

# Digital Image Filtering with LabVIEW

Liljana Docheva<sup>1</sup>

**Abstract** – Digital filters are indispensable tools for image processing. In this article an implementation of some local operators using Labview is considered. The local operators are used to perform image filtering. The results of the noise reducing and sharpening of the image are presented. Labview block diagram of different two-dimensional convolution kernels implementation that are applied to the test image is given. For the filter effect illustration the histograms of the resulting images are presented.

**Keywords** – image processing, local operators, digital filters, convolution kernel, convolution matrix, Labview.

## I. INTRODUCTION

Image Filtering is an important operation for improving the image quality. Thus enables further image processing. For example in the cases of neural network image processing, it is desirable to filter the input image before neural network application to improve the recognition capability of the neural network. After processing the image have to present to the network. There are several ways to do this. Extracting features from the image (e.g. edges) is commonly used approach. These features present to the neural network inputs. On this way incorporating prior knowledge before neural network training is made. This leads to neural network learning speed increasing, but only if extracted features are correctly selected. One of the relevant features for image recognition are the edges. Image filtering can be used before edge detection to obtain reducing of false edges.

In this article is introduced implementation of digital filters for image processing with LabVIEW. These filters will be applied in neural network realized for image processing with LabVIEW. The filters are experimentally obtained. The investigation purpose is to find most appropriate filter for application to neural network realized with LabVIEW. From the one hand, this filter have to reduce image noise, from the other - essentially affect to the image is undesirable.

Two-dimensional convolution is used to implement local filters. The resulting image histograms are given to illustrate the filter effects in the output image.

## II. IMAGE FILTERING

There are many methods for image filtering. Most important of them are based on operations convolution and correlation. With convolution a number of filtering operations using local kernels can be implemented. Correlation uses matching the image neighborhood with pattern or mask [1].

<sup>1</sup>Liljana Docheva is with the Faculty of Telecommunications at Technical University of Sofia, 8 Kl. Ohridski Blvd, Sofia 1000, Bulgaria.

Image filtering, using local operators is used to perform smoothing, noise reducing, edge detection or sharpening. In this article is presented sharpening of the edges of image and noise reducing. Sharpening consists in isolating the edges and amplifying them and adding amplified edges back to the original image [2].

Image filtering, using local operators can be presented in Eq. (1).

$$R(x, y) = F\{I(x \pm \Delta x, y \pm \Delta y)\} , \quad (1)$$

where  $I(x \pm \Delta x, y \pm \Delta y)$  is a pixel of the input image with coordinates  $(x \pm \Delta x, y \pm \Delta y)$ , correspondingly  $R(x, y)$  is a pixel of the resulting image with coordinates  $(x, y)$  and  $F$  is an appropriate function. The output pixel is weighted combination of the gray values of pixels in the neighborhood of the input pixel. The size of the neighborhood and the pixel weights determine the action of the operator.

## III. IMPLEMENTATION OF IMAGE FILTERING WITH LABVIEW

Fig. 1 depicts one way of block diagram building for image filtering, realized with LabVIEW.

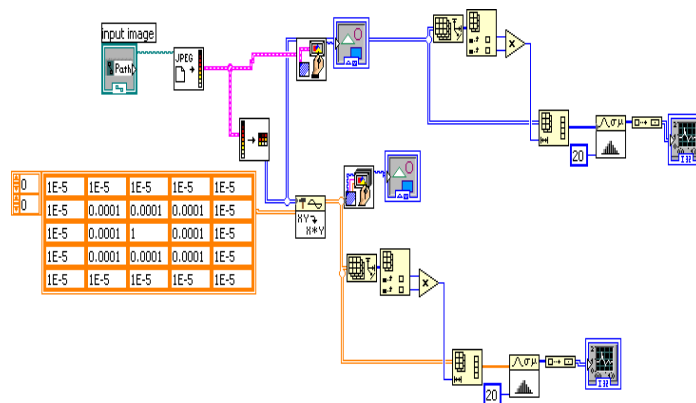


Fig.1. Block diagram of an image filtering implementation using LabVIEW

In order to realize the kernel, the convolution virtual instrument is used. For edge detection implementation the following virtual instruments are required:

- Read JPEG File to read a JPEG file,
- Unflatten Pixmap to converts a cluster of image data into a 2D array,
- Draw Flattened Pixmap to draw a 24-bit RGB pixmap into a picture,
- Draw Unflattened Pixmap to convert a pixmap into a picture image.

- Histogram to find the discrete histogram of the input sequence.

- Waveform Graph to display the histogram.

The histogram is created in order to point out the filter effect of the resulting images.

The test picture and image histogram are presented in Fig. 2. In purpose to perform noise reducing the local operators, shown in Figs. 3, 4 and 5 are used. Ones are experimentally obtained in order to find most appropriate filter for application to neural network realized with LabVIEW. The results can be seen too. The LabVIEW implemented filters coefficients increasing lead to image quality decreasing.

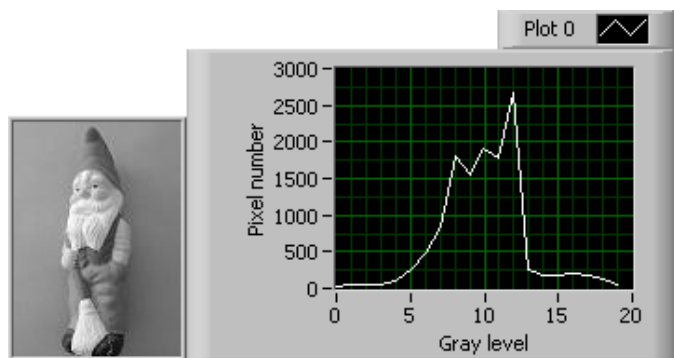


Fig. 2. Input image for image filtering and the corresponding histogram

0	1E-7	1E-7	1E-7	1E-7	1E-7
0	1E-7	1E-5	1E-5	1E-5	1E-7
	1E-7	1E-5	1	1E-5	1E-7
	1E-7	1E-5	1E-5	1E-5	1E-7
	1E-7	1E-7	1E-7	1E-7	1E-7

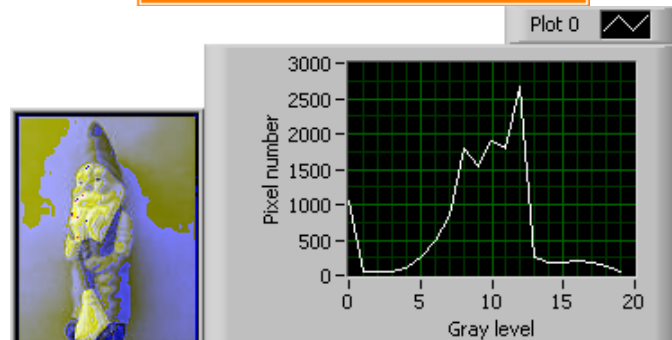


Fig. 4. Noise reducing

0	1E-6	1E-6	1E-6	1E-6	1E-6
0	1E-6	1E-5	1E-5	1E-5	1E-6
	1E-6	1E-5	1	1E-5	1E-6
	1E-6	1E-5	1E-5	1E-5	1E-6
	1E-6	1E-6	1E-6	1E-6	1E-6

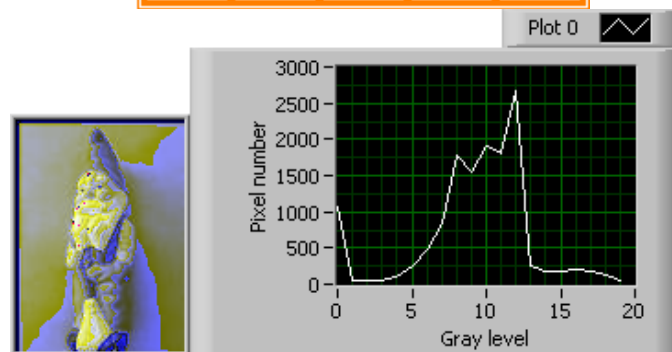


Fig. 5. Noise reducing

0	1E-9	1E-9	1E-9	1E-9	1E-9
0	1E-9	1E-7	1E-7	1E-7	1E-9
	1E-9	1E-7	1	1E-7	1E-9
	1E-9	1E-7	1E-7	1E-7	1E-9
	1E-9	1E-9	1E-9	1E-9	1E-9

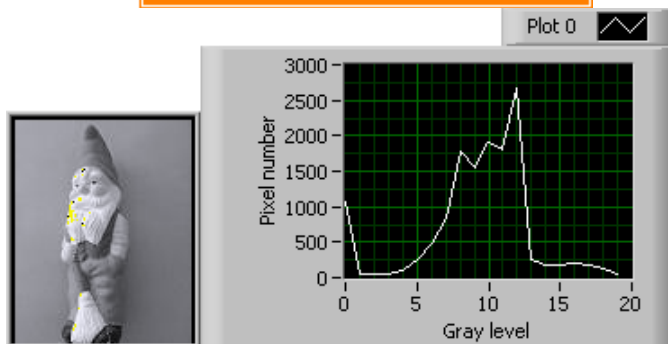


Fig. 3. Noise reducing

It can be seen from Fig. 4 that the LabVIEW implemented filters coefficients increasing lead to undesirable effects in the resulting image. Accordingly, Fig. 6 shows that very little increasing coefficients enhances this effect. Therefore, the filters shown in the Figs. 4 and 5 are not suitable for neural network application.

After image filtration the edges can be detected. The aim is to reduce the noises first and then to detect the edges. Fig. 6 depicts one way of block diagram building for edge detection after image filtering, realized with LabVIEW. For digital image filtering the first filter (Fig. 3) is used. It can be seen from Fig. 3 that this filter blur an image and so reduces noise. The results are presented in Figs. 7 and 8 for two different modified Laplas operators. The modification is made in order edge detection implemented with LabVIEW to give better results. The left picture is the result of the image filtering, the right one is the result of edge detection.

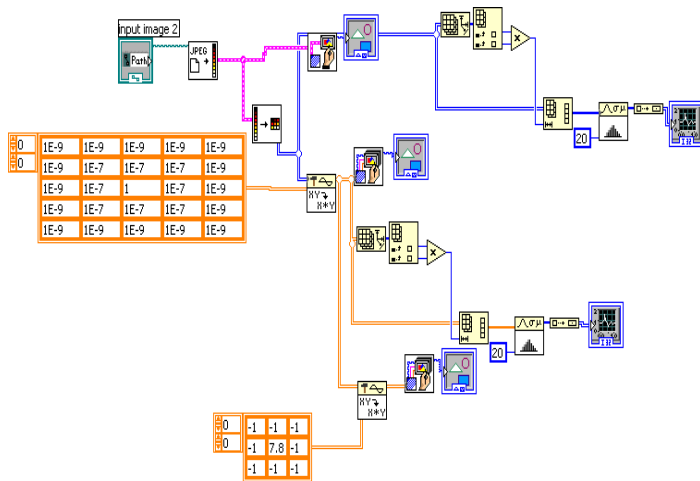


Fig. 6. Block diagram for edge detection after image filtering, realized with LabVIEW

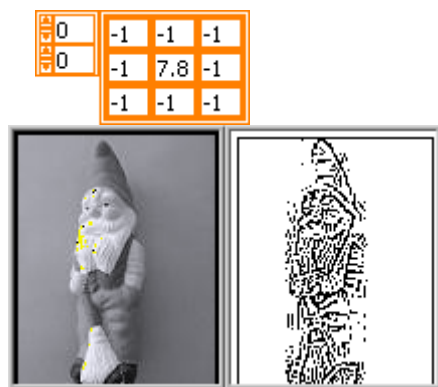


Fig. 7. Edge detection application after image filtering, realized with LabVIEW. The Laplas operator is shown at the top

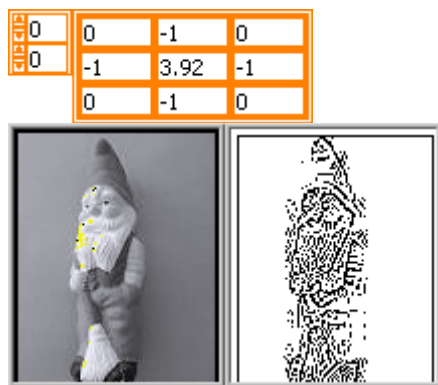


Fig. 7. Edge detection application after image filtering, realized with LabVIEW

The left picture from Figs. 7 and 8 is the result of image filtering, the right one is the result of edge detection.

The Block diagram of an image filtering implementation, using LabVIEW from Fig. 1 can be used for sharpening the edges of the image. One is obtained by means of the filters, presented in Figs. 8, 9 and 10. The results can be seen too.

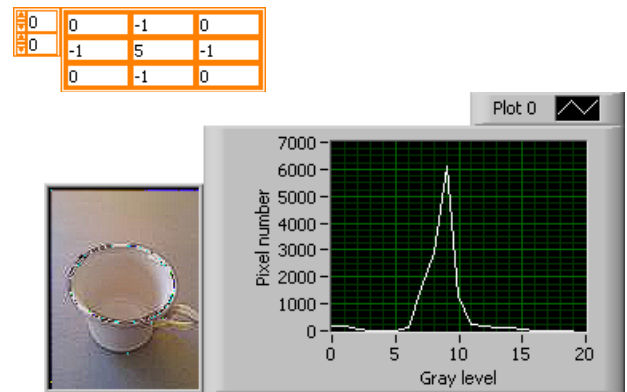


Fig. 8. Sharpening the edges of the image

Filters sharpen an image by enhancing the color contrast around edges in the image. Figs. 8, 9 and 10 show that increasing the LabVIEW implemented sharpen filters coefficients lead to image quality decreasing. From the figures can be seen the presence of the noiselements which are highly visible in Fig. 10.

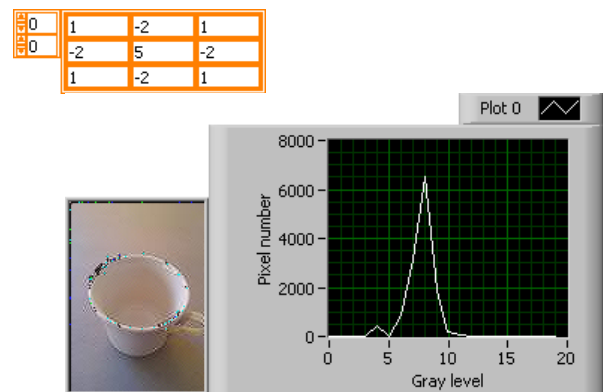


Fig. 9. Sharpening the edges of the image

From the filters investigated in this paper, most appropriate for application in neural network, realized for image processing with LabVIEW is the shown in Fig. 3 one.

This will be investigated in further research.

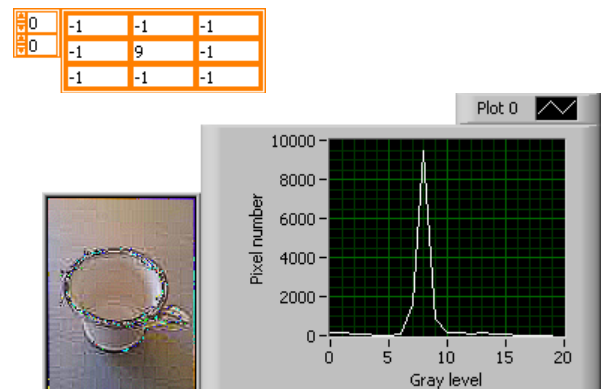


Fig. 10. Sharpening the edges of the image

#### IV. DIGITAL FILTER APPLICATION TO NEURAL NETWORK REALIZED WITH LABVIEW

In some articles have been successfully used LABVIEW for building neural networks [7], [8] because one has the ability to develop data flow that are highly parallel in structure. Therefore LABVIEW seems to be a very effective approach for building and investigating neural networks.

The first experimentally obtained filter (Fig. 3) will be applied in neural network realized for image processing with LabVIEW [6]. After processing the image can be presented to the network. The image values can be put into input vector and feed this vector into the network. Another way for neural network image presentation is by extracting features from the image and using them for neural network inputs. The presented in [6] neural network can be used in both cases. This will be the subject of further research.

#### V. CONCLUSION

In this article LabVIEW implementation of two-dimensional convolution kernels for image filtering is presented. With local operators sharpening of the edges of an image and noise reducing are performed. There can be seen the effects of the input image filtering by means of LabVIEW.

Two-dimensional convolution is used to implement local filters. The resulting image histograms are given to illustrate the filter effects in the output image.

It's studied which of these filters are appropriate to be applied in the neural network realized for image processing with LabVIEW. The first three filters are experimentally obtained. The investigation purpose is to find most appropriate filter for application to neural network realized with LabVIEW. It's shown that very little increasing

coefficients leads to undesirable effects in the resulting image (Fig. 4) and (Fig. 5). Therefore only the first of the presented three filters (Fig. 3) is suitable for neural network application.

Presence of the noise elements in the image can be seen in the Sharpening filters (Fig. 8), (Fig. 9), (Fig. 10). Therefore ones are not suitable for neural network application.

#### REFERENCES

- [1] Ch. Relf (2003), *Image acquisition and processing with LabVIEW*, CRC Press, 2004.
- [2] R. Maini, H. Aggarwal, "Study and Comparison of Various Image Edge Detection Techniques", *International Journal of Image Processing (IJIP)*, Volume (3) : Issue (1).
- [3] D. Rao, S. Patil, N. Babu, V. Muthukumar, "Implementation and Evaluation of Image Processing Algorithms on Reconfigurable Architecture using C-based Hardware Descriptive Languages", *International Journal of Theoretical and Applied Computer Sciences*, GBS Publishers and Distributors (India) Volume 1 Number 1, 2006, pp. 9-34.
- [4] R. Ingold, *Image Processing: Local Operators*, 2004.
- [5] G.T. Shrivakshan, C. Chandrasekar, "A Comparison of various Edge Detection Techniques used in Image Processing", *IJCSI International Journal of Computer Science Issues*, Vol. 9, Issue 5, No 1, September 2012, pp. 269-276.
- [6] L. Docheva, A. Bekjarski, "Implementation of Analog Neural Networks with Labview", *XLVII International Scientific Conference Information, Communication And Energy Systems And Technologies*, vol: 2, pp. 367 - 370, June 28-30, Veliko Tarnovo, Bulgaria, 2012.
- [7] J. Fernandez de Canete, S. Gonzalez-Perez, P. del Saz-Orozco, "Artificial Neural Networks for Identification and Control of a Lab-Scale Distillation Column using LABVIEW", *World Academy of Science, Engineering and Technology*, 2008.
- [8] M.A. Panait, T. Tudorache, "A Simple Neural Network Solar Tracker for Optimizing Conversion Efficiency in Off-Grid Solar Generators", *International conference on renewable energies and power quality (ICREPQ'08)*.2008.