Designing a Software to Generate Digital Elevation Models Using Engineering Survey Data and Performing 3d Analysis

Mr. A.M.A.R. Geethalankara (Author)
Department of Spatial Sciences
Faculty of Built Environment and Spatial Sciences
Suriyawawa, Sri Lanka
ranjanageethalankara@gmail.com

Mr. K.L.P. Hemantha (Author)
Department of Surveying and Geodesy
Faculty of Geomatics
Belihuloya, Sri Lanka
hemanthaklp@geo.sab.ac.lk

Abstract- This research study was conducted for designing a new plug in software (DemCAD), which will generate DEMs using grid format Data such as Engineering Survey Data. This study was mainly focused on generating two Digital Elevation Models, those are Contour Model and Mesh Model, in that contour model gives 90% accurate result and Mesh Model gives almost perfect accurate result when comparing with Civil 3D and Matlab Software. DemCAD software contains set of programs which will allow user to input grid data, after it will perform a data analysis then finally it will output a script file automatically. Then user has to run this script file in AutoCAD 2D/3D Environment. This script contains set of commands which will be recorded in particular sequence and pattern, also to get complete DEM in AutoCAD Script file have to run on AutoCAD without interrupting.

The most important fact in this study was, Mesh DEM could be used to perform various type 3D analyses, such as Flow Direction Map, Flow Accumulation Map, Pour Point, etc. In that, input grid data was processed in a specific way then data will be passed to another executable file which will use that data and perform 3d analysis process. In this procedure functions were designed for kernel operations, various kinds of calculations such as calculating distance, Bearing, Interpolate Coordinates, etc.

Future more this mesh model and 3D analysis could be used to lots of applications such as Land Survey, Engineering, Quantity Survey, Irrigation, and even for military applications. Even though in this study only land survey data was used to generate DEMs, any grid format data sources such as Laser Scanning Remote Sensing Data, Radar Data, even Close Range Photogrammetry stereo pair data, etc. could be used as data sources after some pre-preparation.

Key Words— Digital Elevation Models, 3D Analysis, Script.

I. Introduction

Digital Elevation Models play a major role in representing terrain or surface in computers. Most of the time, data collecting to create Digital Elevation Models are done by Photogrammetry or Remote Sensing. The only DEM created by land survey data collecting is Contour model, other DEMs are used rarely. In this research project, an effort was made to create as many DEMs as possible using land survey data.

Usually surveyors produce and use contour maps in two dimension environment such as plain paper computer screens. Even though in drawing software like AutoCAD contain such three dimension environment facilities. Still surveyors rarely use three dimension environment facilities, to fulfill drawing requirements. So in this research project, an effort was made to find how 3D environment facilities in AutoCAD can be modified and improved according to surveyors requirements. Usually AutoCAD software modifications were mainly done using Auto Lisp or Visual Lisp programming language facilities provided by AutoCAD itself. But in the research most of the modifications were done by using C++ programming language and only linking C++ executable files to AutoCAD was done by using Auto Lisp scripts.

Digital Elevation Models

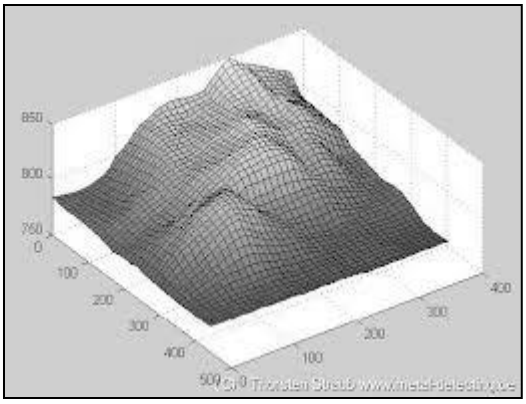


Figure 1 Digital Elevation Modal

Digital representation of terrain elevation in particular area with a digital model as shown in the figure can be described as DEM.

In the duration of this research work, managed to developed software to generate two main DEMs. They were Contour model and Mesh DEM. Both this DEMs were generated using Engineering survey data sets, because they were in grid format.

Free and Open Source Software Development

FOSSES stands for free and open source software. This is software which can classify as free to use and open source. That means anyone is freely licensed to use the software and make copies, can change the software any way. Since the source code of the software is open to access, people are encouraged to voluntarily improve the design of the software.

Use of Script Files in AutoCAD

A script is a macro, a list of commands that user can run all at once, and as many times as necessary. Script files allows user to automate tasks that would take a long time if they were done manually. Scripts can be very powerful and user can run them on objects in one drawing, or on many drawings. Scripts have been around for many years and many people have a library of many scripts that they use.

There are three important points about scripts.

Scripts are text-only (ASCII) files. Usually create them in Notepad.
They have an SCR filename extension, so be sure to save them that way.
Scripts use command-line syntax only. They can’t access dialog boxes, toolbar buttons, etc.

Problem Statement

AutoCAD is one of the most popular languages between many users in the world. Surveyors, Engineers, Architectures are familiar to use this software for drawing their plans. In the case of many geographic data analysis projects, it is required to represent the relief. Since contour model is one of the relief representation methods, surveyors and engineers prepare contour plans for the site. The AutoCAD software environment does not provide any tool for generating contours. So they have to use other software (Civil3D, Terramodel, Surfer) or plug in (LanDTM, Terrain for AutoCAD) which can run inside the AutoCAD for creating contour model. Although above software provide contour models and analyzing functions, there are some difficulties and problems mentioning as follows.

All of the software in general use and drawing are commercial. As a developing country, who is working in CAD environment has no ability to spend considerable amount to buy the plug-ins in Sri Lanka.
There is no permission or license to make copies from the existing drawing software. For working in intranet domain, the companies have to purchase a license version paying more for a key.

II METHODOLOGY

To develop new embedded software for generating contour models required well theoretical knowledge of digital elevation modeling, Computer programming and graphic user interface designing software. Engineering survey data collected by survey students of Sabaragamuwa University and a common data model (a data set contains many contour shapes such as peaks, pits, valleys) are used as input data sets. Programs for data processing and contour drawing (Command Line Interface code) have developed in C++ Win32 Console Application environment. Graphical user interface program code (GUI) was developed in Visual C++ Win32 environment.

Program code for data processing (CLI)

At initial stage the program reads given grid based data from a file called Data.txt. Data.txt file number of points, contour interval, drawing style, contains coordinate list. The code designed to fulfill the above task has mentioned following.

```
ifstream myfil0("D:/AutoCADLISP/Data.txt");
         {
        if(!myfil0)

          {
        cout << "File Not Found\n";
        my=0;
        getch();
          }
        else
          {
        myfil0>>SN;
        myfil0>>CI;
         myfil0>>cb1cke;
        for (int i=0;i<SN; i++)
          {
           myfil0>>PT[i]>>X[i]>>Y[i]>>Z[i];
          }
          }
          }
          myfil0.close();
```

Then the input coordinate list is arranged in ascending order according to their x values.

```
for (int i=0;i<SN;i++)
  {
 for(int j=i+1;j<SN;j++)
       {
       if(X[i]>X[j])
       {
       tempx=X[j];     tempy=Y[j];   tempz=Z[j];
       X[j]=X[i];      Y[j]=Y[i];    Z[j]=Z[i];
       X[i]=tempx;     Y[i]=tempy;   Z[i]=tempz;
       }
       }
       }
```

Minimum and maximum x and y values of the given data set is selected using following functions.

```
long double CalMin(long double *CCZ,int g)
{
```

```
int min=100000000;
 for(int o=0;o<g;o++)
{
if (CCZ[o]<min)
min=CCZ[o];
 }
 return min;
 }


long double CalMax(long double *CCZ,int g)
 {
int max=0;
 for(int o=0;o<g;o++)

    {
    if (CCZ[o]>max)
    max=CCZ[o];
    }
   return max;
    }
```

Above two functions are called in the main program using following codes.

```
minX=CalMin(X,SN);
minY=CalMin(Y,SN);
maxX=CalMax(X,SN);
maxY=CalMax(Y,SN);
```

Minimum distances between two adjacent grid points are calculated along X axis and Y axis. Above CalMin and CalMax functions are used for this step.

Then number of columns and number of rows are determined.

After that the number of spot heights between each and every adjacent grid data points is estimated. Spot height values of the points are also calculated. This step is performed with the help of FindInterpolatingHeight function.

```
int FindInterpolatingHeight(long double zzm,long double zz,int CI,int s,int t,int f,int g)
{
int NC,H,LC;

if(zz>zzm)
{
///cout<<"Large="<<zz;
///cout<<"Smaller="<<zzm<<endl;
if (fmod(zzm,CI)!=0) LC=(abs(zzm/CI)+1)*CI;
else LC=zzm;
 if (fmod(zz,CI)!=0)
 {
  if(((LC)>=zzm) && ((LC)<=zz))
 {
 NC=(round(((zz-zzm)/CI)));/*cout<<"NC="<<NC<<endl;*/
 }
 else
 {
 NC=0;
 }
 }

  else {NC=(round(((zz-zzm)/CI))+1);/*cout<<"NC"<<NC<<endl;*/}
  }
  else
  {
  //cout<<"Large="<<zzm;
  //cout<<"Smaller="<<zz<<endl;
  if (fmod(zz,CI)!=0) LC=(abs(zz/CI)+1)*CI;
  else LC=zz;
  if (fmod(zzm,CI)!=0)
  {
   if(((LC)>=zz) && ((LC)<=zzm))
  {NC=(round(((zzm-  zz)/CI)));/*cout<<"NC="<<NC<<endl*/;}
   else{NC=0;}
   }
   else {NC=(round(((zzm-zz)/CI))+1);/*cout<<"NC="<<NC<<endl;*/}
   }

   //cout<<"LC="<<(LC);
   //cout<<" NC="<<NC<<endl;
   if(NC>0)
  {
  for(int e=0;e<=(NC-1);e++)
  {
  H=(LC+e*CI);
   //cout<<H<<"\t";
   FindLengthBearing(H,s,t,f,g,w);
   w=w+1;
                                     24
    // k=w  ;
    }
    //cout<<endl;
    }
    return w;
    }
```

Inside above function, x,y coordinates for each and every spot height are calculated using linear interpolation method by calling following functions(FindLengthBearing, FindBearing)

```
long double FindLengthBearing(int H,int s,int t,int f,int g,int w)
{
 long double CD,BE;
 ///cout<<"[("<<x[s][t]<<","<<y[s][t]<<")"<<"-"<<x[f][g]<<","<<y[f][g]<<")]";
 ///cout<<z[s][t]<<" "; cout<<z[f][g]<<" "; cout<<H<<endl;
 CD=sqrt(  ((x[f][g]-x[s][t])*(x[f][g]-x[s][t]))+(  (y[f][g]-y[s][t])*(y[f][g]-y[s][t]) )  );
 ///cout<<"Length="<<CD<<endl;
 BE=FindBearing(s,t,f,g);

///cout<<"Bearing="<<BE*180/3.1415926535897932384626433832795<<" ";
 cz[w]=H;
  if(z[f][g]>z[s][t])
```

```
      {
       //cout<<"                                          CD="<<CD<<"
Zmin="<<z[s][t]<<"Zmax="<<z[f][g]<<"      cz="<<cz[w]<<"  ";
       cdp[w]=(((cz[w]-z[s][t])*CD)/(z[f][g]-z[s][t]));
       cx[w]=x[s][t]+cdp[w]*cos(BE);
        cy[w]=y[s][t]+cdp[w]*sin(BE);

//cout<<w<<","<<cdp[w]<<","<<cx[w]<<","<<cy[w]<<","<<cz[w]<
<endl;
        }
        else

      {
      //cout<<"             CD="<<CD<<"             Zmin="<<z[f][g]<<"
Zmax="<<z[s][t]<<"cz="<<cz[w]<<" ";
       cdp[w]=(((cz[w]-z[f][g])*CD)/(z[s][t]-z[f][g]));
       cx[w]=x[f][g]-cdp[w]*cos(BE);
       cy[w]=y[f][g]-cdp[w]*sin(BE);

//cout<<w<<","<<cdp[w]<<","<<cx[w]<<","<<cy[w]<<","<<cz[w]<
<endl;
       }
       return 0;
       }

     long double FindBearing(int s,int t,int f,int g)
     {
     long double A;
      if(y[f][g]!=y[s][t])
     {
         If (x[f][g]!=x[s][t])
         {
          if(((y[f][g]-y[s][t])>0)           &&           ((x[f][g]-x[s][t]))>0)
A=(atanl((y[f][g]-y[s][t])/(x[f][g]-x[s][t])));
          if(((y[f][g]-y[s][t])<0)           &&           ((x[f][g]-x[s][t]))>0)
A=(atanl((y[f][g]-y[s][t])/(x[f][g]-x[s][t])));
          if(((y[f][g]-y[s][t])>0)           &&           ((x[f][g]-x[s][t]))<0)
A=(atanl((y[f][g]-y[s][t])/(x[f][g]-
x[s][t]))+3.14159265358979323846264338327950);
          if(((y[f][g]-y[s][t])<0)           &&           ((x[f][g]-x[s][t]))<0)
A=(atanl((y[f][g]-y[s][t])/(x[f][g]-
x[s][t]))+3.14159265358979323846264338327950);
         }
         else
         {
         if                                                    (y[f][g]>y[s][t])
A=(3.14159265358979323846264338327950/2);
         else A=(3*3.14159265358979323846264338327950/2);
                                              26
     }
     }
      else
     {
        if (x[f][g]>x[s][t]) A=0;
        else A=(3.14159265358979323846264338327950);
     }
     return A;
     }
```

After calculating every possible spot heights, there might be identical x,y,z coordinates. So such coordinates were eliminated using following filtering code.

```
count=0;
for(int i=0;i<w;i++)
{
for(int j=0;j<i;j++)
{
 if(i!=j)
{
 if((cx[i]==cx[j])&&(cy[i]==cy[j])&&(cz[i]==cz[j]))
{
 cout<<i<<" ";
 cx[i]=1000000; cy[i]=1000000; cz[i]=1000000;
 cout<<endl;
 count=count+1;
 }
 }
 }
                                         27
 }
for (int i=0;i<w;i++)
{
for(int j=i+1;j<w;j++)
{
 if(cz[i]>cz[j])
{
 Temph=cz[j];     Tempx=cx[j];    Tempy=cy[j];
 cz[j]=cz[i];     cx[j]=cx[i];   cy[j]=cy[i];
 cz[i]=Temph;     cx[i]=Tempx;   cy[i]=Tempy;
 }
 }
 }
 w=w-count;
```

Then the minimum & maximum heights are calculated from above filtered spot heights.

```
minh=CalMinZ(cz,w);
maxh=CalMaxZ(cz,w);
```

Amount of possible spot height value groups is estimated.

```
nn=((maxh-minh)/CI);
```

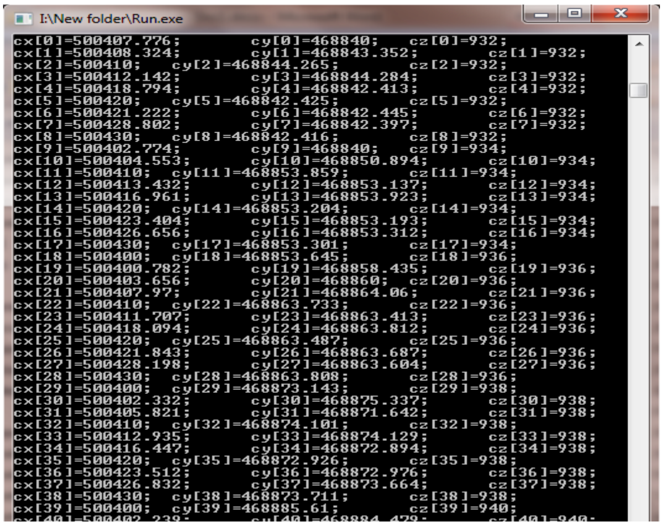
References for the spot heights were stored in to array named Hino. As an example if there are spot heights between 932 to 938, then the references were stored as follows.

```
Hino[0]=932;
Hino[1]=934;
Hino[2]=936;
Hino[3]=938;
```

Following code has use for above task.

```
for(int zz=0;zz<=nn;zz++)
   {
        Hino[zz]=minh+CI*zz;
        }
        for(int zz=0;zz<=nn;zz++)
        {
        cout<<zz<<" "<<Hino[zz]<<endl;
  }
.
```

Above grouped Spot height coordinate lists are arranged (ascending order) according to their x coordinates.



```
cx[0]=500407.776;       cy[0]=468840;   cz[0]=932;
cx[1]=500408.324;       cy[1]=468843.352;       cz[1]=932;
cx[2]=500410;   cy[2]=468844.265;       cz[2]=932;
cx[3]=500412.142;       cy[3]=468844.284;       cz[3]=932;
cx[4]=500418.794;       cy[4]=468842.413;       cz[4]=932;
cx[5]=500420;   cy[5]=468842.425;       cz[5]=932;
cx[6]=500421.222;       cy[6]=468842.445;       cz[6]=932;
cx[7]=500428.802;       cy[7]=468842.397;       cz[7]=932;
cx[8]=500430;   cy[8]=468842.416;       cz[8]=932;
cx[9]=500402.774;       cy[9]=468840;   cz[9]=934;
cx[10]=500404.553;      cy[10]=468850.894;      cz[10]=934;
cx[11]=500410;  cy[11]=468853.859;      cz[11]=934;
cx[12]=500413.432;      cy[12]=468853.137;      cz[12]=934;
cx[13]=500416.961;      cy[13]=468853.923;      cz[13]=934;
cx[14]=500420;  cy[14]=468853.204;      cz[14]=934;
cx[15]=500423.404;      cy[15]=468853.193;      cz[15]=934;
cx[16]=500426.656;      cy[16]=468853.312;      cz[16]=934;
cx[17]=500430;  cy[17]=468853.301;      cz[17]=934;
cx[18]=500400;  cy[18]=468853.645;      cz[18]=936;
cx[19]=500400.782;      cy[19]=468858.435;      cz[19]=936;
cx[20]=500403.656;      cy[20]=468860;  cz[20]=936;
cx[21]=500407.97;       cy[21]=468864.06;       cz[21]=936;
cx[22]=500410;  cy[22]=468863.733;      cz[22]=936;
cx[23]=500411.707;      cy[23]=468863.413;      cz[23]=936;
cx[24]=500418.094;      cy[24]=468863.812;      cz[24]=936;
cx[25]=500420;  cy[25]=468863.487;      cz[25]=936;
cx[26]=500421.843;      cy[26]=468863.687;      cz[26]=936;
cx[27]=500428.198;      cy[27]=468863.604;      cz[27]=936;
cx[28]=500430;  cy[28]=468863.808;      cz[28]=936;
cx[29]=500400;  cy[29]=468873.143;      cz[29]=938;
cx[30]=500402.332;      cy[30]=468875.337;      cz[30]=938;
cx[31]=500405.821;      cy[31]=468871.642;      cz[31]=938;
cx[32]=500410;  cy[32]=468874.101;      cz[32]=938;
cx[33]=500412.935;      cy[33]=468874.129;      cz[33]=938;
cx[34]=500416.447;      cy[34]=468872.894;      cz[34]=938;
cx[35]=500420;  cy[35]=468872.926;      cz[35]=938;
cx[36]=500423.512;      cy[36]=468872.976;      cz[36]=938;
cx[37]=500426.832;      cy[37]=468873.664;      cz[37]=938;
cx[38]=500430;  cy[38]=468873.711;      cz[38]=938;
cx[39]=500400;  cy[39]=468885.61;       cz[39]=940;
cx[40]=500402.239;      cy[40]=468884.479;      cz[40]=940;
```

Figure 3.2.2 Command Line interface

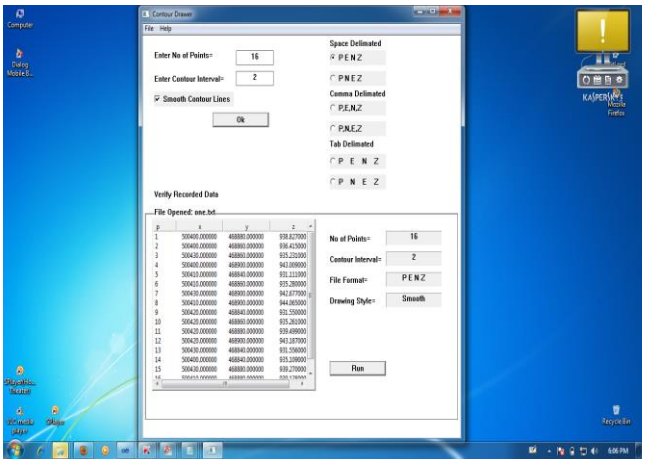
After selecting spot height points from same heights, the distances between one of that point to all other points from the same height group are calculated. DisCal function is used for that. Minimum distance is selected from above distances. Finally positions of two nearest spot height points are assigned to an array. First mentioned spot height point is then deleted from this array system(cx,cy,cz) to another array system for future purposes.

Introduction for GUI

A new graphical user interface for the software is created using Microsoft Visual Studio to manipulate the software easily as well as user friendly. Other advantages are ability of the user to apply mouse options for give commands to software, use dialog options such as open dialog, save dialog to enhance software operations.

Preview of GUI

New graphical user interface can be created using above mentioned files. Figure 3.3.1.4 shows the GUI.



Enter No of Points= 16

Enter Contour Interval= 2

☑ Smooth Contour Lines

● P E N Z

○ P N E Z

○ P,E,N,Z

○ P,N,E,Z

○ P E N Z

○ P N E Z

Verify Recorded Data

File Opened: one.txt

p	x	y	z
1	500400.000000	468880.000000	938.827000
2	500400.000000	468860.000000	936.415000
3	500430.000000	468860.000000	935.231000
4	500400.000000	468900.000000	943.009000
5	500410.000000	468840.000000	931.111000
6	500410.000000	468860.000000	935.280000
7	500430.000000	468900.000000	942.677000
8	500410.000000	468900.000000	944.065000
9	500420.000000	468840.000000	931.550000
10	500420.000000	468860.000000	935.261000
11	500420.000000	468880.000000	939.499000
12	500420.000000	468900.000000	943.187000
13	500430.000000	468840.000000	931.556000
14	500400.000000	468840.000000	935.109000
15	500430.000000	468880.000000	939.270000

No of Points= 16

Contour Interval= 2

File Format= P E N Z

Drawing Style= Smooth

Graphical User Interface

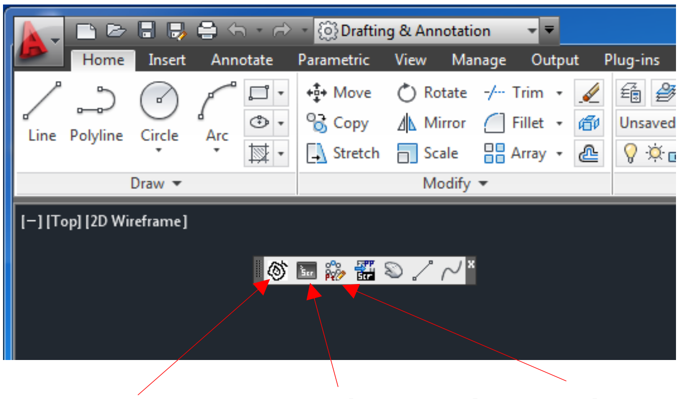
III RESULTS AND DISCUSSION

New Contour Generation Software

This new software can generate contour drawing of input data directly in your AutoCAD. The software was named as DEM CAD. Its tools and other accessories have discussed following.

Quick Access Toolbar

The software provides a graphic toolbar (quick access toolbar) to the user. It allows easy use of the software. This quick access toolbar consist of several icons to draw contours, plotting points, to get the script file of the drawing, pan, zoom etc. Figure 4.1.1 shows the created quick access toolbar in AutoCAD environment.



For Contour Generation Draw the Script File Points Plotting

Quick Access Toolbar

New Menu

The DEM CAD software introduces a new menu called ‘Engineering Survey’ to the existing AutoCAD environment. It can be seen on main menu bar in AutoCAD. This also provides all the functions contained in Quick Access Toolbar. This is the usual way of using DEM CAD. The new menu exists permanently in AutoCAD and user can keep Quick Access Toolbar according to his will. Figure 4.1.2 shows the new menu.

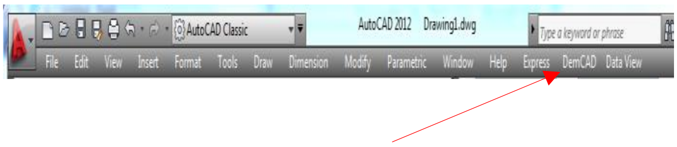
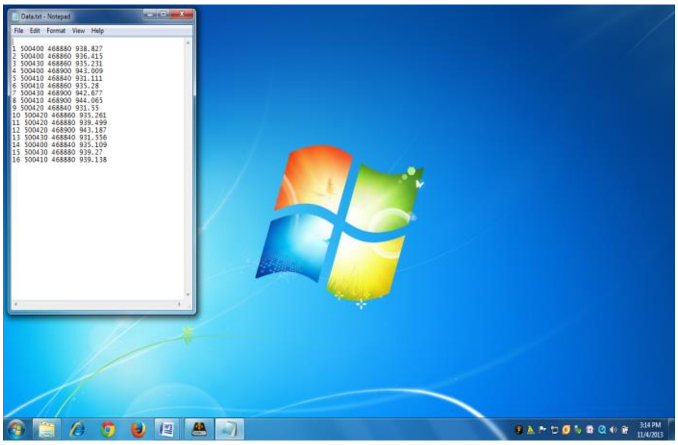


Figure 4.1.2 New Menu DemCAD

Drawing Contours

The steps to be followed by the user to generate contour model for a given dataset and the output model are represented following. Several datasets were used in development stage of the software. It contains three datasets from the Engineering Survey 2012 which was handled by the students of the faculty of Geomatics, Sabaragamuwa university of Sri Lanka and one data set from a topographic survey data collected for a government project. The input data set file is given following.

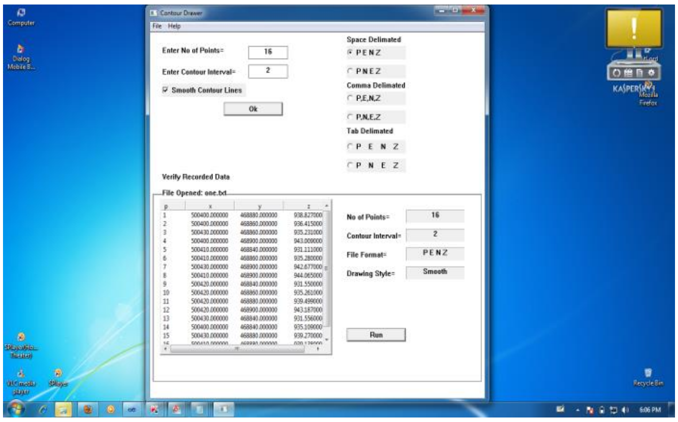


Data.txt - Notepad

```
1 500400 468880 938.827
2 500400 468860 936.415
3 500430 468860 935.231
4 500400 468900 943.009
5 500410 468840 931.111
6 500410 468860 935.28
7 500430 468900 942.677
8 500410 468900 944.065
9 500420 468840 931.55
10 500420 468860 935.261
11 500420 468880 939.499
12 500420 468900 943.187
13 500430 468840 931.556
14 500400 468840 935.109
15 500430 468880 939.27
16 500410 468880 939.138
```

Figure 4.2.1 Input Data File

When user click the contour drawing icon in Quick Access Toolbar or select the Contour Drawer menu in Engineering Survey menu, figure 4.2.2 GUI is viewed.



Contour Drawer

Enter No of Points= 16

Enter Contour Interval= 2

☑ Smooth Contour Lines

Space Delimated
● P E N Z
○ P N E Z

Comma Delimated
○ P,E,N,Z
○ P,N,E,Z

Tab Delimated
○ P E N Z
○ P N E Z

Verify Recorded Data

File Opened: one.txt

p	x	y	z
1	500400.000000	468880.000000	938.827000
2	500400.000000	468860.000000	936.415000
3	500430.000000	468860.000000	935.231000
4	500400.000000	468900.000000	943.009000
5	500410.000000	468840.000000	931.111000
6	500410.000000	468860.000000	935.280000
7	500430.000000	468900.000000	942.677000
8	500410.000000	468900.000000	944.065000
9	500420.000000	468840.000000	931.550000
10	500420.000000	468860.000000	935.261000
11	500420.000000	468880.000000	939.499000
12	500420.000000	468900.000000	943.187000
13	500430.000000	468840.000000	931.556000
14	500400.000000	468840.000000	935.109000
15	500430.000000	468880.000000	939.270000

No of Points= 16

Contour Interval= 2

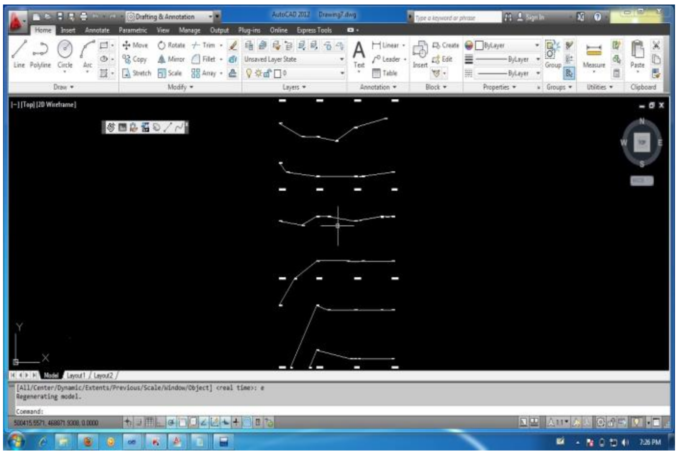
File Format= P E N Z

Drawing Style= Smooth

Figure 4.2.2 Contour Drawer

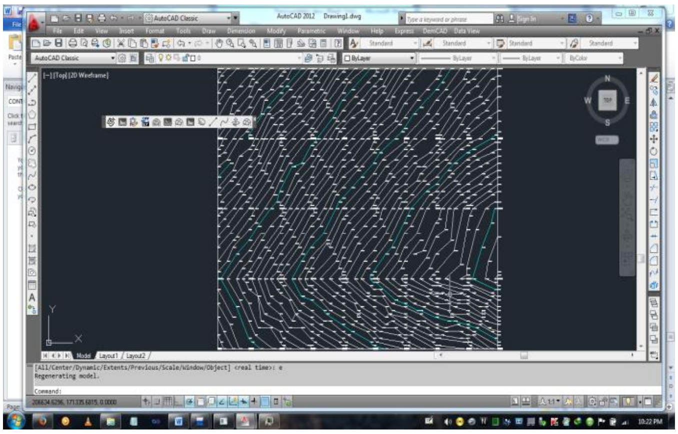
This GUI provides the ability to open the data file and set the data order, number of points, required contour interval. It also shows the preview of the data when select the data file.

After that the user can click Run button, then generated contour model can be seen in AutoCAD environment. The process is programmed to show the contour model by zoom and pan automatically. So the user can see the model directly. The user can easily create topographic plans and view the model in AutoCAD 3D environment without data exporting or importing. Figure 4.2.3 shows the generated contour model using the new software for above dataset.



```
[All/Center/Dynamic/Extents/Previous/Scale/Window/Object] <real time>: e
Regenerating model.
Command:
```

Figure 4.2.3 Contour Model



```
[All/Center/Dynamic/Extents/Previous/Scale/Window/Object] <real time>: e
Regenerating model.
Command:
```

Figure 4.4.3 Contour with 50 points

Contour model generated in AutoCAD Civil3D was used as a reference model and both the models were very similar.

References

[1] Ou, S.C., Sung, W.T., Hsiao, S.J. and Fan, K.C., Interactive web-based training tool for CAD in a virtual environment. Comput. Appl. Eng. Educ. Vol. 10, pp. 182-193, 2003.

[2] Badan, I. and Oancea, G., Software Tool Used for Holes Modeling in AutoCAD Environment. In: Proceedings of the Latest Trends on Engineering Mechanics, Structures, Engineering Geology, pp. 152-156, 2010.I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[3] John W. Horch (2005). "Two Orientations On How To Work With Objects." In: IEEE Software. vol. 12, no. 2, pp. 117–118, Mar., 1995.