**ST.XAVIER’S COLLEGE**

MAITIGHAR, KATHMANDU



**Computer Graphics Assignment #7**

**Translate, Rotate and Scale an image**

**Submitted By:**

Bikash Paneru

013BSCCSIT012

**Submitted to:**

|  |  |
| --- | --- |
| Er. Anil K. Sah  Lecturer, Department of Computer Science |  |

**Date of Submission:** 1 September, 2015

**STATEMENT**

Write a program to Translate, Rotate and Scale an image.

­

**SOURCE CODE**

//---------------------------------------------------------------------------

#include <vcl\vcl.h>

#pragma hdrstop

#include <Math.h>

#include "balltrans.h"

//---------------------------------------------------------------------------

#pragma resource "\*.dfm"

TForm1 \*Form1;

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

void translate(TCanvas \* source, TCanvas \* destination,int x,int y,int tX,int tY) {

destination->Pixels[x+tX][y+tY] = source->Pixels[x][y];

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::translateButtonClick(TObject \*Sender)

{

//Get the number of pixels to translate in each direction

int tX = StrToInt(translateX->Text);

int tY = StrToInt(translateY->Text);

//Load references to the canvases to take pixels and put pixels in

TCanvas \* source = imageSrc->Canvas;

TCanvas \* destination = imageDest->Canvas;

destination->FillRect(ClientRect);

for(int x = 0; x<(imageSrc->Width); ++x) {

for(int y = 0; y<(imageSrc->Height); ++y) {

translate(source,destination,x,y,tX,tY);

}

}

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::rotateButtonClick(TObject \*Sender)

{

float angle = StrToFloat(rotationAngle->Text)\*(3.14/180);

//Load references to the canvases to take pixels and put pixels in

TCanvas \* source = imageSrc->Canvas;

TCanvas \* destination = imageDest->Canvas;

int ptX = StrToInt(pointX->Text);

int ptY = StrToInt(pointY->Text);

destination->FillRect(ClientRect);

for(int x = 0; x<(imageSrc->Width); ++x) {

for(int y = 0; y<(imageSrc->Height); ++y) {

int \_x = x-ptX;

int \_y = y-ptY;

\_x = x \* cos(angle) - y \* sin(angle);

\_y = y \* cos(angle) + x \* sin(angle);

\_x+=ptX;

\_y+=ptY;

destination->Pixels[\_x][\_y] = source->Pixels[x][y];

}

}

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::scaleButtonClick(TObject \*Sender)

{

//Get the number of pixels to scale in each direction

float sX = StrToFloat(scaleX->Text);

float sY = StrToFloat(scaleY->Text);

//Load references to the canvases to take pixels and put pixels in

TCanvas \* source = imageSrc->Canvas;

TCanvas \* destination = imageDest->Canvas;

destination->FillRect(ClientRect);

int ptX = StrToInt(pointX->Text);

int ptY = StrToInt(pointY->Text);

for(int x = 0; x<(imageSrc->Width); ++x) {

for(int y = 0; y<(imageSrc->Height); ++y) {

int \_x = x-ptX;

int \_y = y-ptY;

\_x = \_x\*sX;

\_y = \_y\*sY;

\_x+=ptX;

\_y+=ptY;

destination->Pixels[\_x][\_y] = source->Pixels[x][y];

}

}

}

//---------------------------------------------------------------------------

**OUTPUT:**

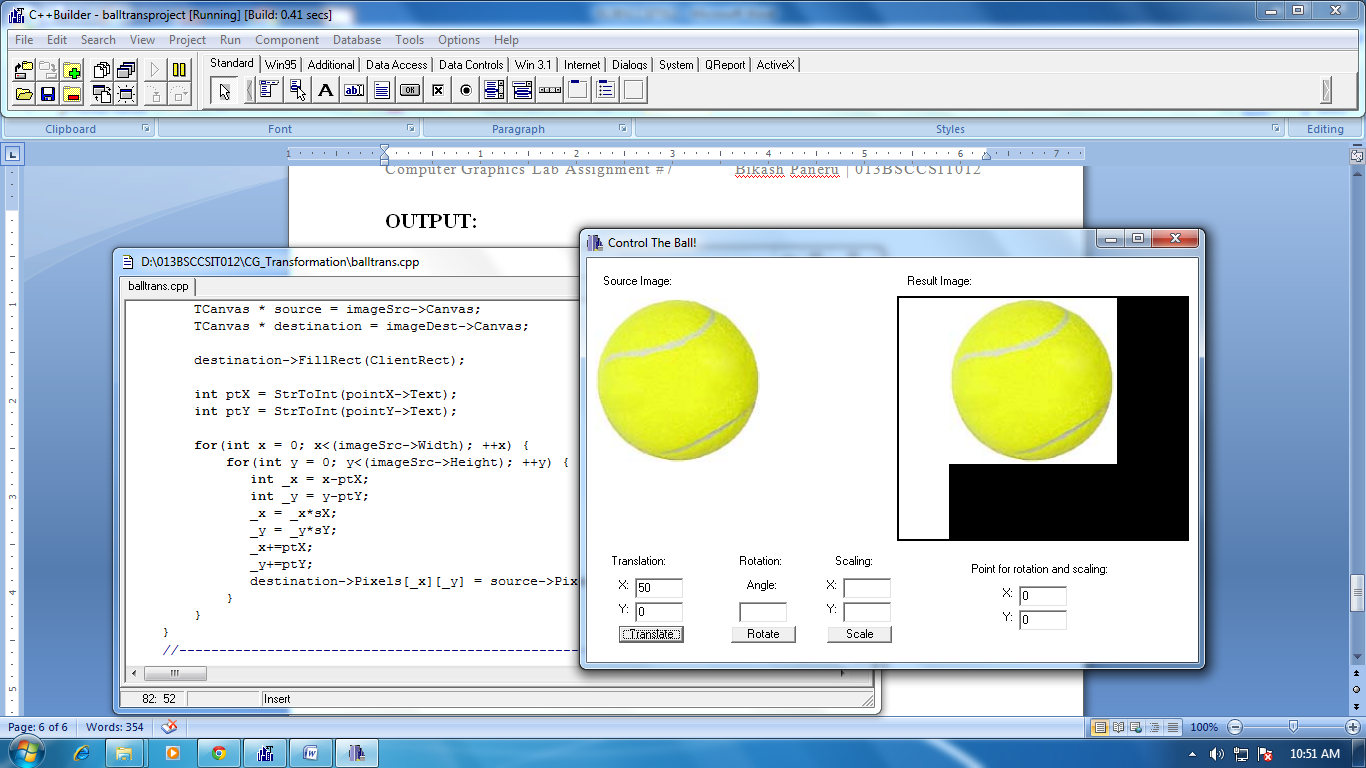
****

Figure Translation

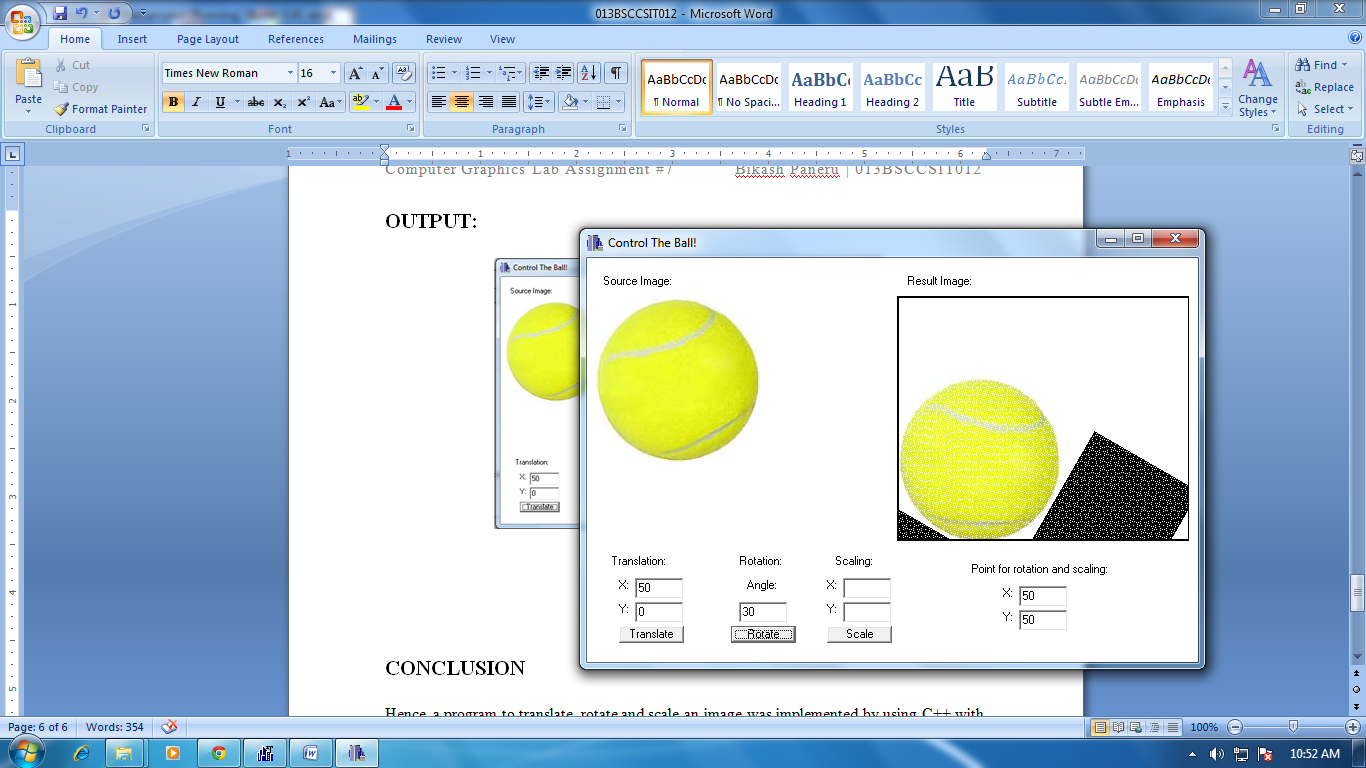
****

Figure Rotation

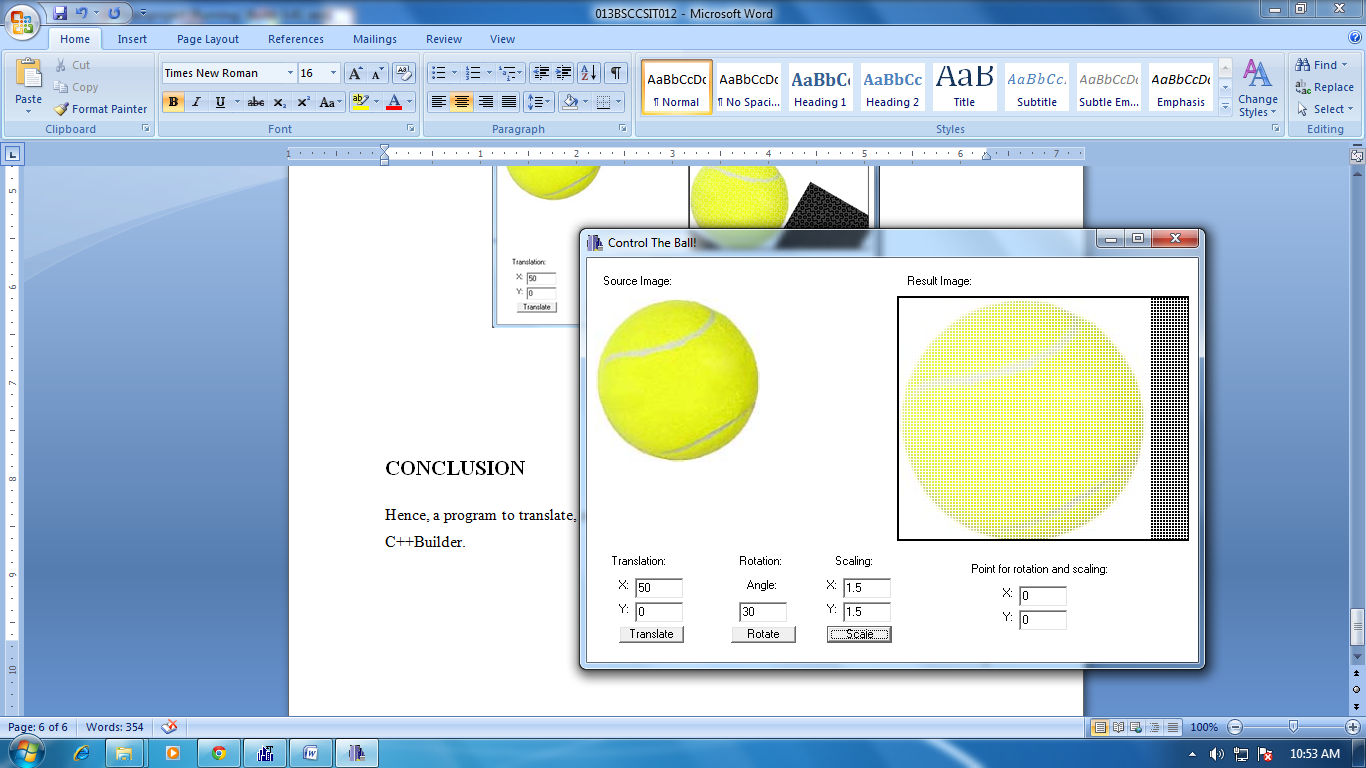
****

Figure Scaling

**CONCLUSION**

Hence, a program to translate, rotate and scale an image was implemented by using C++ with C++Builder.