Development of a machine tool auxiliary machining system based on android phone

C M Zhang¹, L S Zhang², T Zhang³ and S T Zhang⁴

¹ Zhuhai College of Jilin University, Zhuhai 519000, China
² College of Mechanical Science and Engineering, Jilin University, Changchun 130022, China
³ College of Mechanical Science and Engineering, Jilin University, Changchun 130022, China
⁴ Easson Measurement Technology Ltd, Guangdong Zhuhai, 519000, China

Corresponding author and e-mail: C M Zhang, chuanminzhang@163.com

Abstract. This paper presents an Android-based system for assisting manual machine tool machining. We developed some programs used on the PC for reading data from DXF (Drawing Exchange Format) files with C# language. These programs extracted the data from DXF files, formed custom EXF files and uploaded these files to the file server. The EXF format files were used as data source for an Android application. This application got files from the file server and displayed two dimensional engineering drawings by using OpenGL ES. Meanwhile, it was connected with the digital display device of the grating scale through Bluetooth in order to realize the real-time tracking of tool path and improve machining efficiency. Experiments proved this system can accurately display coordinates of the tool in real time and improve machining efficiency.

1. Introduction
At present, in machinery manufacturing industry, the penetration rate of CNC machine tools is increasing. But manual machine tools are still in use, especially in the small and medium enterprise. In the foreseeable future, it still occupies a pivotal position in the manufacturing industry. So it’s necessary to make manual machine tools easy to operate and improve machining efficiency.

The impact of CAD on machinery manufacturing industry is no doubt huge. Now, CAD is very mature and popular on the computer terminal. The popularity of smart phones has a great impact on people’s daily life and the manufacturing industry. Smart phones have more functions and can do the work done by the computer in the past [1]. Many Android applications realize functions of CAD software. The existing applications can use a variety of file formats such as DWG, DXF and so on. They can be used in construction, machinery, heating and ventilation, electrical engineering, interior design and other industries. But all of them can’t realize real-time tracking of tool path.

This paper presents an Auxiliary Machining System that can help people view engineering drawings and communicate with the digital display device of the grating scale applied to machine tool.
2. Related studies of the system

2.1. OpenGL ES
To introduce OpenGL ES has to mention the basic concepts of OpenGL (Open Graphics Library). OpenGL is a professional graphic application interface for computers and other devices. It defines a cross programming language, an application interface across the runtime platform and standard specification for processing two-dimensional images and three-dimensional images in applications [2]. OpenGL is a hardware-independent software interface that can be ported between different platforms. Therefore, software that supports OpenGL is very portable and can be used in a wide range of applications. Since OpenGL is the underlying graphics library for 3D graphics, no geometric entity primitives are provided and cannot be used directly to describe the scene. However, the DXF and 3DS model files produced by AutoCAD, 3DS and other 3D graphics design software can be easily converted into an OpenGL vertex array through some conversion programs [3].

OpenGL ES is a free, fully functional 2D and 3D image application program interface for embedded systems, including controllers, smart phones, electrical appliances, and vehicle mounted devices. It consists of a series of defined OpenGL subset of the specification, and creates a flexible and powerful low-level interface between the application and the graphic accelerator. The interface is made up of a series of processing processes and function functions. These interface processes and functions can be used to specify related objects and operations in the process of dealing with high quality pictures [2].

OpenGL ES inherits many OpenGL features and has the advantage of openness, enabling developers to develop applications on mobile devices more easily.

2.2. DXF (drawing exchange format)
DXF (Drawing Exchange Format) is a representation of the labeled data of all the information contained in AutoCAD graphics files. It is ACSII or binary file format of AutoCAD drawing file that can be used to input, output and exchange graphics files between AutoCAD or other graphics applications. In this paper, reading conversion of DXF file is in ASCII format [4].

DXF file consists of multiple sections [5]:
- HEADER section: This section contains basic information related to a graphical file, which consists of an AutoCAD database version number, various settings in the storage disk, and many system variables.
- CLASSES section: This section contains information of the classes defined by the application, whose instances appear in the BLOCKS, ENTITIES, and OBJECTS segments of the database.
- TABLES section: It contains the definitions of the following tables: APPID, BLOCK_RECORD, DIMSTYLE, LAYER, STYLE, LTYPE, UCS, VIEW and VPORT. The number of table items for each table is variable.
- BLOCKS section: This section contains all the blocks defined in the AutoCAD, including the definition of the block and the graphic elements that make up each block in the graph.
- ENTITIES section: This section contains all the graphic objects that appear in the graph, including the reference of the block. This section can’t be ignored in the DXF file.
- OBJECTS section: This section contains non-graphic objects in the graph, which are stored in this section except for the diagram element, the symbol table record and all objects other than the symbol table.
- THUMBNAILIMAGE section: This section contains a graphical preview of the image data.

2.3. Bluetooth
Bluetooth is a universal short-range low-power radio protocol working in the unlicensed industrial, scientific, and medical frequency band. The operating range of Bluetooth is from 10m to 100m. Bluetooth can transmit voice and data without applying permission to work in the 2.4–2.485GHz band. It has many properties such as low cost, low power, low radiation and so on [6].
The Android platform supports the Bluetooth protocol stack, so it supports transmitting data between two Bluetooth devices. The Android Application Framework Layer provides APIs that allow Bluetooth to connect. These APIs enable Bluetooth applications to connect wirelessly and establish end-to-end connections. Using Bluetooth API, the following functions can be implemented.

- Search for other Bluetooth devices
- Query the device that matches the local Bluetooth adapter.
- Transmit data between two different Bluetooth devices.
- Manage multiple Bluetooth connections.

3. **Design and implement**

The whole system is mainly composed of four parts: PC terminal for DXF files reading conversion and uploading, the file server for data storage, Android phone application and the digital display device of grating scale for communicating through Bluetooth. Figure 1 (a) illustrates the structure of this system. Figure 1 (b) shows the grating sale and the digital display device.

**Figure 1 (a).** The structure of this system.

**Figure 1 (b).** The structure of this system.
3.1. PC terminal
PC terminal mainly realizes the following functions:

- Read DXF file.
- Extract the data required by Android application to reproduce the drawing and form EXF file based on DXF file.
- Upload EXF file to the file server.
- Set the basic information of the user, which is used for the Android application login function.
- Show the basic information of DXF file after parsing the file.

3.1.1. Reading DXF file. OpenGL ES needs the geometry data information and attribute data of the graphic file. Because these data are contained in the ENTITIES section, OpenGL ES simply focuses on the ENTITIES section of the DXF file. The ENTITIES section records the data of the layer name, line, color, vertex and so on.

The open source code library named netDxf is used when reading DXF files. It can fully read DXF files. The reason for choosing netDxf is that netDxf is not dependent on any CAD software, and users don’t need to install any CAD software when system to realize the reading operation of DXF files. The call to netDxf to achieve reading DXF files function requires the following four steps:

- Step 1: Add netDxf to the newly built C# project and then compile the project.
- Step 2: After compiling, define a class named DXF_Open to call netDxf’s Blocks, Collections, Entities, Header, Objects, Tables and DxfDocument classes so as to complete reading all the data information of DXF files.
- Step 3: The method named DxfVersion of Header class in netDxf is called to detect the version information of DXF file, and the method named Load in DxfDocument class is called to read DXF file in accordance with the specification.
- Step 4: Show the basic information of DXF file and return a DxfDocument object, which includes all information of DXF file and is used as data source of DXF file conversion.

Figure 2 shows the workflow of reading DXF file.

![Figure 2. Read DXF file.](image-url)
### 3.1.2. DXF file conversion

The necessary data are extracted from DXF file to form a custom readable file called EXF file. Take the LINE in DXF file as an example, it is represented in EXF file as follows:

```
# li
rgb,255,255,255
layer_name,1
sxy,560.046851731832,324.638415162713
exy,560.30152182155,324.383745072994
line_weight,-3
DottedLine,F
%
```

“#” marks the beginning of the primitive’s information file and “%” marks the end of the primitive’s information file; “li” indicates that the primitive is LINE; “rgb” indicates the color of LINE; “layer_name” represents the name of the layer; “sxy” indicates the coordinates of the linear start point; “exy” indicates the coordinates of the linear end point; “line_weight” indicates the width of LINE. Because there is dot line in the annotation, “DottedLine” indicates whether LINE is a dot line.

We have made similar conversion to other primitives. Then we upload EXF files to the file server. Figure 3 shows the workflow.

![Figure 3. DXF file conversion and upload.](attachment:image.png)

### 3.2. Android phone application

The Android phone application mainly implements the following functions:

- Users can download EXF files after login.
- Connect with the digital display device of grating scale through Bluetooth.
- Read the file, redraw the drawing and display 2D drawing after downloading the EXF file.
- Display the coordinates of the tool in real time.

Android application draws drawings through OpenGL ES. OpenGL ES uses vertices to draw all primitives. In this way, the calculator and vertex-by-vertex operation can be operated on each vertex, then the primitives are rasterized to form fragment. Finally, a series of operations are performed on all the fragments, and then send the fragments to the frame buffer to realize graphics display.
Figure 4. shows the basic workflow of OpenGL ES [7]:

![OpenGL ES programmable pipeline](image)

The entities in EXF files correspond to that in DXF files one by one. In the program, the corresponding classes are defined for the entities of EXF files. These classes contain all information of these entities.

Define OpenGL_exf class. The main function of this class is to define the basic data parameters in the drawing of each entity. For example, Line entity is defined as follows:

```java
public float[] fline = new float[Main.Instance.exf_main.listLine.size()*4];
public FloatBuffer mline;
public int mlineCount = 0;
```

“fline” is an array of vertex indices. “mline” is that vertex data cache and the true vertex data source and “mlineCount” represent the number of LINE entity in the engineering drawing.

Define OpenGL_init class to initialize data and provide data for drawing the drawing.

Define OpenGL_main class to implement android.Opengl.glsurfaceview.renderer interface and provides Android activity for drawing the drawing.

Define OpenGL_draw class, which completes the drawing of entity. For example, the realization process of LINE entity is as follows:

```java
gl.glVertexPointer(2, GL10.GL_FLOAT, 0, op_exf.mline);
gl.glDrawArrays(GL10.GL_LINES,0,Main.Instance.exf_main.listLine.size()*2);
```

Android application mainly use these classes to draw engineering drawings, and the drawing process is shown in Figure 5.
Taking an engineering drawing with an opening of 15 degrees ellipse as an example, the drawing on the PC are compared with that on the Android phone. As can be seen in Figure 6, the Android application can display the engineering drawing completely.

![Drawing Process Diagram](image)

**Figure 5.** The drawing process.

![Drawing on PC](image)

**Figure 6(a).** The drawing on the PC.
3.3. Bluetooth module

We have built-in Bluetooth module in the digital display device of grating scale to communicate with mobile phone. After the drawings are displayed on Android phone, users can locate the coordinates of some points in the drawings according to the requirements, and send the coordinates of these points to the digital display device. In the process of machining, the digital display device sends the tool coordinates measured by the grating scale in real time, and Android phone receives and processes the data.

This Serial Bluetooth module is easy to use. It designs for transparent wireless serial connection setup. Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) GFSK Module with complete 2.4GHz radio transceiver and baseband. The maximum transmit power of the module is 4dBm, the receiving sensitivity is - 85dBm, and the board-mounted PCB antenna can realize 10m range communication. It uses CSR BC417 chip and supports AT commands. Connect the + pin to 3.3v and - pin to GND, TX and RX pins connect to the serial pins.
4. Experimental results
The experiment mainly verifies that when the digital display device of grating scale and Android application software are connected through Bluetooth, the corresponding operation steps can be used to assist the machine machining to realize the visualization of tool moving path.

In Figure 8 (a), the yellow cross line in the center of drawing represents the tool center point. The green dot inside blue circle in the top left corner is the start dot of machine processing and it is set to be origin. The points 1, 2, 3, 4, 5 and 6 are set target points. After planning, click the "digital display" icon button to send information of these plans to the digital display device.

![Figure 8 (a). The drawing.](image-url)
Figure 8 (b). Digital display device (ES-17).

We recorded the coordinates of tool at a certain time when the tool was moving from its origin to D1 and D2 and the coordinates of tool when it moved to the target points D1 and D2. For D3, D4, D5 and D6, only the coordinates of tool at the target points were recorded because of drilling operation. As can be seen, data in Table 1 show that the digital display device displays the coordinates of tool, which are relative to that of target point and the application displays the coordinates of tool relative to the origin. When the tool moves to the target point, the coordinates displayed by the digital display device are within error range, which is in line with the expected effect. Operator only needs to pay attention to the coordinates displayed by the digital display device. The calculation steps are eliminated, and the visual machining path are realized.

Table 1. Coordinates of the tool.

|    | Target point | Digital display device | Android phone application |
|----|--------------|------------------------|---------------------------|
| P1 | D1(0.000, -38.082) | (0.000, 19.140)        | (0.000, -18.849)          |
| P2 | D1(0.000, -38.082) | (0.000, -0.002)        | (0.000, -38.001)          |
| P3 | D2(83.998, -38.082) | (-40.177, -0.002)      | (43.821, -38.001)         |
| P4 | D2(83.998, -38.082) | (0.000, -0.002)        | (83.998, -38.001)         |
| P5 | D3(18.314, -10.190) | (-0.002, 0.000)        | (18.312, -10.107)         |
| P6 | D4(18.314, -27.974) | (0.002, 0.000)         | (18.316, -27.891)         |
| P7 | D5(41.153, -28.489) | (-0.002, 0.001)        | (41.151, -28.405)         |
| P8 | D6(41.153, -9.675)  | (0.001, 0.001)         | (41.154, -9.591)          |

5. Conclusions
This system helps newcomers quickly operate manual machine tools, reducing training cycle of enterprises, so as to improve profits for the enterprise. The mobile phone and the digital display device
display the relative coordinates very intuitively, avoiding the occurrence of waste due to calculation errors and reducing the scrap rate.

Acknowledgement

The authors of this study are grateful for the support of Zhuhai College of Jilin University “three levels” talent construction project fund.

References

[1] Qian J X 2016 The development and research of NC machining simulation system based on the mobile terminals[D]. Huazhong University of Science and Technology

[2] Qi M, Zhang R J 2014 The similarities and differences between OpenGL es1.0 and OpenGL es2.0 in the development process J. Electronic Production (21): 67-68.

[3] Liu Q, Chi X S 2008 Similarities and differences between OpenGL and OpenGL ES J. Journal of Liaoning Technical University (Natural Science), (2):261-262.

[4] Zhang Y C, Yue J P, Wang H Q 2009 DXF Document Reading and Interaction Control Based on VB6.0 and OpenGL J. Computer Era, (09):39-41

[5] Wang Z B 2007 Read and XML conversion based on DXF file format J. New science, (06): 240-241

[6] Wu T, H.J.-d., Gao C J, et al. 2010 Wireless Monitoring System Based on Bluetooth smart phones, in 2010 International Conference on Networking and Digital Society p. 648-651.

[7] Chang J 2007 OpenGL Graphics Generation Algorithms for Aircraft Platform[D]. Shanghai Jiao Tong University