ПРИБОРЫ И ТЕХНИКА ЭКСПЕРИМЕНТА, 2011, № 3, с. 76–80

ЭЛЕКТРОНИКА
И РАДИОТЕХНИКА

ACQUIRING NUMERICAL DATA FROM AN OSCILLOSCOPE SCREEN PICTURES

A. A. Azooz

Department of Physics, College of Science, Mosul University
Majmoa Street, Mosul, Iraq
Received September 17, 2010; in final form, November 9, 2010

This work describes an easy yet effective method to acquire numerical data from traces of signals obtained on analog or digital oscilloscope screens. Matlab user interactive based software is written for this purpose. The software uses matlab image possessing toolbox facilities. It is pointed out that data obtained using this technique can be even superior to those obtained by digital data acquisition units. The software is user friendly and it does not need any prior detailed knowledge of matlab programming.

INTRODUCTION

On many occasions, it becomes desirable and even necessary not only to have a visual assessment of a particular signal, but also to have information concerning that signal in one numerical form or another. This necessity arises from the need to carry out further analysis of the signal.

It is the purpose here to describe simple matlab based algorithm that can analyze digital photo pictures taken of an oscilloscope screen in order to obtain numerical data of that particular signal. The numerical data obtained can be saved on hard disc, or exported to other programs such as Microsoft's Excel for further analysis. The data quality is only limited by the picture resolution in pixel number.

BASICS OF IMAGE PROCESSING IN MATLAB

The matlab program has an image processing tool box [1]. A black and white picture in matlab is treated as a two dimensional array. The numerical value of each element in the two dimensional array representing a particular picture, is a measure of the intensity of the corresponding pixel in that picture.

Colored pictures are treated in almost the same way. Each colored picture is basically decomposable into three basic colors Red, Blue and Green RBG, the mathematical representation of such colored picture in matlab would be a three dimensional array consisting of three sub-arrays. It must be pointed out however that matlab uses eight bit representation of these numerical values. These can be easily converted to higher precisions.

One very powerful property that matlab enjoys is its ability to derive contour lines of any two dimensional array. The statement $[x, y]$ = contour(X, n) will produce a contour plot of the matrix X with n contour lines. It is this particular facility which is utilized here to build software that can convert pictures of line curves into x-y numerical data.

THE SOFTWARE USE

The software is written for the purpose of obtaining numerical data from single trace oscilloscope pictures. The software is completely freely available on the matlab file exchange website [2]. Detailed instructions on how to use the program are contained in the pdf user manual, which will be automatically downloaded with the program.

This manual is written with those who have limited or even no prior experience with matlab in mind. For multi-trace images, the image can be sliced into a number of images such that each image is analyzed independently. The program can handle the following picture file formats of the types usually handled by matlab release 13 and higher. These are: jpg, tiff, gif, bmp, png, hdf, pcx, xwd, ico, cur, pas, pbm, pgm, and ppm. Once the program is saved in the active directory, called by entering

[*x*, *y*] = *oscilloscope*(*x_sensitivity*,*y_sensitivity*,'*Picture name.format*',*Number_of_contours*),

the program will automatically load the particular picture of interest and the picture is shown on the screen.

One important step during the program run is the three mouse clicks at the origin, one unit scale to right, and one unit scale up from the origin. The accuracy of the positions of these three clicks determines the accuracy of the final x-y data. Figure 1 shows a summary of the commands and what one would expect to get at each stage during a typical run of the program.

THE SOFTWARE STRUCTURE

It is relatively easy to convert any single, continuous, contour line of a two dimensional matrix in matlab to numerical data, and then calibrate pixels points

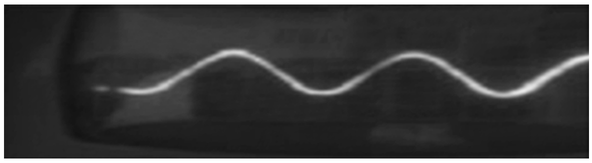
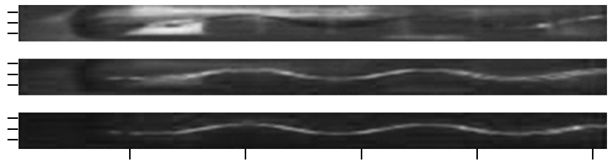
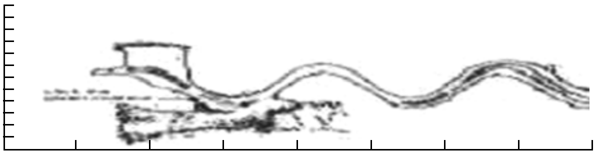
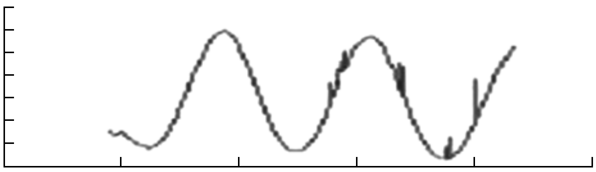
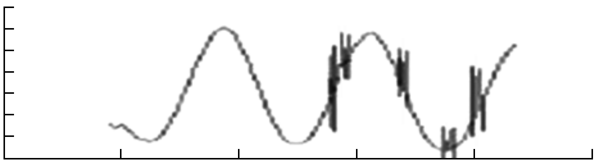
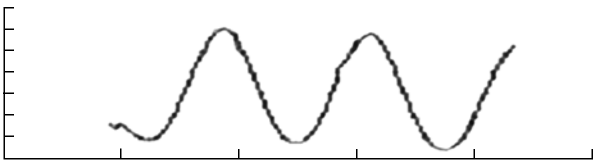
Prompts and Commands	Result
> [x, y] = oscilloscope (0.002.2.*01.jpg'2);	
Three mouse clicks	
Choose 1,2,or 3?? 2 enter	
When done, press RETURN while the Graph window is the active window	
What is your contour value ??? 128.5 Enter	
Does the trace needs any cleaning?? TYPE I for YES 1 enter	
How many astray spikes need to be leaned?? 5 enter	
	

Fig. 1. Summary of steps inputs and results for a typical run of the program.

corresponding to this contour line to numerical x-y data provided that two pairs of points on the x and y real scales are known. In practice however, a particular picture may usually involve a number of separate contours having the same level value. Example of such situation is the single level contour plot of a typical rectangular signal. Each individual contour has its own number of points. In such a case, the matlab contour matrix will have the form summarized in table.

Structure of single level multi-contour matrix in matlab V is the contour level value, N is the number of data point in the contour x and y are the pixel positions of the points in the particular contour

V	x_1	x_2	...	x_{N_1}	V	x_1	x_2	...	x_{N_2}	...
N_1	y_1	y_2	...	y_{N_1}	N_2	y_1	y_2	...	y_{N_2}	...

In order to decode such a matrix, x-y pixel data from all contours at the particular level selected by the user are pooled into the two originally empty one dimensional arrays disregarding very short isolated contour lines consisting of less than ten pixels. These usu-

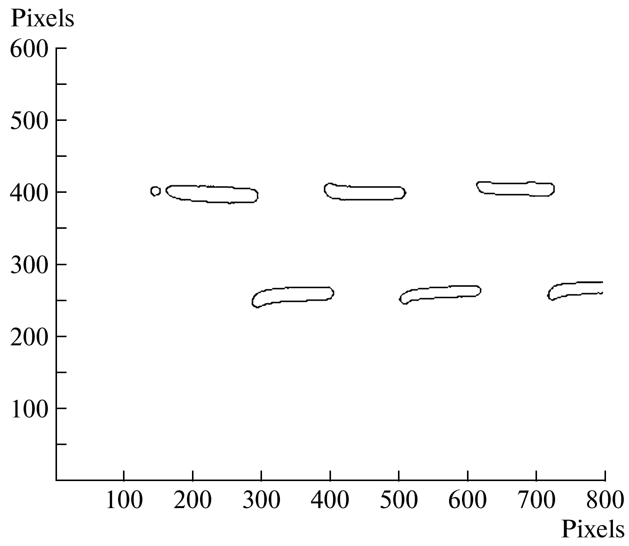


Fig. 2. Single level contour plot of a typical rectangular signal.

ally result from individual spots that may be present in the picture.

It is an inherited property of matlab contour plots produced as being vertically inverted as compared to the original pictures. The software corrects for this inversion. The pre-final problem that still remains is related to the fact that contour lines are only reflections of the color intensity along the edges of the trace. This is clear from Fig. 2.

The real signal position is approximately half way between the upper and lower contour boundaries. Points along these two boundaries are isolated separately "maxima" and "minima". The value of width between the two is calculated and final vertical positions are defined as the lower contour boundary plus half the width. After above analysis of the image, a pre-final x-y data plot is obtained. In some cases, this plot may show few stray data points originating from spots in the original picture. The software is equipped with an optional facility to manually eliminate these stray points. Detailed descriptions of the function of each statement are given as non executable comments lines within the software.

CASE STUDIES

For demonstration purposes, several case studied where oscilloscope screen pictures of signals are analyzed using the proposed software. It must be pointed out at this stage there are no restrictions on how the pictures are taken apart from the proper parallel position of the planes of the oscilloscope screen and the camera. One important necessary restriction however is the need that the picture shows the screen grid divisions and the trace to be extracted.

In Fig. 3a and 4a, pictures of a rectangular, and saw tooth signals of 200 kHz frequency and peak to peak voltage V_{pp} = 3.3 V, on an almost obsolete Metrix oscilloscope are presented. These pictures were taken using an ordinary mobile phone camera with ordinary room lighting. The pictures were originally colored. However, and for printing purposes, they are presented here in gray scale form. It may be worth mentioning here that the software can handle both colored and black and white pictures. For the latter type, step 2 in Fig. 1 is automatically skipped out. Plots of the corresponding numerical data retrieved from these two pictures

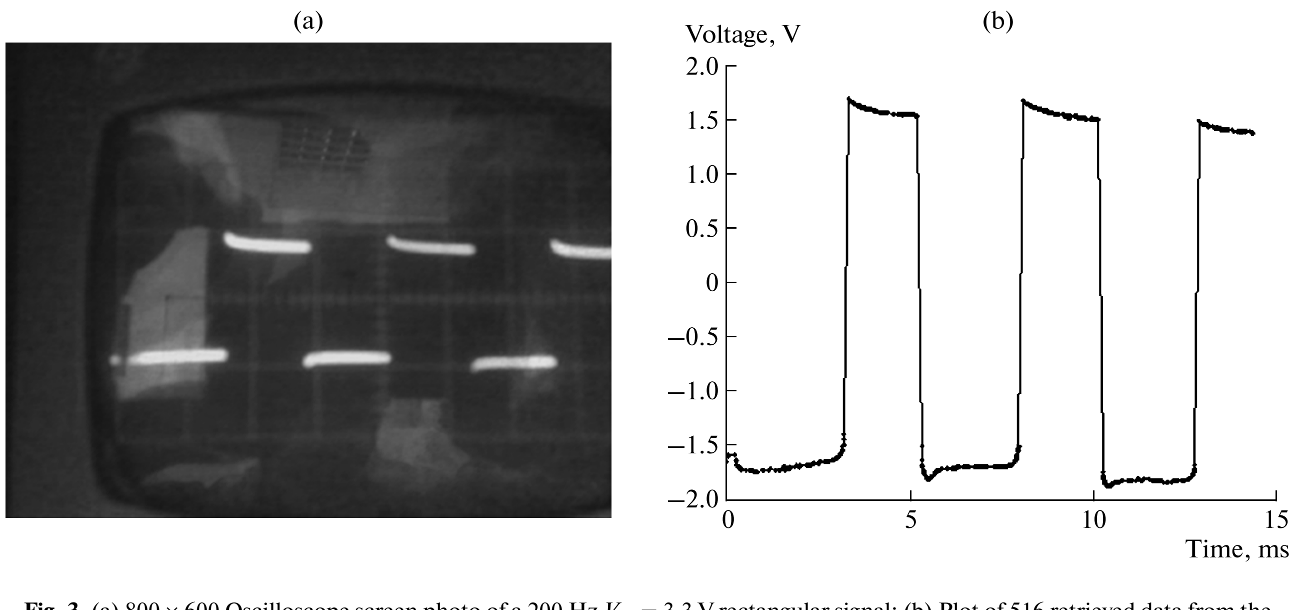


Fig. 3. (a) 800 × 600 Oscilloscope screen photo of a 200 Hz V_{pp} = 3.3 V rectangular signal; (b) Plot of 516 retrieved data from the trace in Fig. 3a.

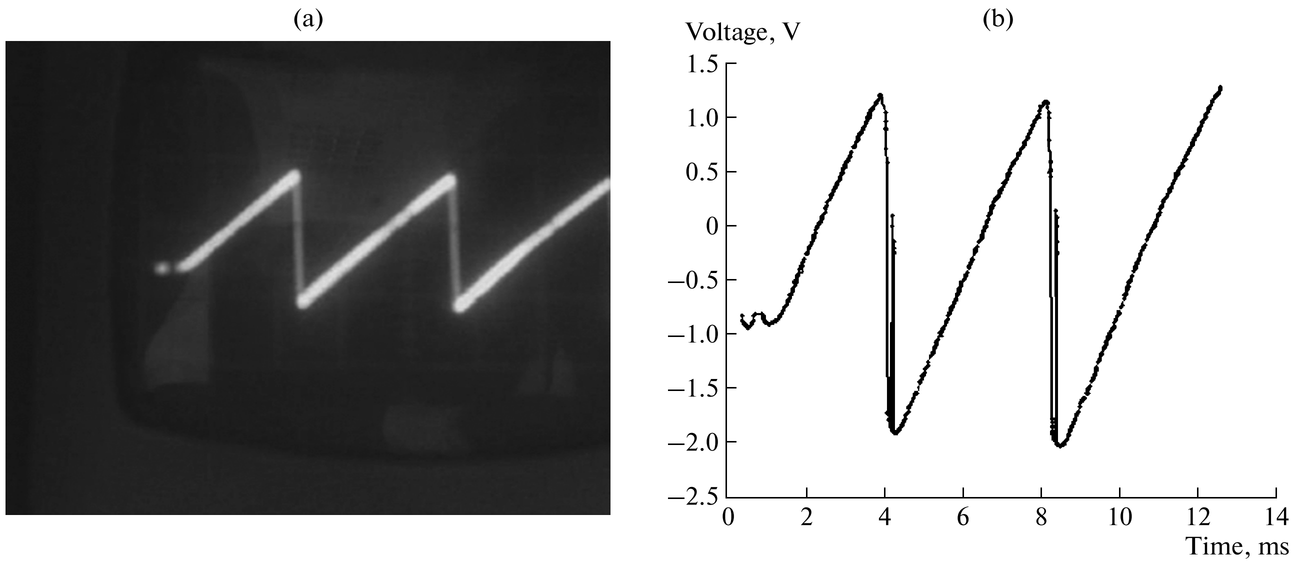


Fig. 4. (a) 800 × 600 Oscilloscope screen photo of a 200 Hz V_{pp} = 3.3 V saw tooth signal; (b) Plot of 662 retrieved data points from the trace in Fig. 4a. The picture was slightly obliquely taken.

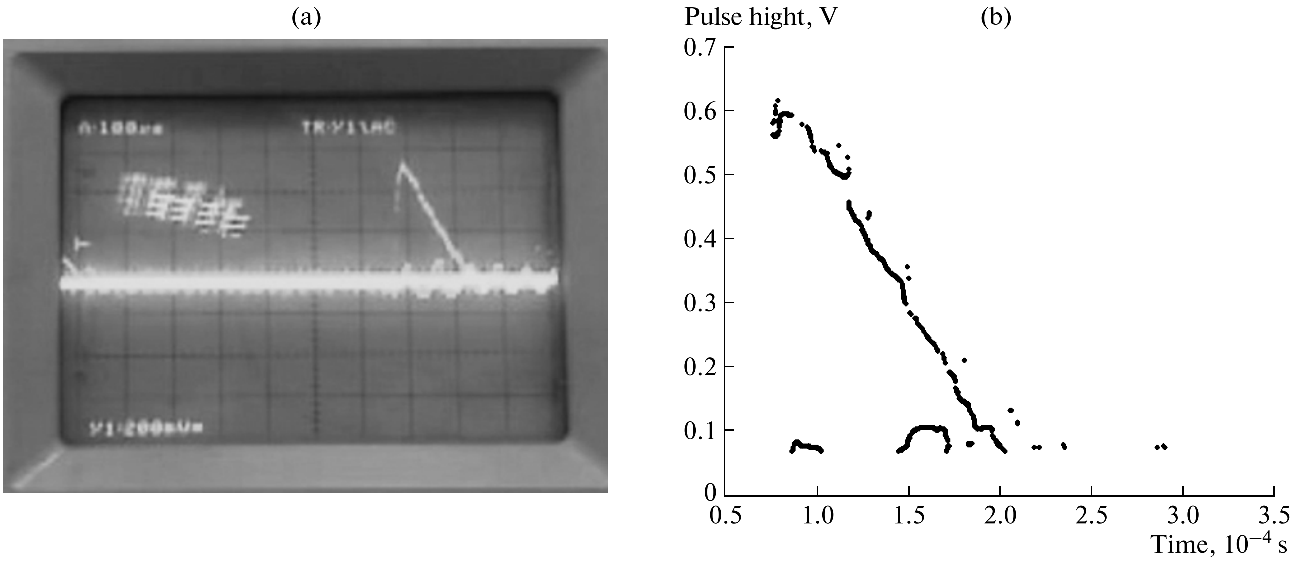


Fig. 5. (a) Picture of a Geiger-Muller tube pulse of about 200 μs duration and 0.6 V height; (b) Plot 708 retrieved data points of the trace in Fig. 5a. In spite of the poor pulse trace quality; this is equivalent to an average time resolution of about 0.3 μs.

are presented in Fig. 3b and 4b. Figure 5 shows the result of analysis of a 200 μs Geiger-Muller tube pulse on a HM400 oscilloscope screen. The pulse height is about 0.6 V.

The picture in Fig. 6 may be of some special interest. It is not an oscilloscope picture. It represents a scanned picture of a paper produced *x*-ray diffraction spectrum of some thin film structure. The image contains signatures of the folding lines of the instrument *x*-*y* recorder's paper. The manual stray data elimination facility has made it possible to delete such signatures from the data with out affecting the rest of the data. It must be pointed out however that the actual spectrum plot was in red color.

DISCUSSION AND CONCLUSIONS

Examination of all above case studies of different types of oscilloscope signals may indicate that the software can form a powerful tool for extracting numerical *x*-*y* values from oscilloscope or scanned paper pictures. The number of data points is in the range of several hundreds for pictures pixel sizes of 1–2 megapixels. The technique can be also utilized in many im-

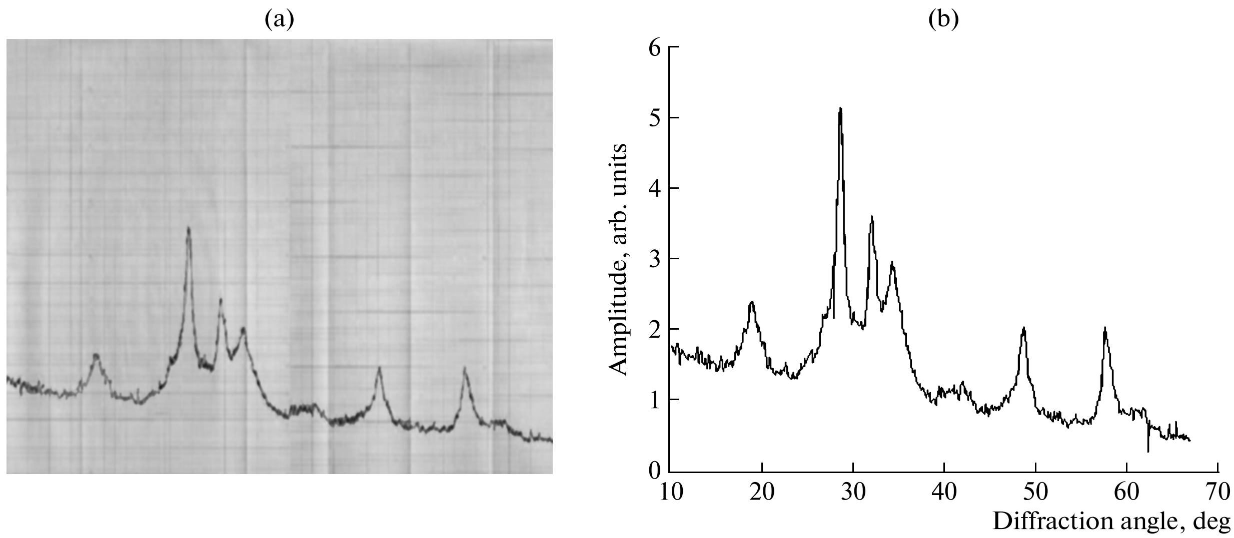


Fig. 6. (a) Scanned copy of the *x*-ray diffraction spectrum of some thin film; (b) Plot of 1746 data points retrieved from the picture in Fig. 6a.

age processing studies involving the characterization of a particular trace type within a particular picture.

The software can be easily used as an alternative to digital data acquisition devices. This is demonstrated for example in the picture of Fig. 6. The 708 data points covering 200 μs is equivalent to using a 3.5 MHz sampling rate data acquisition device. Prices of data acquisition devices operating in the MHz sampling rate range are in the order of few thousand US$ [3]. The trick here is to video record the signal and then selects the particular snapshot of interest.

The software is currently being used in our labs in several applications including Langmuir I-V characteristics analysis, modeling of CR-39 nuclear detector longitudinal track profiles, electromagnetic damping, and Fourier signal analysis undergraduate physics experiment.

REFERENCES

1. http://www.mathworks.com
2. http://www.mathworks.com/matlabcentral/fileexchange
3. http://sine.ni.com/np/app/main/p/bot/no/ap/daq/lang/en/pg/1/sn/n25:device,n17:daq/sb/-nigenso3-default