

Curve Digitizer – A software for multiple curves digitizing

Nicolae APOSTOLESCU, Florentin ȘPERLEA

INCAS - National Institute for Aerospace Research “Elie Carafoli”

Bdul Iuliu Maniu 220, Bucharest 061136, Romania

apostol@incas.ro, fsperlea@incas.ro

DOI: 10.13111/2066-8201.2010.2.2.2

***Abstract:** The Curve Digitizer is software that extracts data from an image file representing a graphic and returns them as pairs of numbers which can then be used for further analysis and applications. Numbers can be read on a computer screen stored in files or copied on paper. The final result is a data set that can be used with other tools such as MSEXCEL. Curve Digitizer provides a useful tool for any researcher or engineer interested in quantifying the data displayed graphically. The image file can be obtained by scanning a document.*

1. INTRODUCTION

The development of the Curve Digitizer is generated by the real need to get the original data which generated graphs (curves) that can be found in old documents or scientific publications which show some results only in graphical form. The Curve Digitizer extracts data from an image file representing a graph and returns them as pairs of numbers (x, y) which can be read on a computer screen, written to files or copied on paper. In the above-mentioned cases, the Curve Digitizer becomes a useful tool for any scientist or engineer interested in quantifying the data displayed graphically. It replaces the digitizer tablet.

Curve Digitizer Key Features

- It supports the most common image files formats such as BMP, GIF, JPEG, PNG and TIFF, etc. It can be obtained by scanning and saving in one of the above formats.
- The Curve Digitizer automatically manipulates the misaligned graphics scans
- It captures coordinates of manually digitized points via mouse click
- The Curve Digitizer automatically traces curve passing through digitized points (automatically draw orthogonal axes, the displacements along the axes are linear)
- It exports the digitized data in the TXT format file
- Several graphs (curves) of the same image can be sequentially and independently digitized
- It can also be used for digitizing any datasets in an image.

Curve Digitizer limits:

- The maximum number of points digitized on a curve should be smaller than 100;
- The maximum number of curves that can be digitized using the same orthogonal landmark must be less than 20;
- A limitation of the software is due to the accuracy and precision with which the user sets the mouse on the curve.

The four main steps for using this software are:

- Upload the picture graph in a predefined space called “picture space”;

- Calibrate the picture graph by mouse clicking on the items defining the axes. This process is needed because the application differentiates the picture coordinates from the display space coordinates;
- Digitize the data (Capture the digitized point coordinates);
- Save the digitized data and create a new graph.

For this application the display space is subdivided in three subspaces:

1. The “picture space” in which the image is loaded by pressing the” Load Picture” button. The size of the picture space is set as “auto size” thus depending of the image size.
2. The “communication space” used by the user to communicate with the application. From here, the user is able to:
 - a) start the import of the image file;
 - b) set 3 points as endpoints for the axis (R_O, R_P,R_Q);
 - c) give optional information like: the file name of output point data, title of graph, x-axis name, y-axis name, data as legend;
 - d) start digitizing a new curve (or a new set of data) if multiple;
 - e) start drawing the new curve (s) which was (were) obtained by digitizing
3. The “mschart space”. In this space, the new design will be automatically created, after the digitization process has been completed.

2. USAGE

It is considered that the user has a graphic image which will be digitized in one of the formats: BMP, GIF, JPEG, PNG and TIFF (e.g. obtained by scanning).

Step 1

- 1.1. The user uploads the image by pressing control "Load Picture"

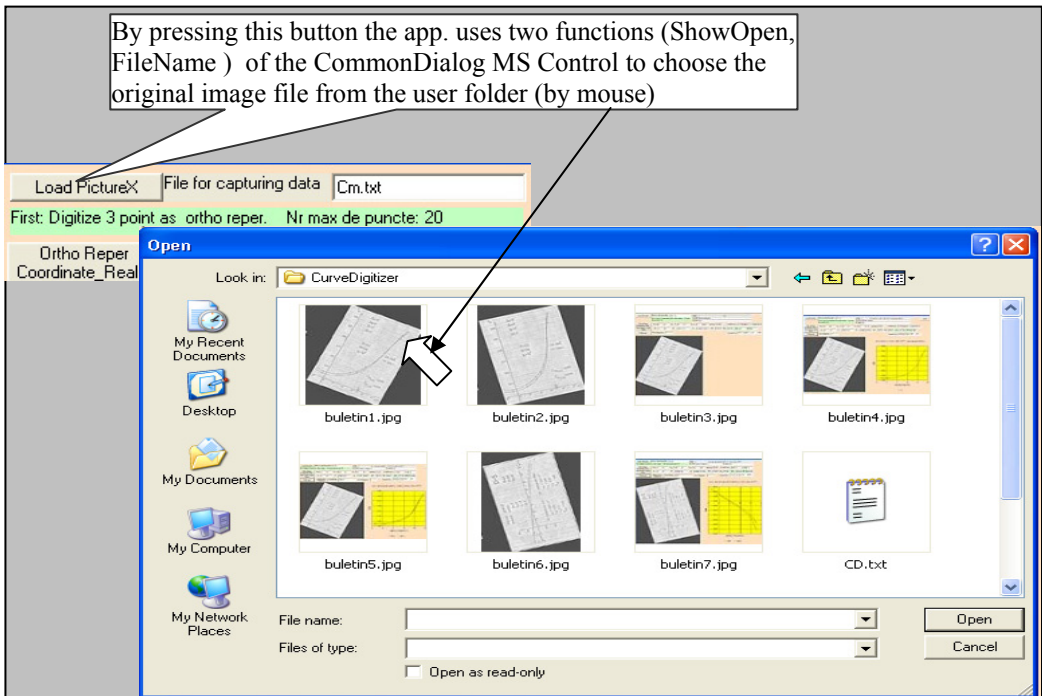


Fig.1

The following will be optionally written in the appropriate boxes:

- File name to store the data captured by digitizing (without extension, because it will automatically receive the “.txt” extension)
- New image name;
- Name of the data on the x and y axes;
- Name of the data set (may be important in the case when digitizing more data sets).

1.2 The user selects three points in the original design as endpoints for the orthogonal coordinate axis and inserts their values into boxes R_ O, R_ P, R_ Q (R_ P as a point on the positive axis Ox , ...)

Step 2

Note. Steps 2 and 3 work in "picture-space"

Whenever pressing the mouse in "picture-space" the data capture process is active (that is the drawings coordinates of the digitized point are recorded and displayed while the display coordinates are only displayed).

2.1. The three points chosen to define the original coordinates mark will be successively digitized on the image. For information, the display coordinates of these points will be written in boxes O, P, Q; the same for the scale factor for each axis and the image rotation angle about the axis Oy of the display.

Step 3

The user digitizes points from the curve of the original image by mouse clicking (maximum 100).

To digitize points on a new curve, press “New Diagram”. “And optionally specify a name that will be used as a legend, then start the process of digitization. This step can be repeated for any new curve.

Note:

The number of digitized points may vary from curve to curve.

It is not necessary to digitize at the same abscises on different curves.

The 3 points used as reference points are digitized once even in case of multiple data sets digitization

Step 4

Press the button. “Plot Graph”.

All digitized data are recorded in the output file and in the "mchart space" curves will be drawn from the digitized points. The output file .txt will contain ASCII records, each line containing: no-curve, x_coord, y_coord.

The following examples illustrate the capabilities of this application and the way to use it.

1. The scanned image is stored in a JPG file. It forms a CW angle of 25.201 degrees to the OY axis; this rotation is calculated automatically.

The OP and OQ are automatically calculated to determine the scale factor on each direction. The box "current pos" contains the real and the display coordinates for the last digitized point called "current point".

Output filename

Optional data : title, x-axis, yaxis

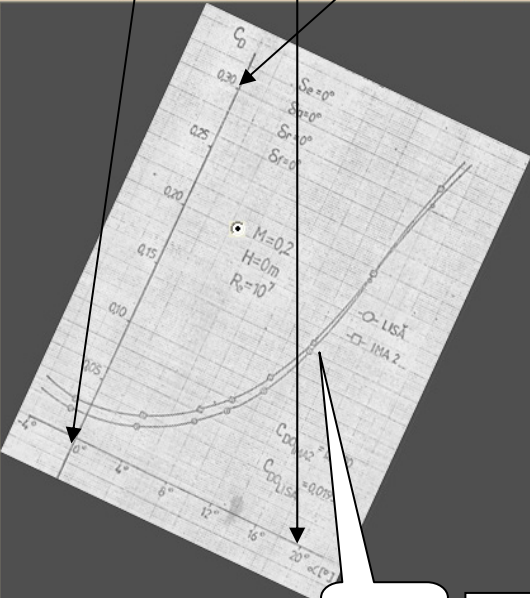
Load Picture: File for capturing data: CD.txt Title: CD (de=da=dr=df=0), M=0.2; H=0; Re=10**7

First: Digitize 3 point as ortho reper. Nr max de puncte: 20 X_Axis: alpha [degrees] Y_Axis: CD

Ortho Reper Coordinate_Real: R_O [0] [0] R_P [20] [0] R_Q [0] [0.3] ALPHA [25.201] ScaleFactor [1.019E-1] [9.98E-4]

New Reper Graph Data: D [0] [0] F [196.35] [0] Q [-4.181] [300.61] OP [196.35] OQ [300.61] DM [372.83] BETA [34.840]

PlotGraph: Digitize New Diagrame: CurveName: IMA 2 Nr of diagrams: 2 Current Pos: 18.96 [0.3223] 517 [837]



Initial image

CD (de=da=dr=df=0), M=0.2; H=0; Re=10**7

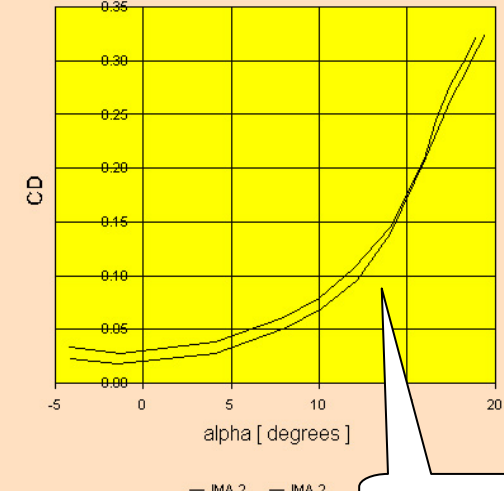


Image after digitize

CD.TXT

CurveNr	x	y
1	-4.136495	2.283994E-02
...		
1	19.42472	0.3244866
2	-4.20155	3.356941E-

Fig.2

- The next BMP image is rotated CCW (or CW with 255.06 degrees) and in landscape position of the paper.

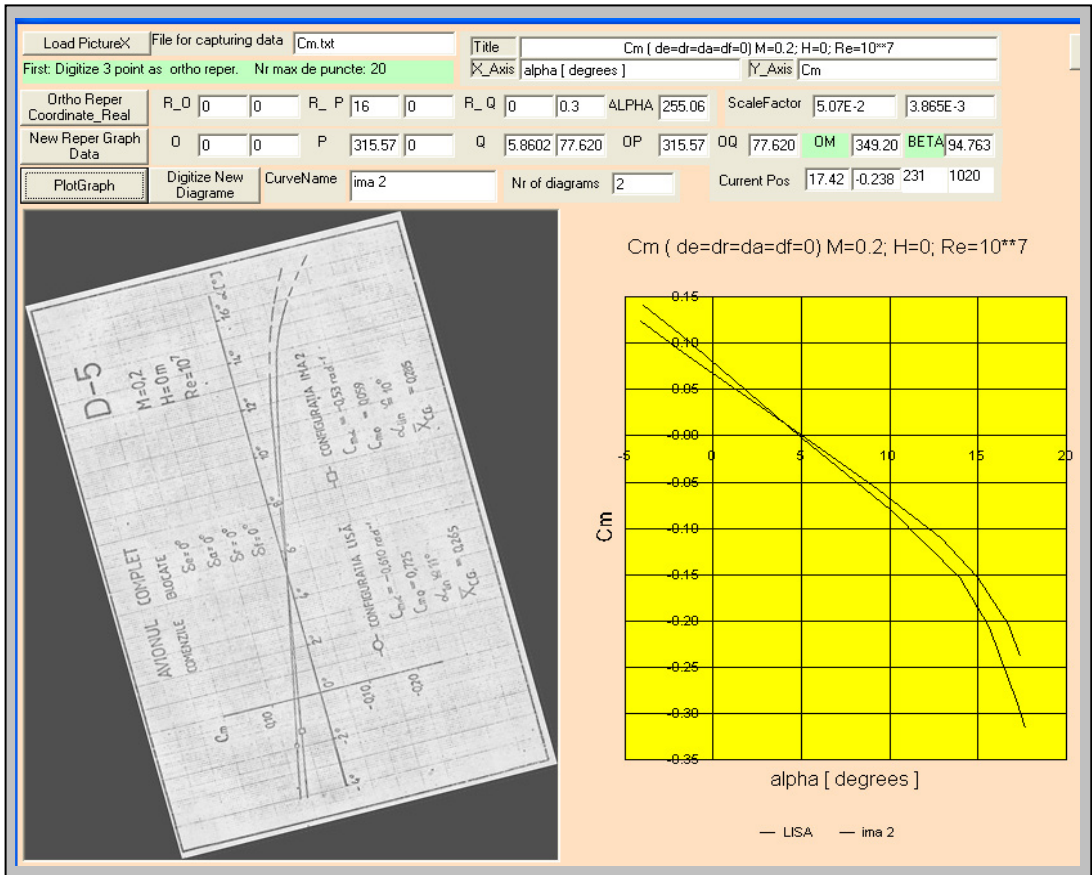


Fig.3

The corresponding Cm.txt file shows like

	CurveNr	x	y
1	1	-3.968155	0.1406639
1	1	-1.871401	0.1100415
		...	
1	1	17.15941	-0.2848134
1	1	17.68197	-0.3154358
2	2	-4.131454	0.1239834
		...	
2	2	16.70871	-0.2036516
2	2	17.42722	-0.2382573

The application is developed using Visual Basic 6 under Windows and it is used through a user graphical interface. The controls that are used are common controls like buttons to generate events that are handled by code, labels and text control to set or capture data, picture as space to image file and special Microsoft components like Common Dialog Control 6 (comdlg32.ocx) to open imagefile and Chart Control 6 (mschrt20.ocx) to create the original graphical chart to be visually verified.

The special Windows functions that are used:

Function GetCursorPos Lib "user32" (lpPoint As POINTAPI) As Long

Function ScreenToClient Lib "user32" (ByVal hWnd As Long, lpPoint As POINTAPI) As Long
 Function GetSystemMetrics Lib "user32" (ByVal nIndex As Long) As Long

Special configuration for Mschart

```
MSChart1.Height = Picture1.Height / 2: .Width = Picture1.Width / 2
MSChart1.Left = Picture1.Left + Picture1.Width
MSChart1.ChartType = VtChChartType2dXY
MSChart1.ShowLegend = True
.Plot.Axis(VtChAxisIdY).AxisScale.Type = VtChScaleTypeLinear
.Plot.Axis(VtChAxisIdX).AxisScale.Type = VtChScaleTypeLinear
```

3. CONCLUSIONS

Curve Digitizer is well suited for digitizing measured curves which have been published as a plot, or similar graphical data. The digitized points can be transformed into the coordinate space of the paper plot by calibrating the virtual grid on the paper (which is simply specifying three points on the grid). The calibrated data can then be transferred into the data matrix (XY).

REFERENCES

- [1] D. Benyon, T. Green, D. Bental, *Conceptual Modeling for User Interface Development*, Editura Springer, ISBN 1-85233-009-0.
- [2] Christopher J. Bockmann, Lars Klander, Lingyan Tang, *Visual Basic*, Editura Teora, ISBN 973-601-912-8.
- [3] Kris Jamsa, Lars Klander, *Visual Basic*, Editura Teora, ISBN 973-9392-48-2.
- [4] Nabajyoti Barkakati, *The Waite Group's Turbo C++ Bible*, SAMS –A Division of Macmillan Computer Publishing, 11711 North College Carmel, Indiana USA, ISBN 0-672-22742-8.