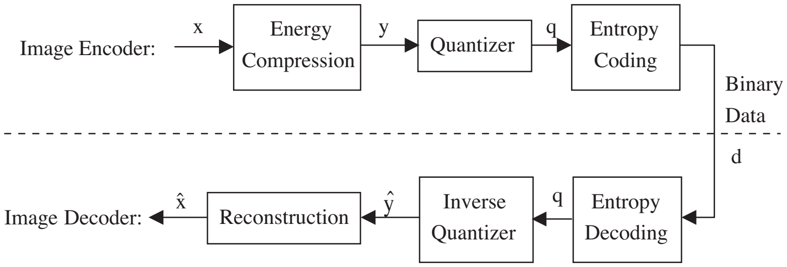


Figure 2. The basic block diagram of an image coding system [17].

3.3. Filtering processes

Image enhancement approaches consist of 2 main categories: the spatial domain and the frequency domain. In the spatial domain, the process is based on the direct manipulation of pixels in an image. Distinctively, in the frequency domain, it is based on modifying the Fourier transform of an image. The most popular area of usage for image enhancement applications is filtering processing, such as edge detection, smoothing, or sharpening. The filtering process offers the ability to enhance an image that is noisy, blurred, etc.

As mentioned above, in the spatial domain, pixels of the image are manipulated by a filter mask. This can be defined for an image that has a M × N dimension by the following expression:

$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t), \tag{5}$$

where a = (m − 1) / 2 and b = (n − 1) / 2. m and n are the filter dimensions. This expression is called convolution; therefore, the mask is a convolution mask. If 2 masks that are on the x and y directions are used for finding an object's edge in an image, it is called Sobel edge detection. An understanding of these processes, and especially a mathematical background, is difficult for students who are new to image processing. The gradient off for a function f(x,y) is defined as:

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}, \tag{6}$$

and its magnitude is:

$$\nabla f = \left[G_x^2 + G_y^2\right]^{1/2} = \left[\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2\right] \tag{7}$$

To simplify this equation, an equivalent can be written as below:

$$\nabla f \approx |G_x| + |G_y| \tag{8}$$

Opposite to the spatial domain, the frequency domain works on the Fourier transforms of the image. A block diagram of the basic steps for filtering in the frequency domain is shown in Figure 3.

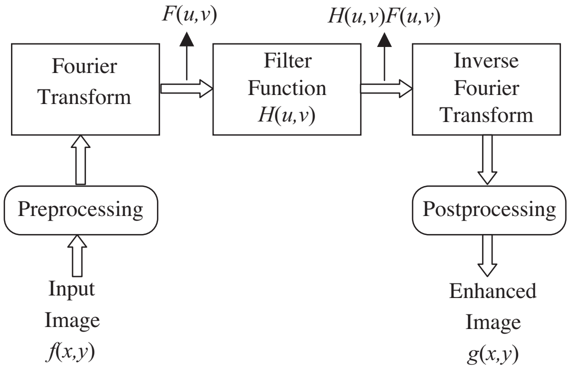


Figure 3. Basic steps for filtering in the frequency domain [16].

A general illustration for the Fourier transform of an image can be given by the following expression:

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(ux/M+vy/N)}. \tag{9}$$

High-frequency components in the Fourier transform of an image give edges and significant variations such as noise in the image. Therefore, a high-pass filter must be used for finding the edges of objects in an image. The basic filtering definition in the frequency domain is given by:

$$G(u,v) = H(u,v)F(u,v), \tag{10}$$

where G and H are the filtered image and filter function, respectively.

3.4. Histogram processes

A histogram of an image is a presentation of the frequency components of the image [18]. In other words, the histogram of a gray-level image offers the distribution of gray levels in the image. The histogram of an image with a gray level can be given by the following expression:

$$p(r_k) = \frac{n_k}{n}, \tag{11}$$

where n_k is number of pixels with gray value r_k and n is total number of pixels in the image.

Commonly used histogram processes are histogram matching and histogram equalization. These techniques are commonly used to enhance the image's appearance, such as making it lighter or darker. Histogram equalization tries to enhance the appearance of the image. It can be described with the formula seen below:

$$p_{in}(r_k) = \frac{n_k}{n} \qquad 0 \le r_k \le 1 \qquad 0 \le k \le L-1 \quad , \tag{12}$$

where L and n_k are the total number of gray levels and the number of pixels with gray value r_k, respectively.

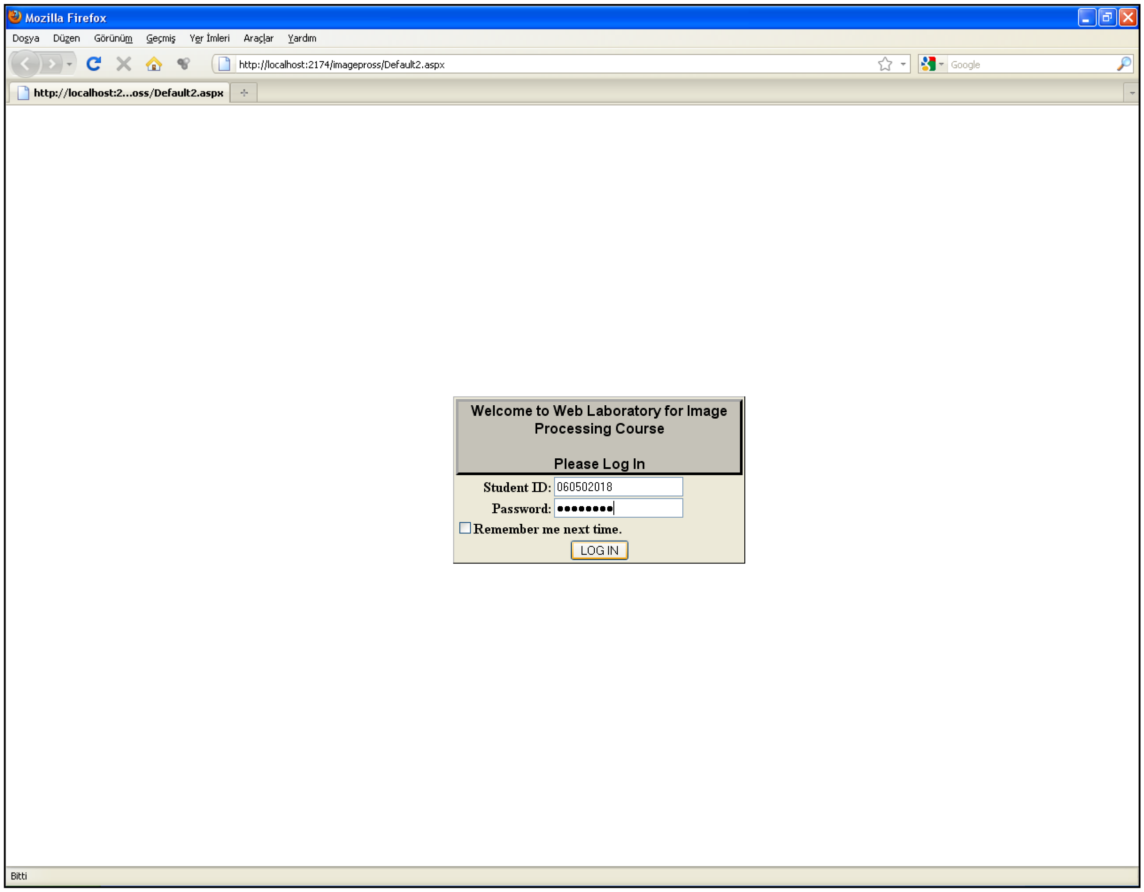
$$s_k = T(r_k) = \sum_{j=0}^{k} p_{in}(r_j) \qquad 0 \le k \le L-1 \tag{13}$$

Here, $T(r_k)$ is the cumulative distribution function of $p_{in}(r)$.

4. Presentation of designed ImageWeb

A web laboratory for an image processing course (ImageWeb) has been designed as a web interface/on-line tool on the ASP.Net platform as an actual technology. In addition, a user-friendly structure and high visibility are provided with ImageWeb. In the background of ImageWeb, MATLAB codes work for image processing. The connection between MATLAB and ASP.Net is provided by MATLAB Builder NE and MATLAB Web Figure Control. MATLAB Builder NE is a tool for deploying MATLAB codes to use them with CLS-compliant languages and MATLAB Web Figure Control is a part of it for using MATLAB figures on web pages [19]. For an ASP.Net application with MATLAB Builder NE and MATLAB Web Figure, MATLAB functions should be created first and compiled with MATLAB Builder NE second. They would then be .Net components and designers could use them for .Net applications. This web interface has been generated by this method. For all of the image processing techniques, a MATLAB function that is appropriate for using MATLAB Web Figure has been produced and all of the functions have been deployed for use in ASP.Net.

To access ImageWeb, users must have an ID and password and enter them correctly, as in Figure 4. This is designed so that only the students of the course can use the web interface.


Mozilla Firefox
Dosya
Düzen
Görünüm
Geçmiş
Yer İmleri
Araçlar
Yardım
http://localhost:2174/imagepross/Default2.aspx
Google
http://localhost:2...oss/Default2.aspx
Bitti

Welcome to Web Laboratory for Image Processing Course

Please Log In

Student ID: 060502018

Password: ●●●●●●●●●

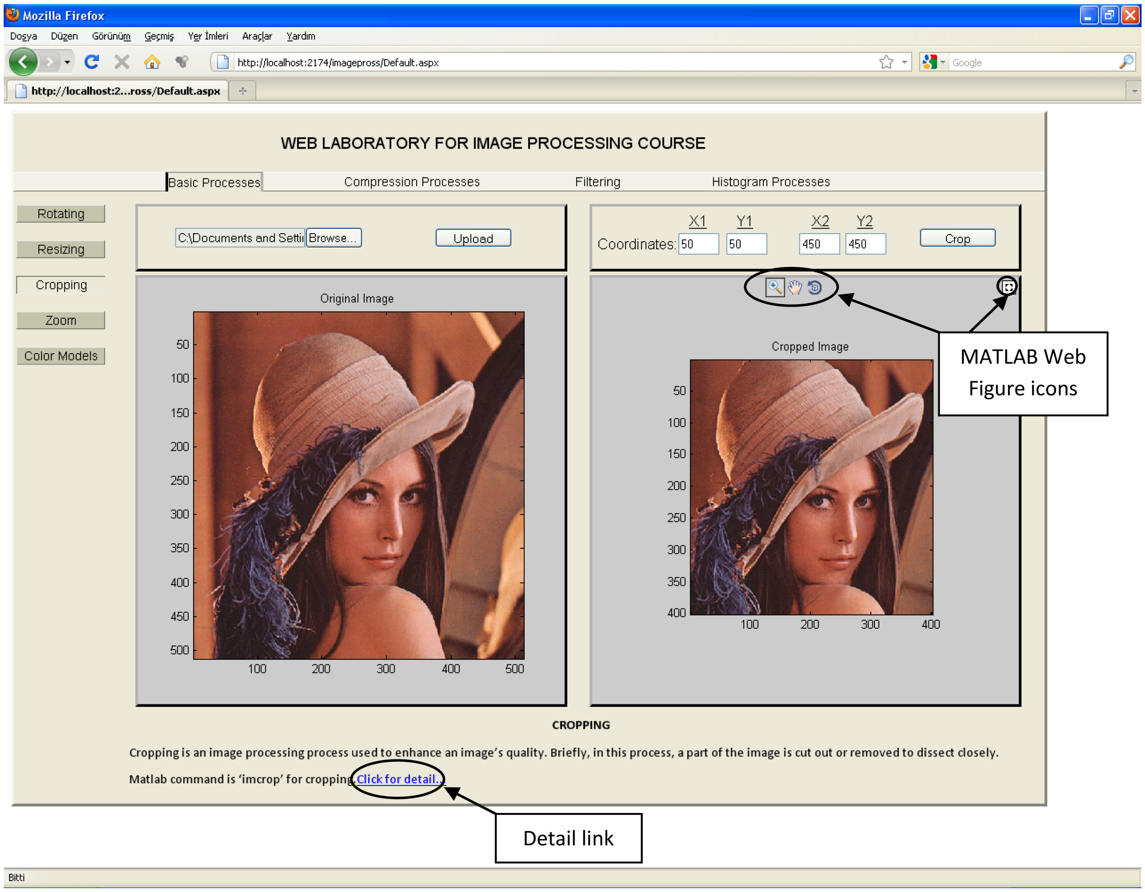
□ Remember me next time.

LOG IN

Figure 4. Login page of the developed ImageWeb.

After logging in, the users access ImageWeb and they can perform the processes selected from the menus. When a menu is selected, subprocesses appear on the left side of the web interface and the user should select one of them. After selection, for all of the processes, an image must be uploaded to process. The uploaded image is shown as the original image. Next, the selected process is performed. For example, a cropping process is seen in Figure 5.

In Figure 5, it is seen that the basic processes menu is selected from the menu bar and cropping is selected from the left menu. If a user wants to get detailed information about the process, he/she should click the link at the bottom of the web interface. This link is available for all processes in ImageWeb. An image named 'Lena' has been uploaded to interface as the original image in Figure 5. This is shown in the left frame and the right frame includes the processed image, named 'cropped image'. To perform cropping in Figure 5, the coordinates are written in the textboxes over the cropped image first, and the 'crop' button is clicked second. Finally, the cropped image is shown in the right frame.



WEB LABORATORY FOR IMAGE PROCESSING COURSE

Coordinates:

X1	Y1	X2	Y2
50	50	450	450

CROPPING

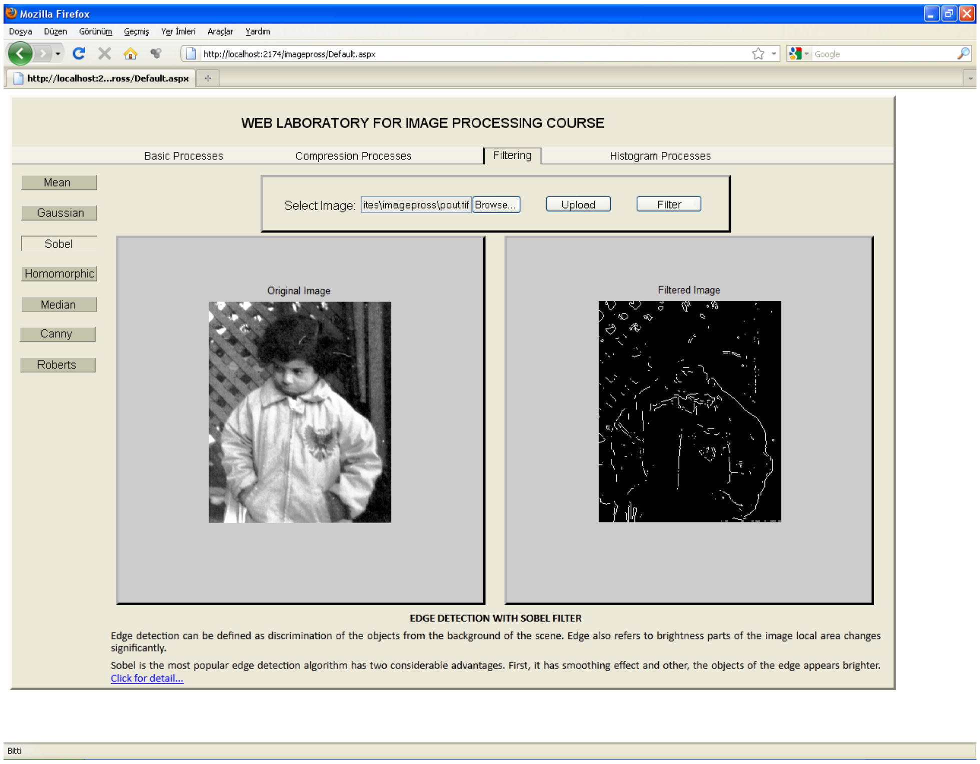
Cropping is an image processing process used to enhance an image's quality. Briefly, in this process, a part of the image is cut out or removed to dissect closely.

Matlab command is 'imcrop' for cropping Click for detail..

Figure 5. Cropping process under the basic processes menu in ImageWeb.

Four icons exist on the top of the images when the mouse cursor is over them. The icons are provided to allow the use of the MATLAB Web Figure features. The 1st icon on the left is 'zoom', the 2nd is 'moving', the 3rd is '3D-rotating', and the last returns the image to its original dimensions.

In Figure 6, edge detection with the Sobel filter and, in Figure 7, a histogram equalization process have been performed as other examples. The related menus are selected from the menu bar and the desired processes are clicked on from the subprocesses on the left side of the web interface. At the end of the processes, the processed images exist on the right frame of the web interface, as in Figures 6 and 7.

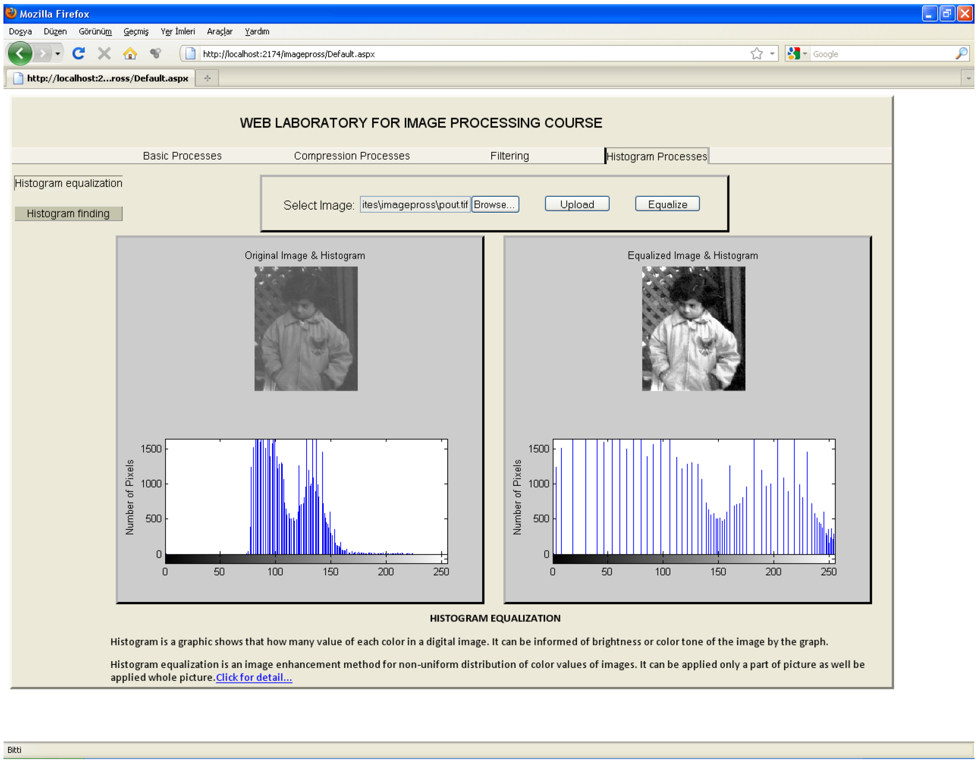

Mozilla Firefox
Dosya Düzen Görünüm Geçmiş Yer İmleri Araçlar Yardım
http://localhost:2174/imagepross/Default.aspx
Google
http://localhost:2...ross/Default.aspx
WEB LABORATORY FOR IMAGE PROCESSING COURSE
Basic Processes
Compression Processes
Filtering
Histogram Processes
Mean
Gaussian
Sobel
Homomorphic
Median
Canny
Roberts
Select Image: ites\imagepross\pout.tif
Browse...
Upload
Filter
Original Image
Filtered Image
Bitti

EDGE DETECTION WITH SOBEL FILTER

Edge detection can be defined as discrimination of the objects from the background of the scene. Edge also refers to brightness parts of the image local area changes significantly.

Sobel is the most popular edge detection algorithm has two considerable advantages. First, it has smoothing effect and other, the objects of the edge appears brighter. Click for detail...

Figure 6. Edge detection with the Sobel filter under the filtering menu in ImageWeb.


Mozilla Firefox
Dosya Düzen Görünüm Geçmiş Yer İmleri Araçlar Yardım
http://localhost:2174/imagepross/Default.aspx
Google
http://localhost:2...ross/Default.aspx
WEB LABORATORY FOR IMAGE PROCESSING COURSE
Basic Processes
Compression Processes
Filtering
Histogram Processes
Histogram equalization
Histogram finding
Select Image: ites\imagepross\pout.tif
Browse...
Upload
Equalize
Original Image & Histogram
Equalized Image & Histogram
Number of Pixels
1500
1000
500
0
0
50
100
150
200
250
Number of Pixels
1500
1000
500
0
0
50
100
150
200
250
Bitti

HISTOGRAM EQUALIZATION

Histogram is a graphic shows that how many value of each color in a digital image. It can be informed of brightness or color tone of the image by the graph.

Histogram equalization is an image enhancement method for non-uniform distribution of color values of images. It can be applied only a part of picture as well be applied whole picture.Click for detail...

Figure 7. Histogram equalization process under the histogram processes menu in ImageWeb.

5. Conclusions

In this paper, a new web laboratory has been designed for a basic computer image processing course. The web laboratory of the image processing course, called ImageWeb, includes general image processes that are grouped as basic processes, compression processes, and filtering and histogram processes. A number of subprocesses can be realized under these groups. All of the processes have been coded with MATLAB and deployed by MATLAB Builder NE. The deployed processes have then been used with ASP.Net while ImageWeb is used for design. The presented work is an innovative application using actual tools and platforms; however, there have been sample works performed previously. With these used tools and platforms, ImageWeb provides some advantages, as follows:

- MATLAB installation is not required (requires only free MATLAB Compiler Runtime)
- User friendly
- High visibility
- Flexibility
- Rapid design

In contrast to the advantages above, when the website gets a request, it responds a little late because of the MATLAB functions running in the background of the website.

In future works, new processes and a recording feature may be added to ImageWeb, and video processing may be performed with a different web laboratory.

References

[1] M. Ayala, M. Adjouadi, M. Cabrerizo, A. Barreto, “A windows-based interface for teaching image processing”, Computer Applications in Engineering Education, Vol. 18, pp. 213–224, 2010.

[2] A. Mani, R. Swamisai, S.E. Umbaugh, “Web site for computer imaging education”, Computer Applications in Engineering Education, Vol. 8, pp. 61–71, 2000.

[3] L. Raeside, B. Busschots, S. Waddington, J.G. Keating, “An online image analysis tool for science education”, Computer & Education, Vol. 50, pp. 547–558, 2008.

[4] University of Bremen Institute of Theoretical Physics, Online-Image Calculations, available at http://axon.physik.uni-bremen.de/online_ calc/, last accessed 25.12.2011.

[5] M. Yasin, L.J. Karam, A. Spanias, “On-line laboratories for image and two-dimensional signal processing”, 33rd ASEE/IEEE Frontiers in Education Conference, Session T3E, pp. 19–22, 2003.

[6] S.E. Mahmoudi, A. Akhondi-Asl, R. Rahmani, S. Faghih-Roohi, V. Taimouri, A. Sabouri, H. Soltanian-Zadeh, “Web-based interactive 2D/3D medical image processing and visualization software”, Computer Methods and Programs in Biomedicine, Vol. 98, pp. 172–182, 2010.

[7] Pixlr Photo Editing Services, Photo Editor Online – Pixlr.com Edit Image, available at http://pixlr.com/, last accessed 28.01.2012.

[8] GA Soft, Online Image Resizer, available at http://www.resizeimage.4u2ges.com/index.asp, last accessed 31.01.2012.

[9] Snipshot, Edit Pictures Online, available at http://snipshot.com, last accessed 08.02.2012.

[10] Picnik – Photo Editing the Easy Way, available at http://www.picnik.com, last accessed 05.02.2012.

[11] S. Kacar, C. Bayilmis, I. Cankaya, M. Cakiroglu, “Design of ASP.NET based WEB interface with MATLAB Builder NE and MATLAB Web Figure for wireless sensor networks”, e-Journal of New World Science Academy, Vol. 4, pp. 360–370, 2009.

[12] C. Bayilmis, “Development of a Web-based educational interface using MATLAB Builder NE with Web Figure for digital modulation techniques”, Computer Applications in Engineering Education, Vol. 20, pp. 604–610, 2010.

[13] E. Güney, Z. Ekşi, M. Çakıroğlu, “WebECG: a novel ECG simulator based on MATLAB Web Figure”, Advances in Engineering Software, Vol. 45, pp. 167–174, 2011.

[14] İ. Kırbaş, C. Bayılmış, “HealthFace: A web-based remote monitoring interface for medical healthcare systems based on wireless body area sensor network”, Turkish Journal of Electrical Engineering & Computer Sciences, Vol. 20, pp. 629–638, 2011.

[15] S. Kaçar, İ. Çankaya, “Design of .NET based interface for non-linear systems analysis in the frequency domain using Volterra series method”, Dokuz Eylül University Engineering Faculty Journal of Engineering Sciences, Vol. 12, pp. 87–102, 2010 (in Turkish).

[16] R.C. Gonzalez, R.E. Woods, Digital Image Processing, Upper Saddle River, NJ, USA, Prentice Hall, 2002.

[17] Connexions, A Basic Image Compression Example, available at http://cnx.org/content/m11086/latest/, last accessed 10.01.2011.

[18] O. Cetin, F. Akar, A.T. Ozcerit, M. Cakiroglu, C. Bayilmis, “A blind steganography method based on histograms on video files”, The Imaging Science Journal, Vol. 60, pp. 75–82, 2011.

[19] MathWorks, MATLAB Builder NE - Documentation, available at http://www.mathworks.com/help/toolbox/dotnetbuilder/dotnetbuilder_product_page.html, last accessed 15.02.2012.