

110 年度 教學實踐研究計畫 結案報告

計畫編號：PEE1101040

計畫名稱：使用開源軟體 Emgu CV 實作自動光學檢測

方信普
南臺科技大學電子工程系

本講義為 110 年度教學實踐研究計畫的成果

計畫編號：PEE1101040

計畫名稱：使用開源軟體 Emgu CV 實作自動光學檢測

目錄

Lesson 1	C# 簡介 -----	1
Lesson 2	Visual C# 的初步-----	12
Lesson 3	Emgucv Load and Show Image -----	19
Lesson 4	Emgucv Image Rotate and Resize-----	20
Lesson 5	Emgucv Image Dilate and Erode -----	22
Lesson 6	Set Pixel Speed-----	24
Lesson 7	二值化 -----	30
Lesson 8	讀圖、顯示滑鼠之座標、單一 pixel 之 RGB -----	49
Lesson 9	Rotation、Add and Subtract 的應用 -----	59
Lesson 10	Histogram-----	79
Lesson 11	找三角形的頂點-----	87
Lesson 12	尋找圖形之邊緣與量測 -----	94
Lesson 13	Crop and Match-----	107
Lesson 14	Mask 與 Blob 的應用-----	124
Lesson 15	在不同的 Form 之間傳遞參數-----	159
Lesson 16	Tesseract 與 OCR-----	164
Lesson 17	LCD 玻璃刮痕瑕疵檢測與玻璃瓶口檢測-----	170
Lesson 18	攝影機的使用與 Read Video File-----	181
附錄(一) 期末專題-----		197

摘要

自動光學檢測 (Automated Optical Inspection, AOI) 是結合光學、影像處理、與資訊科技，對生產過程的瑕疵進行檢測的技術。AOI 之檢測可靠性高，標準統一，可及時回饋問題，增加生產效率，降低人工成本。根據由田新技股份有限公司的資料，2024 年 PCB、半導體、面板三產業的 AOI 市場可達 90.94 億美元。IEK 產業情報網表示 2020 年-2025 年，AOI 的年複合成長率約 16%。

本課程「自動光學檢測實務」的目的為教導學生自動化光學檢測的程式實作，增加同學就業與實習的學識能力。

業界實作自動化光學檢測常使用商業用的套裝軟體，如 Halcon、Matrox 的 MIL、Euresys 的 Open eVision 等。但其價錢昂貴，若欲安裝在一間 60 台電腦的教室，其預估花費新台幣 500 至 1500 萬元。再加上缺乏合適的教材，因此學校鮮少有針對自動光學檢測開實務性課程。

為克服昂貴商用軟體與教材缺乏的困難，本計畫的目的為編撰講義，以手把手的方式，教導學生使用開源軟體 Emgu CV 實作自動光學檢測。

Abstract

Automated Optical Inspection (AOI) is a technology that combines optics, image processing, and information technology to detect defects in the production process. AOI can increase production efficiency and reduce labor costs due to its high reliability and unified standard. According to the information of Utechzone Co., Ltd., the AOI market of PCB, Semiconductor and Panel Industries will reach 9.094 billion US dollars in 2024. IEK Industry Intelligence Network reported the compound annual growth rate of AOI will be about 16% from 2020 to 2025.

The purpose of this course "Automatic Optical Inspection Practice" is to teach students the program implementation of automated optical inspection, and to increase students' academic ability in employment and internship.

Commercial software packages such as Halcon, Matrox's MIL, Euresys' Open eVision, etc. are expensive. It will cost NT\$5-15 million to install a 60-computer classroom. Muchmore, the course is lack of teaching materials. Therefore, schools rarely offer practice courses for automatic optical inspection.

In order to overcome the difficulties of teaching materials and expensive commercial software, the purpose of this project is to compile lecture to teach students how to use the open source software Emgu CV to implement automatic optical inspection.

Lesson 1

C# 簡介

Lesson 1.1 “Hello World”

程式 HelloWorld

```
using System;

namespace HelloWorld
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
        }
    }
}
```

Lesson 1.2

Console read and write

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace CS_Lesson_2
{
    class Program
    {
        static void Main(string[] args)
        {
            string Name;
            int Age;
```

```

        Console.WriteLine(" Hello there!");
        Console.Write("What's your name? ");
        Name=Console.ReadLine();

        Console.Write("How old are you? ");
        Age = int.Parse(Console.ReadLine());

        Console.WriteLine(" You are {0} and {1} years old.", Name, Age);
        Console.Read();
    }
}
}

```

練習 1.2：詢問: What is your occupation? 並輸出 You are a/an ...

Lesson 1.3

if, else, and elseif

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace CS_Lesson_3
{
    class Program
    {
        static void Main(string[] args)
        {
            int num1, num2;
            Console.Write("請輸入第一個整數(num1) : ");
            num1 = int.Parse(Console.ReadLine());
            Console.Write("請輸入第二個整數(num2) : ");
            num2 = int.Parse(Console.ReadLine());
            if (num1 == num2) // 判斷 num1 是否等於 num2
                Console.WriteLine("{0} = {1}", num1, num2);
            else if (num1 > num2) // 判斷 num1 是否大於 num2
                Console.WriteLine("{0} > {1}", num1, num2);
            else // 判斷 num1 是否小於

```

```

num2
        Console.WriteLine("{0} < {1}", num1, num2);
    Console.Read();
    }
}
}

```

練習 1.3：由鍵盤輸入一整數，並判斷它是奇數或偶數 (取餘數 使用%)

Lesson 1.4 switch and case

```

string month;
    Console.WriteLine("=== 請輸入現在的月份: ");
    month = Console.ReadLine(); // 輸入月份
    switch (month)
    {
        case "3": // 判斷 month 是否為"3" ~ "5"
        case "4":
        case "5":
            Console.WriteLine("\n ... 現在是春天 ...");
            break;
        case "6": // 判斷 month 是否為"6" ~ "8"
        case "7":
        case "8":
            Console.WriteLine("\n ... 現在是夏天 ...");
            break;
        case "9": // 判斷 month 是否為"9" ~ "11"
        case "10":
        case "11":
            Console.WriteLine("\n ... 現在是秋天 ...");
            break;
        case "12": // 判斷 month 是否為"12"、"1"、"2"
        case "1":
        case "2":
            Console.WriteLine("\n ... 現在是冬天 ...");
            break;
        default: // month 為其他值
            Console.WriteLine("\n ... 輸入值超出範圍 ....");
    }

```

```
        break;
    }
    Console.Read();
```

練習: 輸入代碼: 1 學生 、2 教師 、3 職員 、其他 校外人士

(Solution : A5 習題)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            string id;
            Console.Write("身分: ");
            id = Console.ReadLine();
            switch (id)
            {
                case "1":

                    Console.WriteLine("\n 你是學生");
                    break;
                case "2":

                    Console.WriteLine("\n 你是教師");
                    break;
                case "3":

                    Console.WriteLine(" \n 你是職員");
                    break;
```

```

        default:
            Console.WriteLine(" \n 校外人士");
            break;
    }
    Console.Read();
}
}
}
}
}

```

Lesson 1.5

Try, Catch, and Finally

Ex 1:

```

int i, k, p;

    i = 5;
    k = 0;
    p = i / k;

Console.WriteLine("The program is over.");
    Console.Read();

```

Ex 2:

```

int i, k, p;

    i = 5;
    k = 0;
    try
    {
        p = i / k;           // 將可能發生例外的程式碼置於 try
    }

    catch (Exception ex)    // 當發生的例外符合 Exception 時會執行此
    {
        Console.WriteLine("發生例外");
    }

```

```

        Console.WriteLine(ex.Message);
    }

    finally // 無論是否發生例外皆會執行 finally 區塊
    {
        Console.WriteLine(".... 沒中斷繼續 ....");
    }

    for (i=0;i<10;i=i+1)
    {
        Console.WriteLine(i);
    }

    Console.Read();

```

Ex 3:

```

int i, k, p;

i = 5;
k = 0;
try
{
    p = i / k; // 將可能發生例外的程式碼置於 try 區塊
}
catch (DivideByZeroException ex)
{
    Console.WriteLine("例外訊息：{0}", ex.Message);
    Console.WriteLine("發生例外的函式：{0}", ex.StackTrace);
    Console.WriteLine("發生例外的物件：{0}", ex.Source);
    Console.WriteLine("發生例外的物件型別：{0}", ex.GetType());
    Console.WriteLine("發生例外的文字說明：{0}", ex.ToString());
}
finally // 無論是否發生例外，皆會執行 finally 區塊中的程式碼
{
    Console.WriteLine(".... 結束程式執行!! ....");
}

Console.Read();

```

練習:

被除數: 100

除數: 從 10 到 -10

Console 顯示 被除數 除以 除數 商 餘數

例外時顯示例外訊息

Solution : A7 習題

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ConsoleApplication2
{
    class Program
    {
        static void Main(string[] args)
        {
            int i, k, p, r;

            k = 100;
            for (i = 10; i > -11; i--)
            {
                try
                {
                    p = k / i;
                    r = k % i;
                    Console.WriteLine( "{0} 除以 {1} 商{2} 餘數
{3}",k,i,p,r); // 將可能發生例外的程式碼置於 try 區塊
                }

                catch (Exception ex)    // 當發生的例外符合 Exception 時會執行此處
                {
                    Console.Write("{0} 除以 {1}", k, i);
                    Console.Write(" 發生例外");
                    Console.WriteLine(ex.Message);
                }
            }
        }
    }
}
```

```

        finally // 無論是否發生例外皆會執行
        finally 區塊
        {
            // Console.WriteLine(p);
        }
    }

    Console.Read();

}
}
}

```

Lesson 1.6

物件__以亂數為例

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace CS_Lesson_9
{
    class Program
    {
        static void Main(string[] args)
        {
            Random irandom = new Random();

            for (int k = 1; k <= 6; k++)
            {
                Console.Write(" {0}", irandom.Next(1, 21)); //六個亂數 1-20
            }
            Console.WriteLine();
            Console.Read();
        }
    }
}

```

```
    }  
    }  
}
```

練習: 以亂數產生一 1-20 數目

輸入猜測

若太大 輸出"太大 再猜一次"

若太小 輸出"太小 再猜一次"

猜中 輸出" Bingo 你是神"

五次機會

完全沒猜中 輸出"不玩了，你猜的數目分別是.....平均....."

Soution : A9 習題

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
  
namespace ConsoleApplication3  
{  
    class Program  
    {  
        static void Main(string[] args)  
        {  
            Random iRand = new Random();  
            int target = iRand.Next(1, 21);  
            int [] guess=new int[5];  
            int sum=0;  
  
            int num=0;  
  
            for ( int k = 0; k < 5; k++)  
            {
```

```

        Console.WriteLine("輸入猜測 1~20 : " );
        guess[k] =int.Parse(Console.ReadLine());
        sum = sum + guess[k];
        if (guess[k] < target)
        {
            Console.WriteLine("太小");
            num = num + 1;
        }
        else if (guess[k] > target)
        {
            Console.WriteLine("太大");
            num = num + 1;
        }
        else
        {
            Console.WriteLine("Bingo");
            break;
        }
    }

    Console.WriteLine();

    if (num==5)
    {
        Console.WriteLine("完全沒猜中 不玩了,你猜的數目分別是…");

        for (int i=0;i<5;i++)
        {
            Console.Write(" {0} ",guess[i]);
        }

        Console.WriteLine();
        Console.WriteLine("平均");
        Console.WriteLine((double)sum / 5);
    }

```

```

        Console.Read();
    }
}

```

Lesson 1.7

副程式

```

static double CalculateArea(int r)
{
    double area = r * r * Math.PI;
    return area;
}

static void Main()
{
    int radius = 5;
    double result = CalculateArea(radius);
    Console.WriteLine("The area is {0:0.00}", result);
}

```

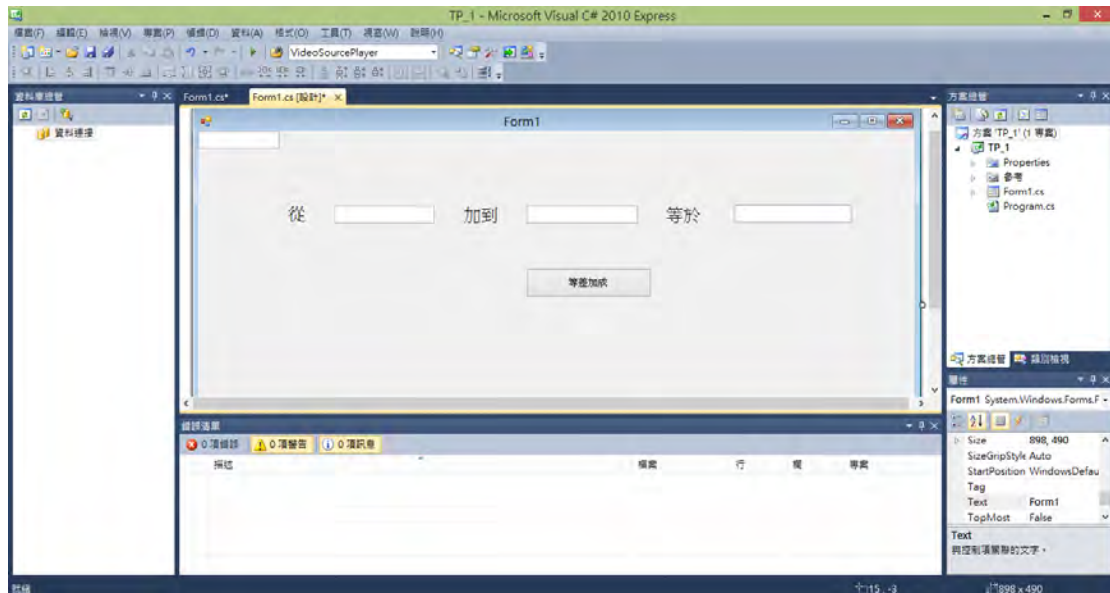
練習

1. 寫二個副程式，分別計算長方形之面積與周長
 2. 寫一個副程式，計算陣列的平均值、與標準差
-

Lesson 2

Visual C# 的初步

等差級數相加 (程式 TP1)



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

//This program is OK    //從 a 一直加到 b
//
//Practice 等差相乘,不同方法的輸出輸入
```

```
namespace TP_1
{
    public partial class Form1 : Form
    {
        public Form1()
```

```

    {
        InitializeComponent();
    }

private void button1_Click(object sender, EventArgs e)
{
    int anssum, ii, a, b;
    a = int.Parse(textBox1.Text);
    b = int.Parse(textBox2.Text);
    anssum = 0;
    for (ii = a; ii <= b; ii++)
    {
        anssum = anssum + ii;
    }
    textBox3.Text = anssum.ToString();
}

private void textBox5_TextChanged(object sender, EventArgs e)
{
}

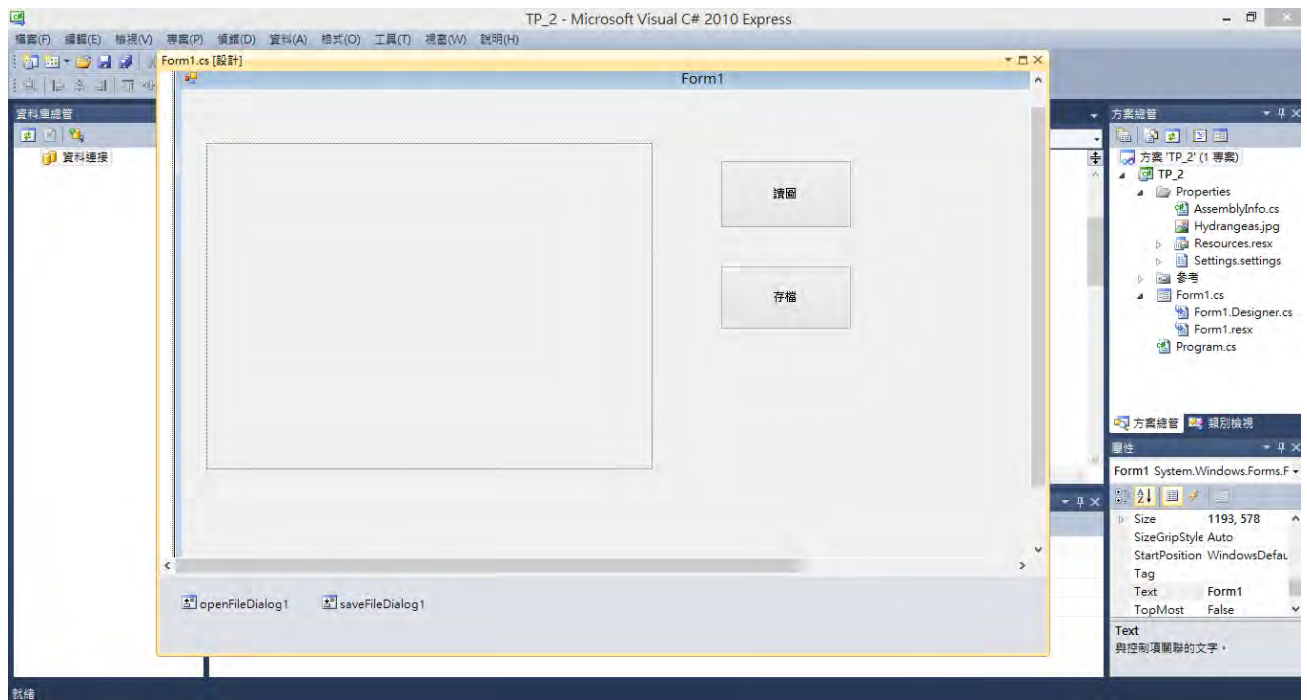
}
}

```

練習(一): 等差相乘,不同方法的輸出輸入

練習(二): 尋找一矩陣之最大與最小值

簡單的讀圖、存圖



```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;  
using AForge.Imaging;  
using AForge.Imaging.Formats;  
using AForge.Video;
```

```
namespace TP_2  
{  
    public partial class Form1 : Form  
    {  
        Bitmap imgone;  
        public Form1()  
        {
```

```

        InitializeComponent();
    }

    private void button1_Click(object sender, EventArgs e)
    {
        openFileDialog1.Filter = "所有檔案 (*.*)|*.*";
        if (openFileDialog1.ShowDialog() == DialogResult.OK)
        {
            imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
            pictureBox1.Image = imgone;
        }
    }

    private void button2_Click(object sender, EventArgs e)
    {
        saveFileDialog1.FileName = @"imgone.bmp";
        saveFileDialog1.Filter = "Bitmap file (*.bmp)|*.bmp";
        if (saveFileDialog1.ShowDialog() == DialogResult.OK)
        {
            imgone.Save(saveFileDialog1.FileName); //, ImageFormat.Bmp);
        }
    }

    private void button3_Click(object sender, EventArgs e)
    {
    }
}

```

練習:直接讀圖、不經 openFileDialog，圖放 debug 資料匣

(TP_2_ReadAndSaveImage_NoDiag)

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;

```

```

using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Video;

namespace TP_2
{
    public partial class Form1 : Form
    {
        Bitmap imgone;
        public Form1()
        {
            InitializeComponent();

        }

        private void button1_Click(object sender, EventArgs e)
        {
            imgone = ImageDecoder.DecodeFromFile("Tulips.jpg");
            pictureBox1.Image = imgone;

        }

        private void button2_Click(object sender, EventArgs e)
        {
            saveFileDialog1.FileName = @"imgone.bmp";
            saveFileDialog1.Filter = "Bitmap file (*.bmp)|*.bmp";
            if (saveFileDialog1.ShowDialog() == DialogResult.OK)
            {
                imgone.Save(saveFileDialog1.FileName); //, ImageFormat.Bmp);
            }
        }
    }
}

```

```
    }

    private void pictureBox1_Click(object sender, EventArgs e)
    {

    }

    private void button3_Click(object sender, EventArgs e)
    {

    }
}

}
```

Lesson 3

Emgucv Load and Show Image

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
//-----
using System.Threading;
using Emgu.CV;
using Emgu.CV.Structure;
using Emgu.CV.CvEnum;

namespace EmguVS2010LoadImage
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        Image<Bgr, Byte> srcimg;

        private void button1_Click_1(object sender, EventArgs e)
        {
            OpenFileDialog Openfile = new OpenFileDialog();
            if (Openfile.ShowDialog() == DialogResult.OK)
            {
                srcimg = new Image<Bgr, byte>(Openfile.FileName);
                pictureBox1.Image = srcimg.ToBitmap();
            }
        }
    }
}
```

Lesson 4

Emgucv Image Rotate and Resize

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
//-----
using System.Threading;
using Emgu.CV;
using Emgu.CV.Structure;
using Emgu.CV.CvEnum;

namespace EmguVS2010LoadImage
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        Image<Bgr, Byte> srcimg;
        Mat img = new Mat(400, 400, DepthType.Cv8U, 3);

        private void button1_Click_1(object sender, EventArgs e)
        {
            OpenFileDialog Openfile = new OpenFileDialog();
            if (Openfile.ShowDialog() == DialogResult.OK)
            {
                srcimg = new Image<Bgr, byte>(Openfile.FileName);
                imageBox1.Image = srcimg;
            }
        }

        private void button2_Click(object sender, EventArgs e)
```

```

        {
            Image<Bgr, byte> rotateImage = srcimg.Rotate(30, new
Bgr(System.Drawing.Color.Black));
            imageBox2.Image = rotateImage;
        }

private void button3_Click(object sender, EventArgs e)
{
    Image<Bgr, byte> zoomImage = srcimg.Resize(2.0,
Emgu.CV.CvEnum.Inter.Cubic);
    imageBox2.Image = zoomImage;
}

private void button4_Click(object sender, EventArgs e)
{
    Image<Bgr, byte> resizeImage = srcimg.Resize(400, 200,
Emgu.CV.CvEnum.Inter.Cubic);
    imageBox2.Image = resizeImage;
}

private void button5_Click(object sender, EventArgs e)
{
    Image<Bgr, byte> notImage = srcimg.Not();
    imageBox2.Image = notImage;
}
    }
}

```

Lesson 5

Emgucv Image Dilate and Erode

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
//-----
using System.Threading;
using Emgu.CV;
using Emgu.CV.Structure;
using Emgu.CV.CvEnum;

namespace EmguVS2010LoadImage
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        Image<Bgr, Byte> srcimg;
        Image<Gray, byte> grayImage, thresholdImage;

        private void button1_Click_1(object sender, EventArgs e)
        {
            OpenFileDialog Openfile = new OpenFileDialog();
            if (Openfile.ShowDialog() == DialogResult.OK)
            {
                srcimg = new Image<Bgr, byte>(Openfile.FileName);
                imageBox1.Image = srcimg;
            }
            grayImage=srcimg.Convert<Gray,byte>();
        }
    }
}
```

```

        //二值化的閾值
        Gray thresholdValue = new Gray(127);
        //取得二值化影像
        thresholdImage = grayImage.ThresholdBinary(thresholdValue, new
Gray(255));
        imageBox2.Image=thresholdImage;
    }

    private void button2_Click(object sender, EventArgs e)
    {
        //擴張處理(擴張 2 像素)
        Image<Gray, byte> dilateImage = thresholdImage.Dilate(2);
        imageBox2.Image = dilateImage;
    }

    private void button3_Click(object sender, EventArgs e)
    {
        //侵蝕處理(侵蝕 2 像素)
        Image<Gray, byte> erodeImage = thresholdImage.Erode(2);
        imageBox2.Image = erodeImage;
    }
}
}

```

Lesson 6
Set Pixel Speed
(Emgucv_VS2015_speed)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Diagnostics;

using Emgu.CV;
using Emgu.Util;
using Emgu.CV.Structure;
using Emgu.CV.CvEnum;

namespace Emgucv_VS2015
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        Mat srcimg;
        Image <Bgr,Byte> img;
        Image<Gray, Byte> grayimg, bwimg;
        Stopwatch stw = new Stopwatch();

        private void button1_Click(object sender, EventArgs e)
        {
            Bitmap source = new Bitmap(1024, 1024);
            stw.Start();
            for (int h = 0; h < source.Height; h++)
```

```

        {
            for (int w = 0; w < source.Width; w++)
            {
                source.SetPixel(w, h, Color.White);
            }
        }

        stw.Stop();
        TimeSpan ts = stw.Elapsed;
        label1.Text = ts.ToString();

        pictureBox1.Image = source;
    }

private void button2_Click(object sender, EventArgs e)
{
    Image<Bgr, Byte> img = new Image<Bgr, Byte>(1024, 1024);
    stw.Reset();
    stw.Start();
    for (int h = 0; h < img.Height; h++)
    {
        for (int w = 0; w < img.Width; w++)
        {
            img[h, w] = new Bgr(255, 255, 255);
        }
    }
    stw.Stop();
    TimeSpan ts = stw.Elapsed;
    label2.Text = ts.ToString();
    pictureBox2.Image = img.ToBitmap();
}

private void button3_Click(object sender, EventArgs e)
{
    Image<Bgr, Byte> img = new Image<Bgr, Byte>(1024, 1024);
    stw.Reset();

```

```

        stw.Start();
        //需加入 using System.Drawing.Imaging;
        byte[, ] data = img.Data;
        for (int h = 0; h < img.Height; h++)
        {
            for (int w = 0; w < img.Width; w++)
            {
                data[h, w, 0] = 0;
                data[h, w, 1] = 0;
                data[h, w, 2] = 0;
            }
        }

        stw.Stop();
        TimeSpan ts = stw.Elapsed;
        label2.Text = ts.ToString();
        pictureBox2.Image = img.ToBitmap();
    }

    private void button4_Click(object sender, EventArgs e)
    {
        Image<Bgr, Byte> img = new Image<Bgr, Byte>(1024, 1024);
        stw.Reset();
        stw.Start();
        Parallel.For(0, img.Height, h =>
        {
            Parallel.For(0, img.Width, w =>
            {
                img[h, w] = new Bgr(255, 255, 255);

            });
        });

        stw.Stop();
        TimeSpan ts = stw.Elapsed;
        label2.Text = ts.ToString();
        pictureBox2.Image = img.ToBitmap();
    }

```

```
}  
}
```

Emgucv 與 System 之 SetPixel Speed 比較 (原程式 Emgu2.9_SetPixel_Speed)

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;  
  
using System.Diagnostics;  
using Emgu.CV;  
using Emgu.Util;  
using Emgu.CV.Structure;  
  
namespace Emgu2._9_Matching  
{  
    public partial class Form1 : Form  
    {  
        public Form1()  
        {  
            InitializeComponent();  
        }  
  
        Bitmap myBitmap;  
        Image<Bgr, Byte> srcimg;  
        Stopwatch stp = new Stopwatch();
```

```

private void button1_Click(object sender, EventArgs e)
{
    OpenFileDialog Openfile = new OpenFileDialog();
    if (Openfile.ShowDialog() == DialogResult.OK)
    {
        srcimg = new Image<Bgr, Byte>(Openfile.FileName);
        myBitmap = srcimg.ToBitmap();
        pictureBox1.Image = myBitmap;
    }
}

```

```

private void button2_Click(object sender, EventArgs e)
{
    stp.Reset();
    stp.Start();

    for (int x = 0; x < srcimg.Width; x++)
    {
        for (int y = 0; y < srcimg.Height; y++)
        {
            myBitmap.SetPixel(x, y, Color.Yellow);
        }
    }

    stp.Stop();
    label1.Text = stp.Elapsed.ToString();
    pictureBox2.Image = myBitmap;
}

```

```

private void button3_Click(object sender, EventArgs e)
{
    stp.Reset();
    stp.Start();
    byte[, ] data = srcimg.Data;
    for (int x = 0; x < srcimg.Width; x++)
    {
        for (int y = 0; y < srcimg.Height; y++)

```

```

        {
            data[y,x,0]=0;
            data[y,x,1]=255;
            data[y,x,2]=255;
        }
    }

    stp.Stop();
    label2.Text = stp.Elapsed.ToString();
    pictureBox3.Image = srcimg.ToBitmap();
}

private void button4_Click(object sender, EventArgs e)
{
    stp.Reset();
    stp.Start();
    byte[, ,] data = srcimg.Data;
    for (int y = 0; y < srcimg.Height; y++)
    {
        for (int x = 0; x < srcimg.Width; x++)
        {
            data[y, x, 0] = 0;
            data[y, x, 1] = 255;
            data[y, x, 2] = 255;
        }
    }

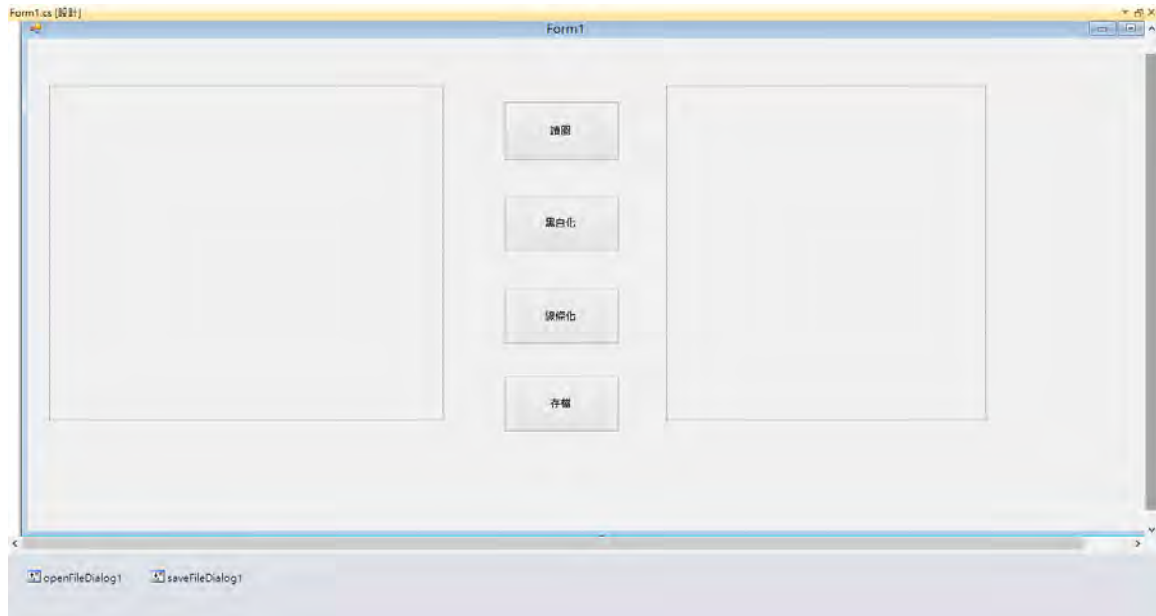
    stp.Stop();
    label3.Text = stp.Elapsed.ToString();
    pictureBox4.Image = srcimg.ToBitmap();
}
}
}

```

Lesson 7

二值化

(TP_3_BW_and_Line)



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using AForge.Imaging;
using AForge.Imaging.Filters;
using AForge.Imaging.Formats;
```

```
namespace TP_2
{
    public partial class Form1 : Form
    {
        Bitmap imgone, imgtwo, img_3;
```

```

public Form1()
{
    InitializeComponent();
}

private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案 (*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        pictureBox1.Image = imgone;
    }
}

private void pictureBox1_Click(object sender, EventArgs e)
{
}

private void button3_Click(object sender, EventArgs e)
{
    // create grayscale filter (BT709)
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    // apply the filter
    imgtwo = filter.Apply(imgone);
    pictureBox2.Image = imgtwo;
}

private void pictureBox2_Click(object sender, EventArgs e)
{
}

private void button4_Click(object sender, EventArgs e)
{
}

```

```

        // create filter
        CannyEdgeDetector filter = new CannyEdgeDetector();
        // apply the filter
        imgtwo=filter.Apply(imgtwo);
        pictureBox2.Image = imgtwo;
    }

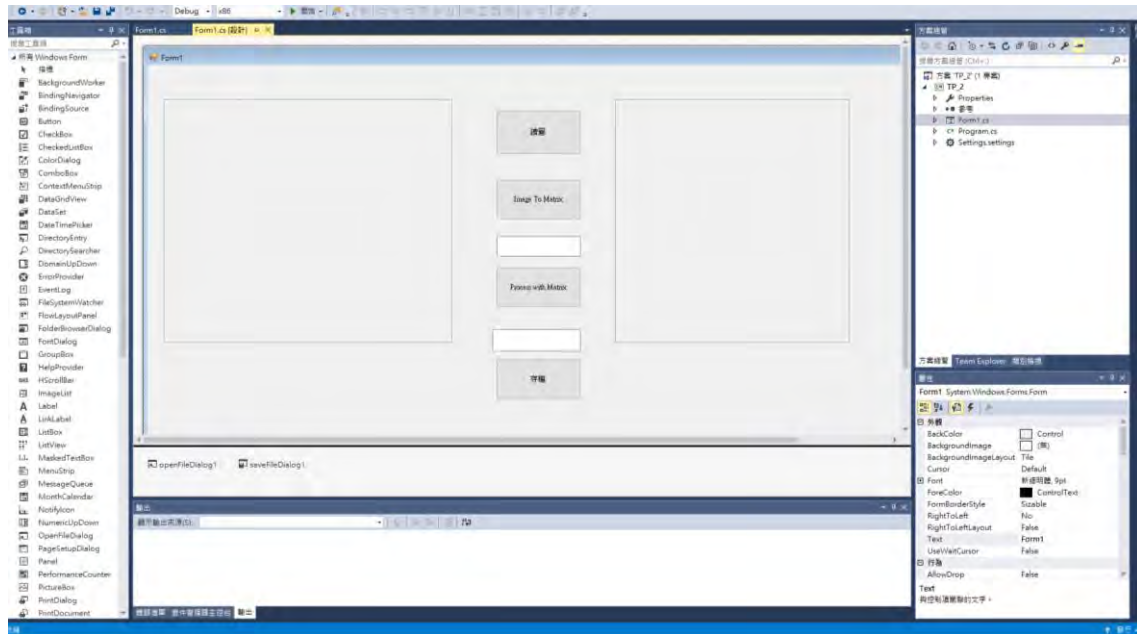
    private void button2_Click(object sender, EventArgs e)
    {
        saveFileDialog1.FileName = @"img_3.bmp";
        saveFileDialog1.Filter = "Bitmap file (*.bmp)|*.bmp";
        if (saveFileDialog1.ShowDialog() == DialogResult.OK)
        {
            img_3.Save(saveFileDialog1.FileName);
        }
    }
}

```

注意 CannyEdgeDetector 只能處理 8bpp (灰階圖)

練習：使用 Homogeneity edge detector、Difference Edge Detector、Difference Edge Detector、Sobel Edge Detector、Canny Edge Detector

圖檔與矩陣的轉換，並做簡單處理
(TP_4 Image to Matrix)



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using AForge.Imaging;
using AForge.Imaging.Filters;
using AForge.Imaging.Formats;
```

```
using System.Diagnostics;
```

```
namespace TP_2
{
    public partial class Form1 : Form
    {
        Bitmap imgone,imgtwo;
        Byte [,] imgdata = new byte [1024, 1024,3 ];
```

```

public Form1()
{
    InitializeComponent();
}
Stopwatch istp = new Stopwatch();

private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案(*.*)|*.*";
    // 最好找 lena //

    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);

        }
        imgtwo = imgone;
        // imgtwo = (Bitmap)imgone.Clone();

        pictureBox1.Image = imgone;
        pictureBox2.Image = imgtwo;

    }

private void pictureBox1_Click(object sender, EventArgs e)
{

}

private void button3_Click(object sender, EventArgs e)
{
    istp.Reset();
    istp.Start();
}

```

```

        for (int i = 0; i < imgone.Width; i++)
        {
            for (int j = 0; j < imgone.Height; j++)
            {
                //Color pixelColor = imgone.GetPixel(i, j);
                imgdata[i,j,0] = imgone.GetPixel(i, j).R;
                imgdata[i, j, 1] = imgone.GetPixel(i, j).G;
                imgdata[i, j, 2] = imgone.GetPixel(i, j).B;

            }
        }
        istp.Stop();
        textBox1.Text = istp.Elapsed.ToString();
    }

    private void pictureBox2_Click(object sender, EventArgs e)
    {

    }

    private void button4_Click(object sender, EventArgs e)
    {
        istp.Reset();
        istp.Start();
        for (int i = 1; i < imgone.Width; i++)
        {
            for (int j = 1; j < imgone.Height; j++)
            {
                if (i > j)
                {
                    // imgdata[i, j, 0] = 0;
                    // imgdata[i, j, 1] = 0;
                    // imgdata[i, j, 2] = 0;
                    Color myColor = Color.FromArgb(0, 0, 0);
                    imgtwo.SetPixel(i, j, myColor);
                }
            }
        }
    }

```

```

    }
    istp.Stop();
    textBox2.Text = istp.Elapsed.ToString();
    pictureBox2.Image = imgtwo;
    pictureBox1.Image = imgone;
}

```

```

private void button2_Click(object sender, EventArgs e)
{
    saveFileDialog1.FileName = @"imgtwo.bmp";
    saveFileDialog1.Filter = "Bitmap file (*.bmp)|*.bmp";
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgone.Save(saveFileDialog1.FileName);
    }
}

```

```

private void Form1_Load(object sender, EventArgs e)
{
}

```

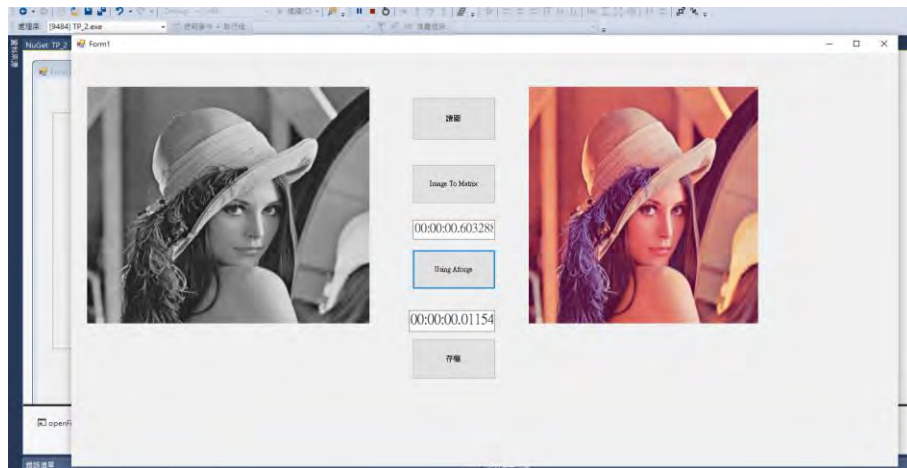
```

}

```

自己寫 Rgb To Gray

程式 TP_4_ Imaging to Matrix To Gray



```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using AForge.Imaging;
using AForge.Imaging.Filters;
using AForge.Imaging.Formats;

```

```

using System.Diagnostics;

```

```

namespace TP_2
{
    public partial class Form1 : Form
    {

        public Form1()
        {
            InitializeComponent();

            Bitmap imgone, imgtwo;

            Stopwatch istp = new Stopwatch();
            Color myColor = Color.FromArgb(0, 0, 0);

            private void button1_Click(object sender, EventArgs e)
            {
                openFileDialog1.Filter = "所有檔案(*.*)|*.*";
            }

```

```

// 最好找 lena //

if (openFileDialog1.ShowDialog() == DialogResult.OK)
{
    imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);

}

//imgtwo = imgone;
imgtwo = (Bitmap)imgone.Clone();

pictureBox1.Image = imgone;
pictureBox2.Image = imgtwo;

}

private Bitmap Rgb2gray(Bitmap srcimg)
{
    Bitmap dstimg = (Bitmap)srcimg.Clone();
    // Byte[,] igray = new byte[1024, 1024];
    Byte igray;
    Color icolor;
    for (int i = 0; i < srcimg.Width; i++)
    {
        for (int j = 0; j < srcimg.Height; j++)
        {
            igray = (Byte)(0.299 * srcimg.GetPixel(i, j).R + 0.587 *
srcimg.GetPixel(i, j).G + 0.144 * srcimg.GetPixel(i, j).B);
            icolor = Color.FromArgb(igray, igray, igray);
            dstimg.SetPixel(i, j, icolor);
        }
    }

    return dstimg;
}

```

```
private void pictureBox1_Click(object sender, EventArgs e)
{

}
```

```
private void button3_Click(object sender, EventArgs e)
{
    istp.Reset();
    istp.Start();
    imgtwo = Rgb2gray(imgone);
    istp.Stop();
    textBox1.Text = istp.Elapsed.ToString();

    pictureBox2.Image = imgtwo;
    pictureBox1.Image = imgone;

}
```

```
private void pictureBox2_Click(object sender, EventArgs e)
{

}
```

```
private void button4_Click(object sender, EventArgs e)
{

    istp.Reset();
    istp.Start();

    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    imgtwo = filter.Apply(imgone);

    istp.Stop();
    textBox2.Text = istp.Elapsed.ToString();

    pictureBox1.Image = imgtwo;
```

```

        pictureBox2.Image = imgone;
    }

    private void button2_Click(object sender, EventArgs e)
    {
        saveFileDialog1.FileName = @"imgtwo.bmp";
        saveFileDialog1.Filter = "Bitmap file (*.bmp)|*.bmp";
        if (saveFileDialog1.ShowDialog() == DialogResult.OK)
        {
            imgone.Save(saveFileDialog1.FileName);
        }
    }

    private void Form1_Load(object sender, EventArgs e)
    {
        }
    }
}

```

使用 Accord.Net

(TP_4 Image to Matrix_使用 Accord)

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using AForge.Imaging;
using AForge.Imaging.Filters;
using AForge.Imaging.Formats;

```

```
using System.Diagnostics;
```

```
using Accord.Imaging.Converters;
```

```
namespace TP_2
{
    public partial class Form1 : Form
    {
        Bitmap imgone, imgtwo;
        Byte[, , ] imgdata = new Byte[1024, 1024, 3];
        Byte[,] imgdata_gray = new Byte[1024, 1024];
        // int[, ] maG = new int[1024, 1024];
        // int[, ] maB = new int[1024, 1024];

        public Form1()
        {
            InitializeComponent();
        }
        Stopwatch istp = new Stopwatch();

        private void button1_Click(object sender, EventArgs e)
        {
            openFileDialog1.Filter = "所有檔案(*.*)|*.*";
            // 最好找 lena //

            if (openFileDialog1.ShowDialog() == DialogResult.OK)
            {
                imgone =
                ImageDecoder.DecodeFromFile(openFileDialog1.FileName);

                imgtwo = imgone;
                // imgtwo = (Bitmap)imgone.Clone();

                pictureBox1.Image = imgone;
                pictureBox2.Image = imgtwo;
            }
        }
    }
}
```

```
}
```

```
private void pictureBox1_Click(object sender, EventArgs e)
{

}
```

```
private void button3_Click(object sender, EventArgs e)
{
    istp.Reset();
    istp.Start();
    ImageToMatrix conv = new ImageToMatrix();
    conv.Convert(imgone, out imgdata);
    istp.Stop();
    textBox1.Text = istp.Elapsed.ToString();
}
```

```
private void button4_Click(object sender, EventArgs e)
{
    // create grayscale filter (BT709)
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    // apply the filter
    Bitmap grayImage = filter.Apply(imgone);
    ImageToMatrix conv = new ImageToMatrix();
    conv.Convert(grayImage, out imgdata_gray);

    istp.Reset();
    istp.Start();
    for (int i = 1; i < imgone.Width; i++)
    {
        for (int j = 1; j < imgone.Height; j++)
        {
            if (i < j)
```

```

        {
            imgdata_gray[i, j] = 0;
        }
    }
}

MatrixToImage M_conv = new MatrixToImage();
M_conv.Convert(imgdata_gray, out imgtwo);

istp.Stop();
textBox2.Text = istp.Elapsed.ToString();
pictureBox2.Image = imgtwo;
pictureBox1.Image = imgone;
}

private void button2_Click(object sender, EventArgs e)
{
    saveFileDialog1.FileName = @"imgtwo.bmp";
    saveFileDialog1.Filter = "Bitmap file (*.bmp)|*.bmp";
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgone.Save(saveFileDialog1.FileName);
    }
}

private void Form1_Load(object sender, EventArgs e)
{
}
}
}

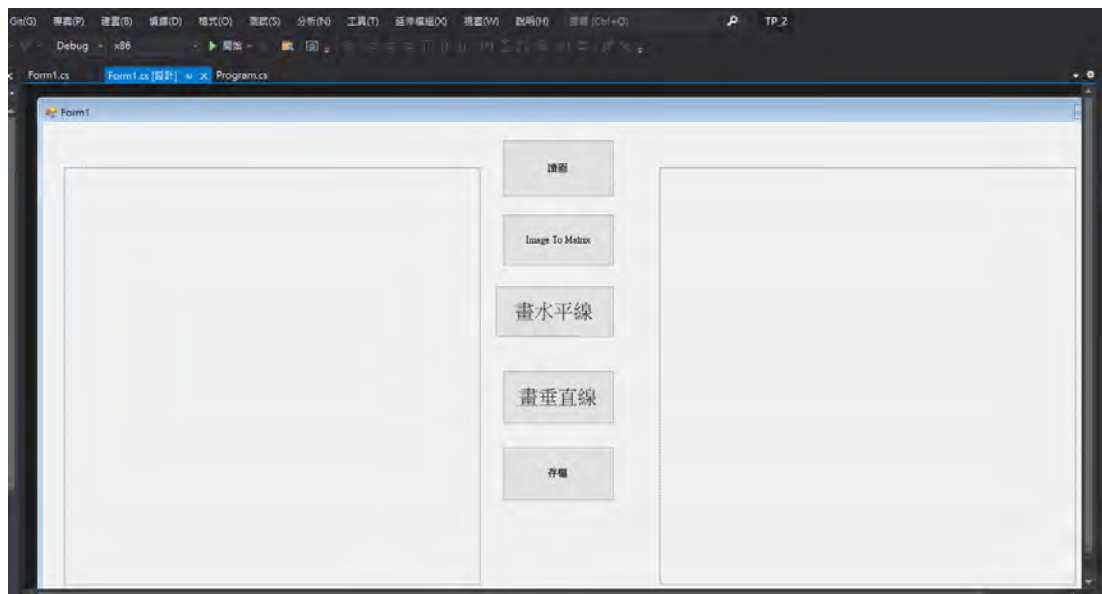
```

```
imgtwo=imgone;
```

```
imgtwo=(Bitmap) imgone.Clone();
```

練習

1. 讀圖、將其右上變成白色
2. 讀圖、中間畫紅水平線
3. 中間畫紅垂直線
4. 比較 `imgtwo=imgone`
與 `imgtwo=new Bitmap(imgone);`



(TP_5_Image_To_Matrix-DrawLine)

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;  
using AForge.Imaging;  
using AForge.Imaging.Filters;  
using AForge.Imaging.Formats;
```

```

namespace TP_2
{
    public partial class Form1 : Form
    {
        Bitmap imgone, imgtwo;
        int[,] maR = new int[1024, 1024];
        int[,] maG = new int[1024, 1024];
        int[,] maB = new int[1024, 1024];

        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            openFileDialog1.Filter = "所有檔案(*.*)|*.*";
            if (openFileDialog1.ShowDialog() == DialogResult.OK)
            {
                imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
                //imgtwo = imgone;
                pictureBox1.Image = imgone;
                pictureBox2.Image = imgtwo;
            }
        }

        private void pictureBox1_Click(object sender, EventArgs e)
        {
        }
    }
}

```

```

private void button3_Click(object sender, EventArgs e)
{
    for (int i = 1; i < imgone.Width; i++)
    {
        for (int j = 1; j < imgone.Height; j++)
        {
            Color pixelColor = imgone.GetPixel(i, j);
            maR[i,j] = pixelColor.R;
            maG[i, j] = pixelColor.G;
            maB[i, j] = pixelColor.B;
        }
    }
}

private void pictureBox2_Click(object sender, EventArgs e)
{
}

private void button4_Click(object sender, EventArgs e)
{
    Int16 dx = (Int16)(imgone.Width / 2);
    imgtwo = new Bitmap(imgone);
    for (int i = dx-1; i < dx+1; i++)
    {
        for (int j = 0; j < imgone.Height; j++)
        {
            imgtwo.SetPixel(i, j, Color.Red);
        }
    }

    pictureBox2.Image = imgtwo;
    pictureBox1.Image = imgone;
}

```

```

private void button2_Click(object sender, EventArgs e)
{
    saveFileDialog1.FileName = @"imgtwo.bmp";
    saveFileDialog1.Filter = "Bitmap file (*.bmp)|*.bmp";
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgtwo.Save(saveFileDialog1.FileName);
    }
}

private void button5_Click(object sender, EventArgs e)
{
    Int16 dy = (Int16)(imgone.Height / 2);
    imgtwo = new Bitmap(imgone);
    for (int i = 0; i < imgone.Width; i++)
    {
        for (int j = dy-1; j < dy+1; j++)
        {
            imgtwo.SetPixel(i, j, Color.White);
        }
    }

    pictureBox2.Image = imgtwo;
    pictureBox1.Image = imgone;
}

private void Form1_Load(object sender, EventArgs e)
{
}

}
}

```

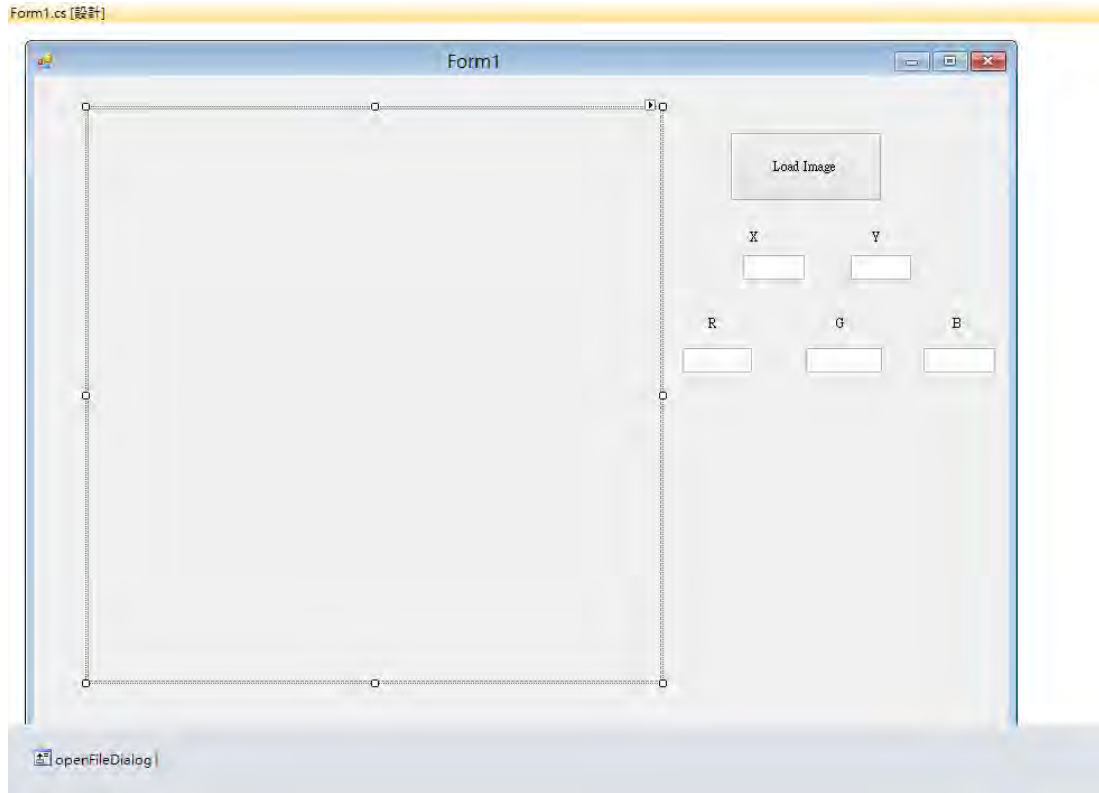
(TP_5_Image_To_Matrix-upperwhite)

3. pictureBox 不因最小化，復原而更改

讀圖、輸入座標、回傳 RGB 值

Lesson 8

讀圖、顯示滑鼠之座標、單一 pixel 之 RGB



程式 GetPixelColor

圖改 Autotize

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace MouseAndGetPixel
{
    public partial class Form1 : Form
    {
```

```

public Form1()
{
    InitializeComponent();
}
Bitmap imgone;
Color colorPixel;

private void button1_Click(object sender, EventArgs e)
{
    OpenFileDialog OpenImage = new OpenFileDialog();
    if (OpenImage.ShowDialog() == DialogResult.OK)
    {
        imgone = new Bitmap(OpenImage.FileName);
        pictureBox1.Image = imgone;
    }
}

private void imgone_MouseMove(object sender, MouseEventArgs e)
{
    // imgone_MouseMove 自己從事件處命名//
    if (imgone!=null)
    {
        //顯示滑鼠座標
        textBox4.Text = e.X.ToString();
        textBox5.Text = e.Y.ToString();

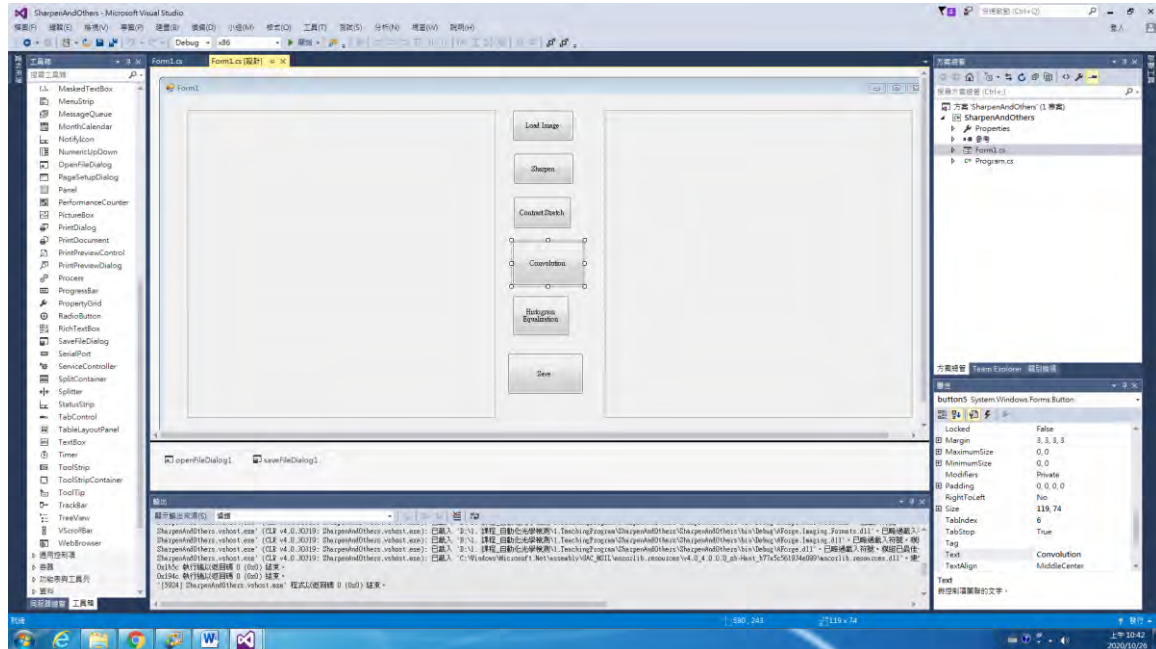
        //取得 pixel 的 RGB 值
        colorPixel = imgone.GetPixel(e.X, e.Y);
        textBox1.Text = colorPixel.R.ToString();
        textBox2.Text = colorPixel.G.ToString();
        textBox3.Text = colorPixel.B.ToString();
    }
}
}

```

Sharpen, Contrast Stretch, and Histogram Equalization

Note: explain convolution filter

使用 T2MV0.tif
程式 SharpenAndOthers



using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

using AForge;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;

```
namespace GammaCorrection  
{  
    public partial class Form1 : Form  
    {  
        public Form1()  
        {
```

```

        InitializeComponent();
    }

    Bitmap imgone;

    private void button1_Click(object sender, EventArgs e)
    {
        openFileDialog1.Filter = "所有檔案 (*.*)|*.*";
        if (openFileDialog1.ShowDialog() == DialogResult.OK)
        {
            imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
            pictureBox1.Image = imgone;
        }
    }

    private void button2_Click(object sender, EventArgs e)
    {

        Sharpen filter = new Sharpen();

        Bitmap img=filter.Apply(imgone);
        pictureBox2.Image = img;

    }

    private void button3_Click(object sender, EventArgs e)
    {
        Bitmap img;
        // create filter
        ContrastStretch filter = new ContrastStretch();
        // process image
        img=filter.Apply(imgone);
        pictureBox2.Image = img;
    }

    private void button4_Click(object sender, EventArgs e)

```

```

{

    // create filter
    HistogramEqualization filter = new HistogramEqualization();
    // process image
    Bitmap img=filter.Apply(imgone);
    // create filter
    pictureBox2.Image = img;

}

private void pictureBox2_Click(object sender, EventArgs e)
{

}

private void button5_Click(object sender, EventArgs e)
{

    // define emboss kernel
    int[,] kernel = {
        { -6, -2,  0 },
        { -2,  1,  2 },
        {  0,  2,  6 } };
    // create filter
    Convolution filter = new Convolution(kernel);
    // apply the filter
    Bitmap img=filter.Apply(imgone);
    pictureBox2.Image = img;

}

private void button6_Click(object sender, EventArgs e)
{
    saveFileDialog1.FileName = @"imgone.bmp";
    saveFileDialog1.Filter = "Bitmap file (*.bmp)|*.bmp";
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {

```

```

        imgone.Save(saveFileDialog1.FileName); //, ImageFormat.Bmp);
    }
}
}
}

```

// 自己寫 sharpen filter 查 Convolution filter 對 remote 做處理
 //filter.Apply 與 filter.ApplyInPlace 之差別

Elapsed Time

```

//程式 ElapsedTime_1
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

using System.Diagnostics;

namespace MyElapsedTime
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            Stopwatch stopWatch = new Stopwatch();

            private void button1_Click(object sender, EventArgs e)
            {

```

```

        stopWatch.Reset();
        stopWatch.Start();
    }

    private void button2_Click(object sender, EventArgs e)
    {
        stopWatch.Stop();
        // Get the elapsed time as a TimeSpan value.
        TimeSpan ts = stopWatch.Elapsed;
        textBox1.Text = ts.ToString();
    }
}
}

```

// 程式 ElapsedTime _2

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

```

```

using System.Diagnostics;

```

```

namespace MyElapsedTime
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            Stopwatch stopWatch = new Stopwatch();

```

```

private void button1_Click(object sender, EventArgs e)
{

    stopWatch.Reset();
    stopWatch.Start();

    int maxi=10000;
    int maxj=10000;
    int[ , ] ma=new int[maxi,maxj];

    for (int i=0;i<maxi;i++)
    {
        for (int j = 0; j < maxj; j++)
        {
            ma [i,j]= i * j;

        }
    }

    stopWatch.Stop();
    // Get the elapsed time as a TimeSpan value.
    TimeSpan ts = stopWatch.Elapsed;
    textBox1.Text = ts.ToString();

}

}

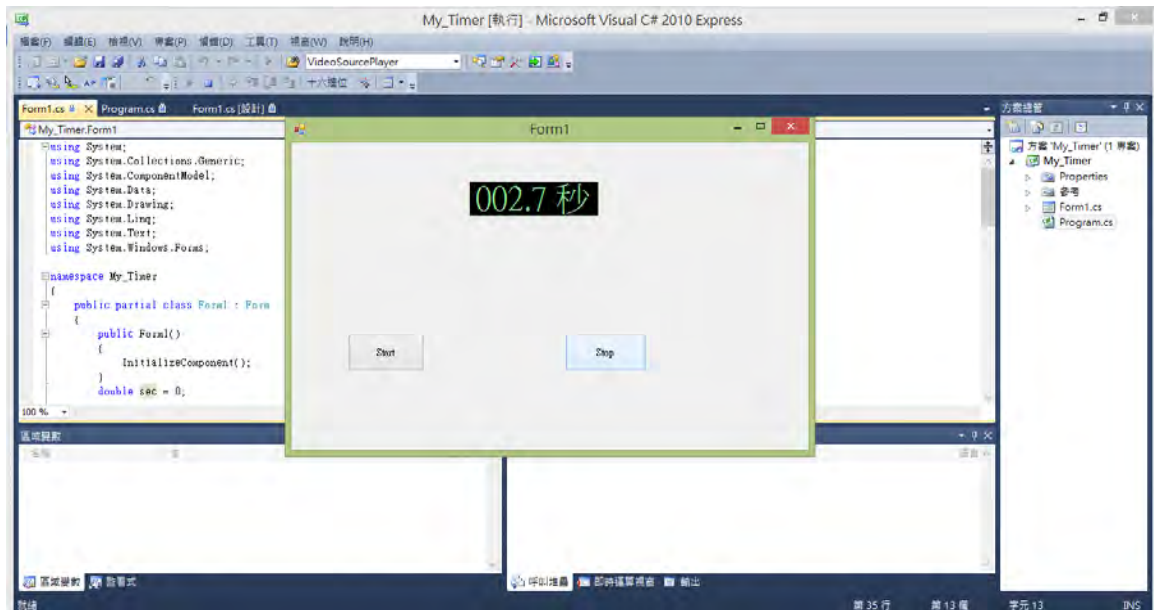
```

練習:

1. ma [i,j]= i * j; i,j 對調 elapsed time=1.3
2. 改 maxi=5000,maxj=5000 的值 elapsed time=0.1
3. ma[i,j,k], k=1:2, elapsed time=1.17

上題: Sharpen, Contrast Stretch, and Histogram Equalization 各自消耗多少時間?

Timer



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
namespace My_Timer
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        double sec = 0;

        private void Form1_Load(object sender, EventArgs e)
        {
            label1.BackColor = Color.Black; //背景為黑色
        }
    }
}
```

```

        label1.ForeColor = Color.LightGreen;    //字為亮綠色
        label1.TextAlign = ContentAlignment.MiddleCenter;    //文字置中
        label1.Text = string.Format("{0:0000.0 秒}", sec);
    }

    private void label1_Click(object sender, EventArgs e)
    {

    }

    private void button1_Click(object sender, EventArgs e)
    {
        sec = 0;    //秒數歸零
        timer1.Enabled = true;    //啟動計時器
    }

    private void timer1_Tick(object sender, EventArgs e)
    {
        label1.Text = string.Format("{0:000.0 秒}", sec += 0.1); //秒數加 0.1
    }

    private void button2_Click(object sender, EventArgs e)
    {
        timer1.Enabled = false; //關閉計時器
    }
}

```

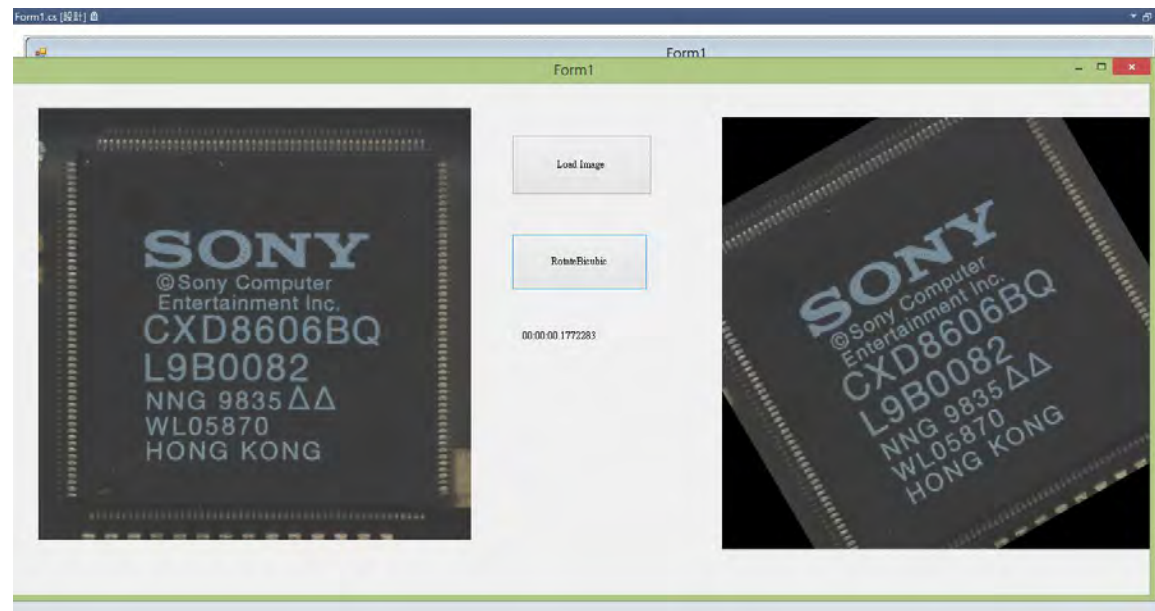
Lesson 9

Rotation、Add and Subtract 的應用

程式:RotateImage

圖使用 TrafficSign_2

範例使用:RotateBicubic，請查 help



```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;
```

```
using AForge.Imaging;  
using AForge.Imaging.Formats;  
using AForge.Imaging.Filters;
```

```
namespace ImageRotate  
{  
    public partial class Form1 : Form  
    {  
        public Form1()  
        {
```

```

        InitializeComponent();
    }

    Bitmap imgone;

    private void pictureBox1_Click(object sender, EventArgs e)
    {

    }

    private void button1_Click(object sender, EventArgs e)
    {
        openFileDialog1.Filter = "所有檔案(*.*)|*.*";
        if (openFileDialog1.ShowDialog() == DialogResult.OK)
        {
            imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
            pictureBox1.Image = imgone;
        }

    }

    private void button2_Click(object sender, EventArgs e)
    {
        // create filter - rotate for 30 degrees keeping original image size
        RotateBicubic filter = new RotateBicubic(30, true);
        //The filter accepts 8 bpp grayscale images and 24 bpp color images for
processing.
        // apply the filter
        Bitmap newImage = filter.Apply(imgone);
        pictureBox2.Image = newImage;
    }
}

```

練習:

1. 增加 elapsed time 計算耗時

2. 增加 RotateBilinear 與 RotateNearestNeighbor 並比較三者之耗時

解答: 增加 elapsed time 計算耗時

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;
using System.Diagnostics;
```

```
namespace ImageRotate
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            Bitmap imgone;
            Stopwatch stopWatch = new Stopwatch();

            private void pictureBox1_Click(object sender, EventArgs e)
            {

            }

            private void button1_Click(object sender, EventArgs e)
            {
                openFileDialog1.Filter = "所有檔案(*.*)|*.*";
                if (openFileDialog1.ShowDialog() == DialogResult.OK)
```

```

        {
            imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
            pictureBox1.Image = imgone;
        }

    }

private void button2_Click(object sender, EventArgs e)
{
    stopWatch.Reset();

    stopWatch.Start();

    // create filter - rotate for 30 degrees keeping original image size
    RotateBicubic filter = new RotateBicubic(30, true);
    //The filter accepts 8 bpp grayscale images and 24 bpp color images for
processing.
    // apply the filter
    Bitmap newImage = filter.Apply(imgone);
    pictureBox2.Image = newImage;

    stopWatch.Stop();
    // Get the elapsed time as a TimeSpan value.
    TimeSpan ts = stopWatch.Elapsed;
    label1.Text = ts.ToString();

}
}
}

```

Add and Subtract

利用 Subtract 製造 mask
 使用 IC_1_Mask.bmp 與 IC_1.bmp
 或使用 Bolt_0、Bolt_1
 必須是相同大小的圖
 程式 AddAndSubtract

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;
using System.Diagnostics;

namespace ImageRotate
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            Bitmap imgone, imgtwo;
            Stopwatch stopWatch = new Stopwatch();

            private void pictureBox1_Click(object sender, EventArgs e)
            {

            }

            private void button1_Click(object sender, EventArgs e)
            {
                openFileDialog1.Filter = "所有檔案(*.*)|*.*";
                if (openFileDialog1.ShowDialog() == DialogResult.OK)
                {
                    imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);

```

```

        pictureBox1.Image = imgone;
    }
}

private void button5_Click(object sender, EventArgs e)
{
    // create filter
    Invert filter = new Invert();
    // apply the filter
    filter.ApplyInPlace(imgone);
    pictureBox1.Image = imgone;
}

private void button2_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案 (*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgtwo =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        pictureBox2.Image = imgtwo;
    }
}

private void button3_Click(object sender, EventArgs e)
{
    // create filter
    Add filter = new Add(imgtwo);
    // apply the filter
    Bitmap resultImage = filter.Apply(imgone);
    pictureBox3.Image = resultImage;
}

private void pictureBox2_Click(object sender, EventArgs e)
{

```

```

    }

    private void button4_Click(object sender, EventArgs e)
    {
        // create filter
        Subtract filter = new Subtract(imgone);
        // apply the filter
        Bitmap resultImage = filter.Apply(imgtwo);
        pictureBox4.Image = resultImage;
    }
}
}

```

使用 Difference filter 尋找 IC_1 與 IC_2 的差異

想辦法 使差異明顯

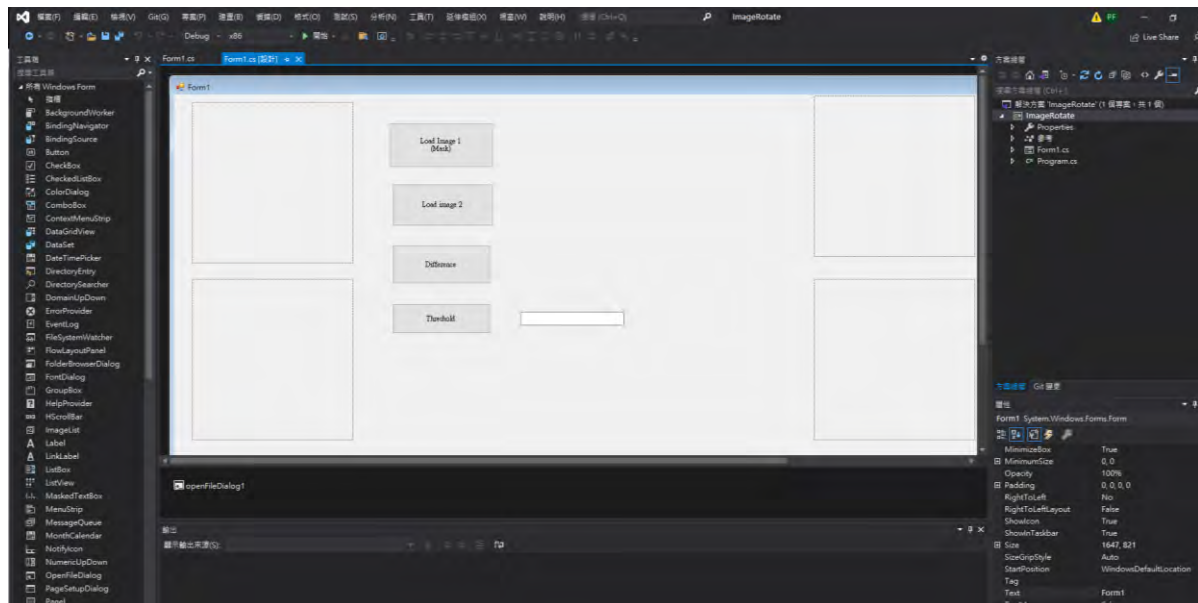
使用 Threshold Filter

解答 : Difference_Threshold

ImageStatistics 之 PixelCount PixelCountWithoutBlack

解答 : Difference_ImageStatistics

解答 : Difference_Threshold



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;
using System.Diagnostics;
```

```
namespace ImageRotate
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
    }
}
```

```
Bitmap imgone, imgtwo, dstimg, grayImage, bwimg;
Stopwatch stopWatch = new Stopwatch();
```

```

private void pictureBox1_Click(object sender, EventArgs e)
{

}

private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案 (*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        pictureBox1.Image = imgone;
    }

}

private void button5_Click(object sender, EventArgs e)
{

}

private void button2_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案 (*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgtwo =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        pictureBox2.Image = imgtwo;
    }

}

private void button3_Click(object sender, EventArgs e)

```

```

{
    // create filter
    Difference filter = new Difference(imgtwo);
    // apply the filter
    dstimg = filter.Apply(imgone);
    pictureBox3.Image = dstimg;
}

private void pictureBox2_Click(object sender, EventArgs e)
{

}

private void button5_Click_1(object sender, EventArgs e)
{

}

private void button4_Click_1(object sender, EventArgs e)
{

    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    // apply the filter
    grayImage = filter.Apply(dstimg);

    Threshold filter_2 = new Threshold(10);
    // apply the filter
    bwimg= filter_2.Apply(grayImage);
    pictureBox4.Image = bwimg;
}

private void button4_Click(object sender, EventArgs e)
{

}

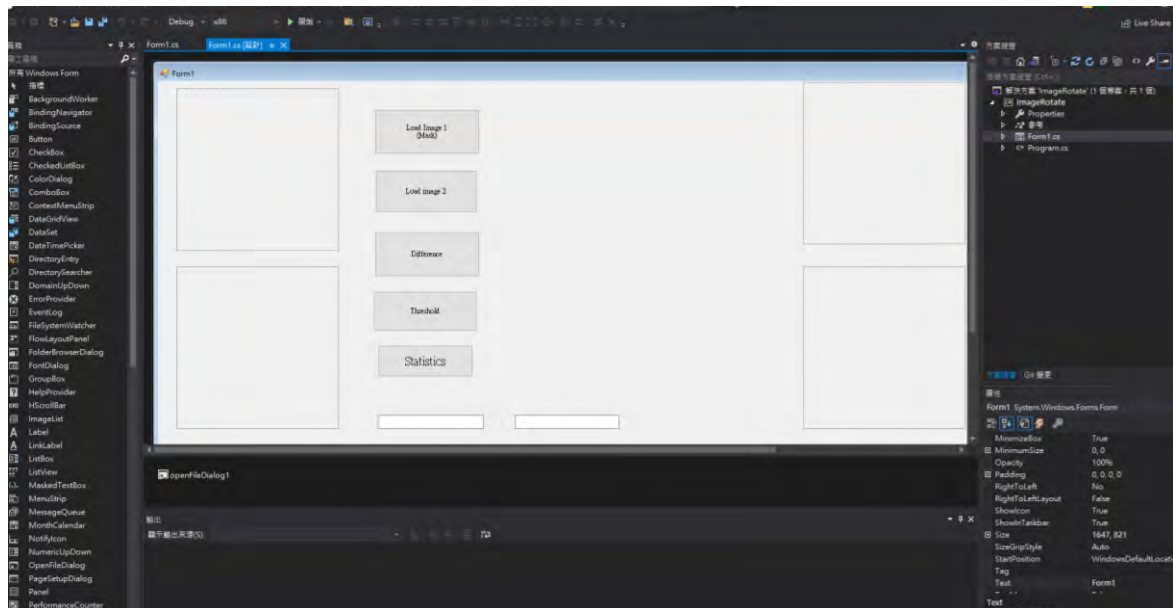
```

```

    }
}

```

解答 : Difference_ImageStatistics



```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;
using System.Diagnostics;

namespace ImageRotate
{
    public partial class Form1 : Form
    {

```

```

public Form1()
{
    InitializeComponent();
}

Bitmap imgone, imgtwo, dstimg, grayImage, bwimg;
Stopwatch stopWatch = new Stopwatch();

private void pictureBox1_Click(object sender, EventArgs e)
{
}

private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案(*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        pictureBox1.Image = imgone;
    }
}

private void button5_Click(object sender, EventArgs e)
{
}

private void button2_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案(*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgtwo =

```

```

ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        pictureBox2.Image = imgtwo;
    }

}

private void button3_Click(object sender, EventArgs e)
{
    // create filter
    Difference filter = new Difference(imgtwo);
    // apply the filter
    dstimg = filter.Apply(imgone);
    pictureBox3.Image = dstimg;
}

private void pictureBox2_Click(object sender, EventArgs e)
{

}

private void button5_Click_1(object sender, EventArgs e)
{
    ImageStatistics i_stat=new ImageStatistics(bwimg);
    textBox1.Text = i_stat.PixelsCount.ToString();
    textBox2.Text = i_stat.PixelsCountWithoutBlack.ToString();
}

private void button4_Click_1(object sender, EventArgs e)
{

    // create grayscale filter (BT709)
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    // apply the filter
    grayImage = filter.Apply(dstimg);

    // create filter
    Threshold filter_2 = new Threshold(10);
    // apply the filter

```

```

        bwimg= filter_2.Apply(grayImage);
        pictureBox4.Image = bwimg;
    }

    private void button4_Click(object sender, EventArgs e)
    {

    }

}
}

```

Draw Rectangle, Line, Polygon__方法一

程式 DrawRectangleLine

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

using AForge;
using System.Drawing.Imaging;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;
using System.Diagnostics;

namespace ImageRotate
{
    public partial class Form1 : Form
    {
        public Form1()

```

```

{
    InitializeComponent();
}

Bitmap sourceImage;
Stopwatch stopWatch = new Stopwatch();

private void pictureBox1_Click(object sender, EventArgs e)
{

}

private void button1_Click(object sender, EventArgs e)
{

    openFileDialog1.Filter = "所有檔案(*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        sourceImage =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        pictureBox1.Image = sourceImage;
    }

}

private void button2_Click(object sender, EventArgs e)
{
    BitmapData data = sourceImage.LockBits(
        new Rectangle(0, 0, sourceImage.Width, sourceImage.Height),
        ImageLockMode.ReadWrite, sourceImage.PixelFormat);

    Rectangle rect = new Rectangle(0, 0, 100, 200);

    Drawing.Rectangle(data, rect, Color.White);

    sourceImage.UnlockBits(data);
}

```

```

        pictureBox3.Image = sourceImage;
    }

    private void button3_Click(object sender, EventArgs e)
    {
        BitmapData data = sourceImage.LockBits(
            new Rectangle(0, 0, sourceImage.Width, sourceImage.Height),
            ImageLockMode.ReadWrite, sourceImage.PixelFormat);
        IntPoint p1=new IntPoint(50,100);
        IntPoint p2=new IntPoint(150,200);
        Drawing.Line(data, p1,p2, Color.Black);

        sourceImage.UnlockBits(data);
        pictureBox3.Image = sourceImage;
    }

```

```

    private void button4_Click(object sender, EventArgs e)
    {
        BitmapData data = sourceImage.LockBits(
            new Rectangle(0, 0, sourceImage.Width, sourceImage.Height),
            ImageLockMode.ReadWrite, sourceImage.PixelFormat);

        IntPoint p1 = new IntPoint(100, 100);
        IntPoint p2 = new IntPoint(200, 200);
        IntPoint p3 = new IntPoint(350, 150);
        List <IntPoint> polygon_point=new List<IntPoint>();
        polygon_point.Add(p1);
        polygon_point.Add(p2);
        polygon_point.Add(p3);

        Drawing.Polygon(data, polygon_point, Color.Red);

        sourceImage.UnlockBits(data);
    }

```

```

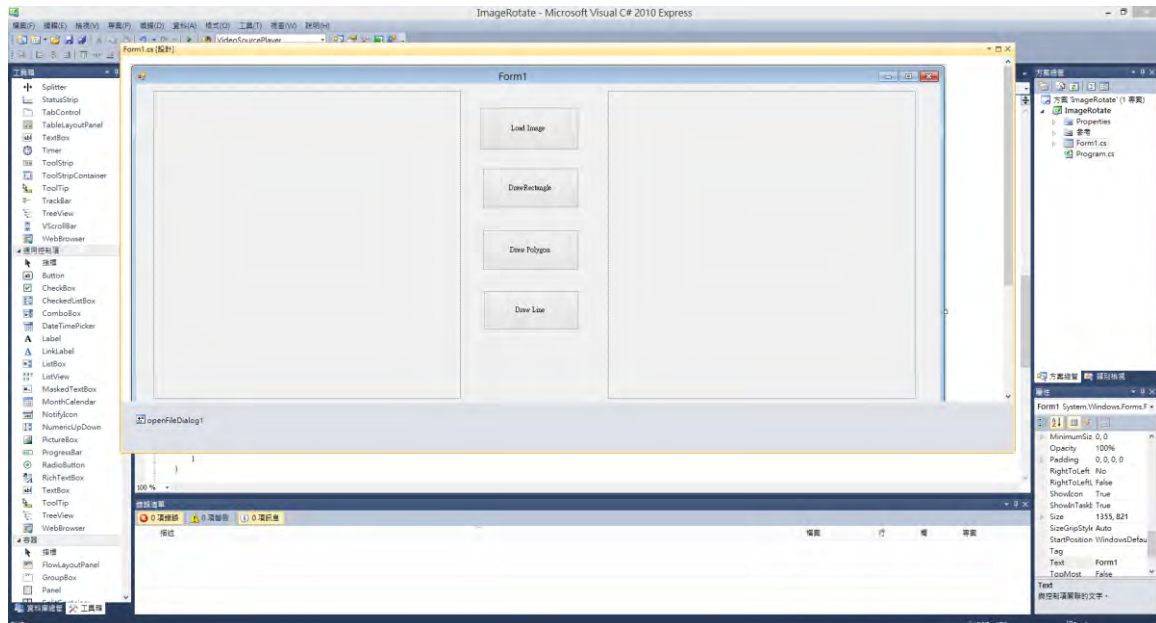
        pictureBox3.Image = sourceImage;
    }

}

}

```

Draw Rectangle, Line, Polygon 方法二



```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

using AForge;
using System.Drawing.Imaging;
//using AForge.Imaging;
using AForge.Imaging.Formats;
//using AForge.Imaging.Filters;
using System.Diagnostics;

namespace ImageRotate

```

```

{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            Bitmap sourceImage;
            Stopwatch stopWatch = new Stopwatch();

            private void pictureBox1_Click(object sender, EventArgs e)
            {

            }

            private void button1_Click(object sender, EventArgs e)
            {
                openFileDialog1.Filter = "所有檔案(*.*)|*.*";
                if (openFileDialog1.ShowDialog() == DialogResult.OK)
                {
                    sourceImage =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
                    pictureBox1.Image = sourceImage;

                }
            }

            private void button2_Click(object sender, EventArgs e)
            {
                Bitmap imgone = new Bitmap(sourceImage);

                using (Graphics g = Graphics.FromImage(imgone))
                {
                    g.DrawRectangle(new Pen(Color.Red), 40, 70, 300, 300);
                }
                pictureBox3.Image = imgone;
            }
        }
    }
}

```

```
}
```

```
private void button4_Click(object sender, EventArgs e)
{
    Bitmap imgone = new Bitmap(sourceImage);
    using (Graphics g = Graphics.FromImage(imgone))
    {
        // Create pen.
        Pen blackPen = new Pen(Color.Black, 3);

        // Create points that define polygon.
        System.Drawing.Point point1 = new System.Drawing.Point(50, 50);
        System.Drawing.Point point2 = new System.Drawing.Point(100, 25);
        System.Drawing.Point point3 = new System.Drawing.Point(200, 5);
        System.Drawing.Point point4 = new System.Drawing.Point(250, 50);
        System.Drawing.Point point5 = new System.Drawing.Point(300, 100);
        System.Drawing.Point point6 = new System.Drawing.Point(350, 200);
        System.Drawing.Point point7 = new System.Drawing.Point(250, 250);
        System.Drawing.Point[] curvePoints =
        {
            point1,
            point2,
            point3,
            point4,
            point5,
            point6,
            point7
        };

        // Draw polygon to screen.
        g.DrawPolygon(blackPen, curvePoints);
        pictureBox3.Image = imgone;
    }
}
```

```
private void button5_Click(object sender, EventArgs e)
{
    Bitmap imgone = new Bitmap(sourceImage);
    using (Graphics g = Graphics.FromImage(imgone))
    {
        g.DrawLine(new Pen(Color.Red), 400, 100, 400, 400);
    }

    pictureBox3.Image = imgone;
}
}
```

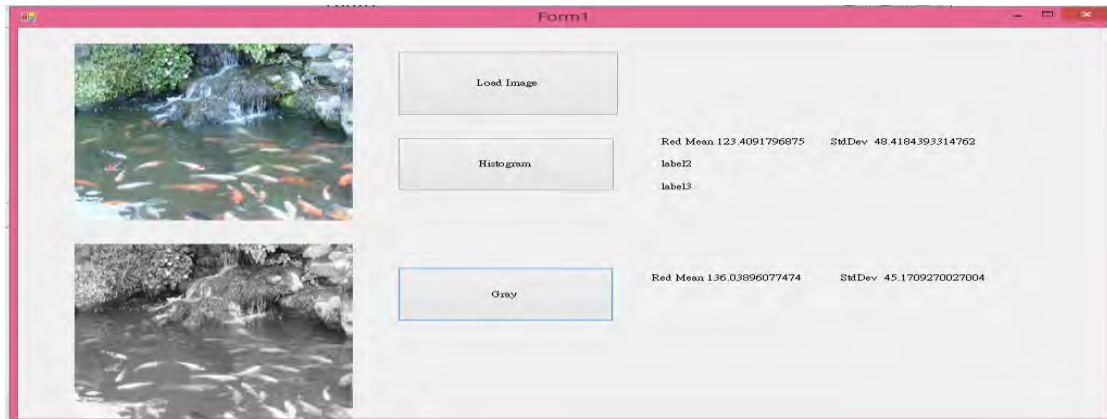
練習:畫出五邊形

練習: `Bitmap imgone = new Bitmap(sourceImage);`

改成 `imgone=sourceImage;` 會怎樣???

Lesson 10

Histogram



程式 Histogram_1

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;
```

```
using AForge;  
using AForge.Imaging;  
using AForge.Math;  
using AForge.Imaging.Formats;  
using AForge.Imaging.Filters;
```

```
namespace Histogram_1  
{  
    public partial class Form1 : Form  
    {  
        public Form1()  
        {
```

```

        InitializeComponent();
    }

    Bitmap srcimg;

    private void button1_Click(object sender, EventArgs e)
    {
        openFileDialog1.Filter = "所有檔案 (*.*)|*.*";
        if (openFileDialog1.ShowDialog() == DialogResult.OK)
        {
            srcimg =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
            pictureBox1.Image = srcimg;

        }

    }

    private void button2_Click(object sender, EventArgs e)
    {
        ImageStatistics stat = new ImageStatistics(srcimg);
        // get red channel's histogram
        Histogram red = stat.Red;
        label1.Text = "Red Mean " + red.Mean.ToString() + "          StdDev
" + red.StdDev.ToString();
        Histogram green = stat.Green;
        Histogram blue = stat.Blue;
        //補齊 green and blue

    }

    private void Form1_Load(object sender, EventArgs e)
    {

    }

    private void button3_Click(object sender, EventArgs e)
    {

```

```

// create grayscale filter (BT709)
Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
// apply the filter
Bitmap grayimg = filter.Apply(srcimg);
pictureBox2.Image = grayimg;

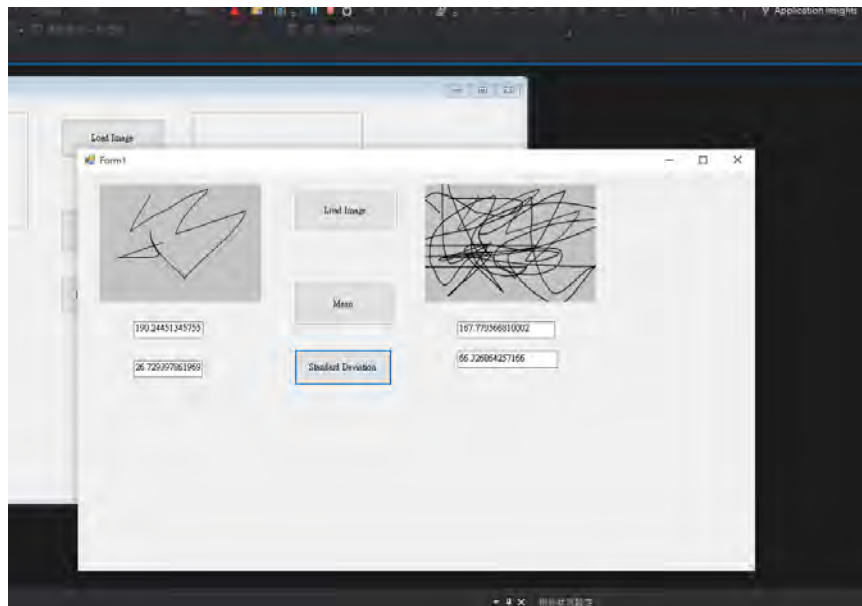
ImageStatistics stat = new ImageStatistics(grayimg);
// get red channel's histogram
Histogram gray = stat.Gray;
label4.Text = "Gray Mean " + gray.Mean.ToString() + "
StdDev  " + gray.StdDev.ToString();

}

}
}

```

-
1. 計算 P1 P2 之平均值、與標準差(請同學完成程式
TP50_P1AndP2_mean_stdDev)



```

sing System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;

```

```

using System.Linq;
using System.Text;
using System.Windows.Forms;

using AForge.Imaging;
using AForge.Imaging.Filters;
using AForge.Imaging.Formats;
using AForge.Math;

namespace TP50
{
    public partial class Form1 : Form
    {
        Bitmap imgone, imgtwo;

        public Form1()
        {
            InitializeComponent();

            private void button1_Click(object sender, EventArgs e)
            {
                Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);

                imgone = ImageDecoder.DecodeFromFile("p1.bmp");
                pictureBox1.Image= imgone;
                imgone=filter.Apply(imgone);

                imgtwo = ImageDecoder.DecodeFromFile("p2.bmp");
                imgtwo = filter.Apply(imgtwo);
                pictureBox2.Image = imgtwo;
            }

            private void button3_Click(object sender, EventArgs e)
            {
                ImageStatistics statone = new ImageStatistics(imgone);
                // get red channel's histogram
                Histogram grayone = statone.Gray;

```

```

        textBox2.Text = grayone.StdDev.ToString();

        ImageStatistics stattwo = new ImageStatistics(imgtwo);
        // get red channel's histogram
        Histogram graytwo = stattwo.Gray;
        textBox4.Text = graytwo.StdDev.ToString();
    }

    private void button2_Click(object sender, EventArgs e)
    {

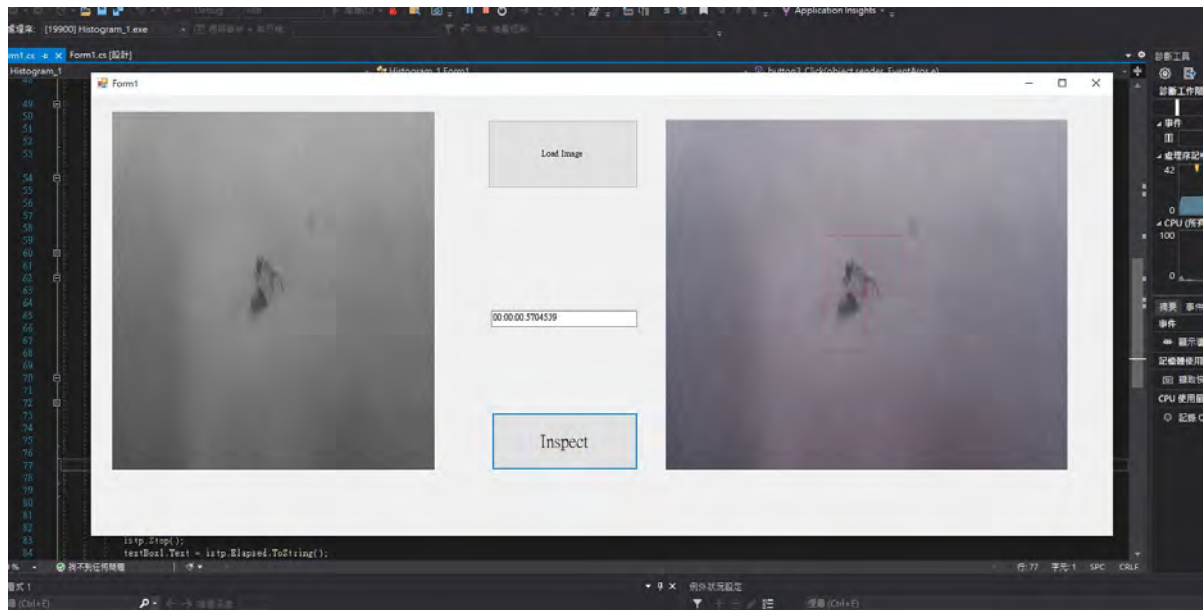
        ImageStatistics statone = new ImageStatistics(imgone);
        // get red channel's histogram
        Histogram grayone = statone.Gray;
        textBox1.Text = grayone.Mean.ToString();

        ImageStatistics stattwo = new ImageStatistics(imgtwo);
        // get red channel's histogram
        Histogram graytwo = stattwo.Gray;
        textBox3.Text = graytwo.Mean.ToString();

    }
}

```

2. 利用標準差找出 L02 之瑕疵 (先利用 crop 方法) 請同學完成程式
L02_Tostudent (此程式展示 crop)



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Diagnostics;
```

```
using AForge;
using AForge.Imaging;
using AForge.Math;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;
```

```
namespace Histogram_1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
    }
}
```

```

Bitmap srcimg,grayimg;
Stopwatch istp = new Stopwatch();

private void button1_Click(object sender, EventArgs e)
{

    srcimg = ImageDecoder.DecodeFromFile("L02.jpg");
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    grayimg = filter.Apply(srcimg);
    pictureBox1.Image = grayimg;

}

private void button2_Click(object sender, EventArgs e)
{

}

private void Form1_Load(object sender, EventArgs e)
{

}

private void button3_Click(object sender, EventArgs e)
{

    istp.Reset();
    istp.Start();
    int sub = 256;
    for (int i=0;i<grayimg.Width-sub; i=i+sub)
    {
        for (int j=0;j<grayimg.Height-sub;j=j+sub)
        {

```

```

Crop filter_crop = new Crop(new Rectangle(i, j, sub, sub));
Bitmap subimg = filter_crop.Apply(grayimg);

ImageStatistics stat = new ImageStatistics(subimg);
Histogram stat_gray = stat.Gray;
if (stat_gray.StdDev > 8.5)
{
    using (Graphics g = Graphics.FromImage(srcimg))
    {
        g.DrawRectangle(new Pen(Color.Red), i, j, sub,
sub);
    }
}

pictureBox2.Image = srcimg;
istp.Stop();
textBox1.Text = istp.Elapsed.ToString();

}

}
}

```

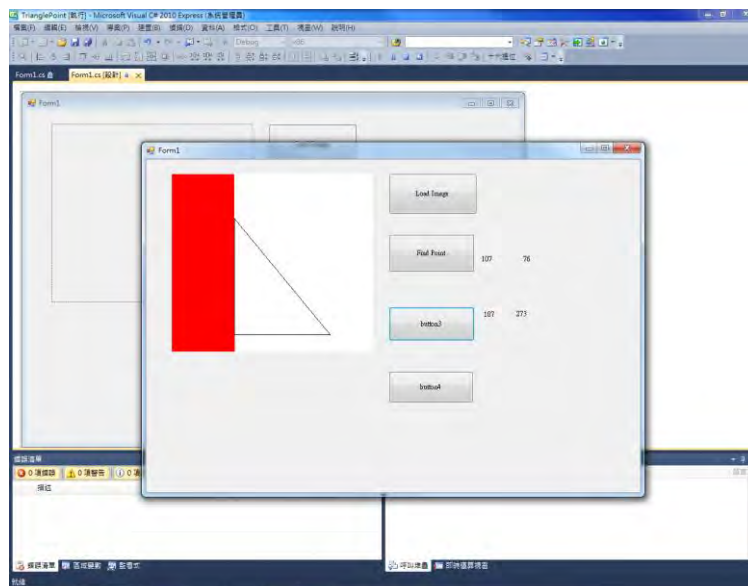
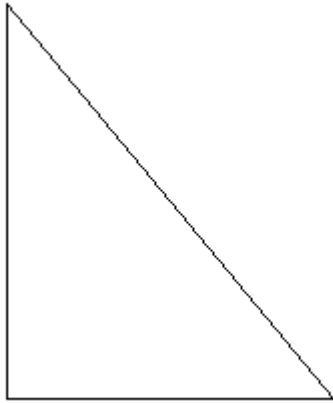
Lesson 11

找三角形的頂點

(程式 TP_8_2_找頂點_OK)

(程式 TP_8_ToStudent)

讀圖，並找三角形的頂點



```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;  
  
using AForge.Imaging;
```

```
using AForge.Imaging.Filters;
using AForge.Imaging.Formats;
```

```
namespace TrianglePoint
```

```
{
```

```
    public partial class Form1 : Form
```

```
    {
```

```
        public Form1()
```

```
        {
```

```
            InitializeComponent();
```

```
        }
```

```
        Bitmap imgone;
```

```
        private void button1_Click(object sender, EventArgs e)
```

```
        {
```

```
            imgone = ImageDecoder.DecodeFromFile("Triangle.bmp");
```

```
            pictureBox1.Image = imgone;
```

```
            //先將圖轉成黑白較保險，Why??
```

```
            Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
```

```
            // apply the filter
```

```
            imgone = filter.Apply(imgone);
```

```
            OtsuThreshold filter_2 = new OtsuThreshold();
```

```
            // apply the filter
```

```
            imgone = filter_2.Apply(imgone);
```

```
        }
```

```
        private void button2_Click(object sender, EventArgs e)
```

```
        {
```

```
            Bitmap imgtwo=new Bitmap (imgone);
```

```
            //Bitmap imgtwo = imgone;
```

```
            for (int i = 1; i < imgone.Width; i++)
```

```
            {
```

```

        for (int j = 1; j < imgone.Height; j++)
        {
            Color pixelColor = imgone.GetPixel(i, j);

            if (pixelColor.R < 20)
            {
                int ii = i;
                label1.Text = ii.ToString();
                int jj = j;
                label2.Text = jj.ToString();
                goto Found;
            }
            imgtwo.SetPixel(i, j, Color.Blue);
        }
    }
    Found:

    pictureBox1.Image = imgtwo;
}

```

```

private void button3_Click(object sender, EventArgs e)
{
    Bitmap imgtwo = new Bitmap(imgone);
    for (int i = 1; i < imgone.Width; i = i + 1)
    {
        for (int j = imgone.Height-1; j > 0;j=j-1 )
        {
            Color pixelColor = imgone.GetPixel(i, j);

            if (pixelColor.R < 20)
            {
                int ii = i;
                label3.Text = ii.ToString();
                int jj = j;
                label4.Text = jj.ToString();
                goto Found1;
            }
        }
    }
    Found1:
}

```

```

        }
        imgtwo.SetPixel(i, j, Color.Red);
    }
}
Found1:
    pictureBox1.Image = imgtwo;
}

}
}

```

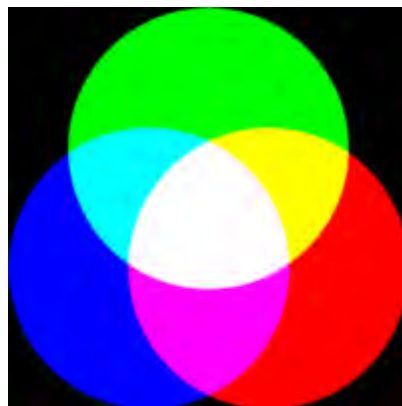
練習；1. 找出其他的頂點

Ans: TP8_2 找頂點

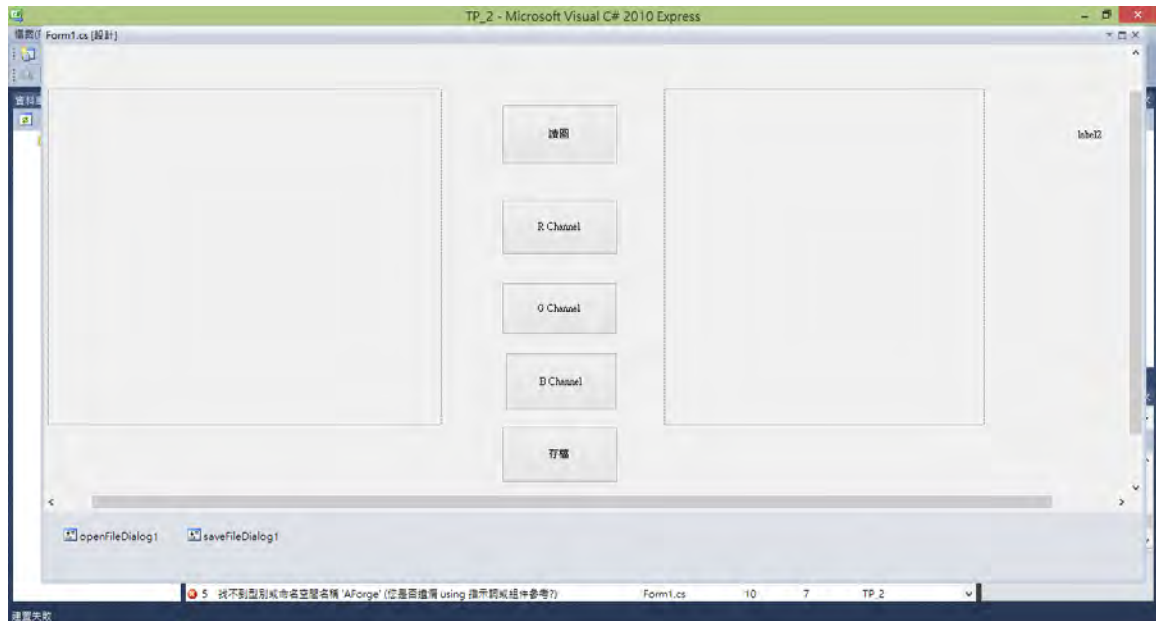
2. 面積?

讀圖，並將紅、藍、綠分開

(程式 TP_9_RGB_Channel)



3Color_RGB



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using AForge.Imaging;
using AForge.Imaging.Filters;
using AForge.Imaging.Formats;
```

```
namespace TP_2
{
    public partial class Form1 : Form
    {
        Bitmap imgone, imgtwo;

        public Form1()
        {
            InitializeComponent();
        }
    }
}
```

```

private void button1_Click(object sender, EventArgs e)
{

    openFileDialog1.Filter = "所有檔案 (*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        pictureBox1.Image = imgone;
    }

}

private void button3_Click(object sender, EventArgs e)
{
    // create filter
    ExtractChannel filter = new ExtractChannel(RGB.R);
    // apply the filter
    Bitmap channelImage = filter.Apply(imgone);
    pictureBox2.Image = channelImage;
}

private void button2_Click(object sender, EventArgs e)
{
    saveFileDialog1.FileName = @"imgtwo.bmp";
    saveFileDialog1.Filter = "Bitmap file (*.bmp)|*.bmp";
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgone.Save(saveFileDialog1.FileName);
    }
}

private void button4_Click(object sender, EventArgs e)
{
    // create filter
    ExtractChannel filter = new ExtractChannel(RGB.G);

```

```

        // apply the filter
        Bitmap channelImage = filter.Apply(imgone);
        pictureBox2.Image = channelImage;
    }

private void button5_Click(object sender, EventArgs e)
{
    // create filter
    ExtractChannel filter = new ExtractChannel(RGB.B);
    // apply the filter
    Bitmap channelImage = filter.Apply(imgone);
    pictureBox2.Image = channelImage;
}
}
}

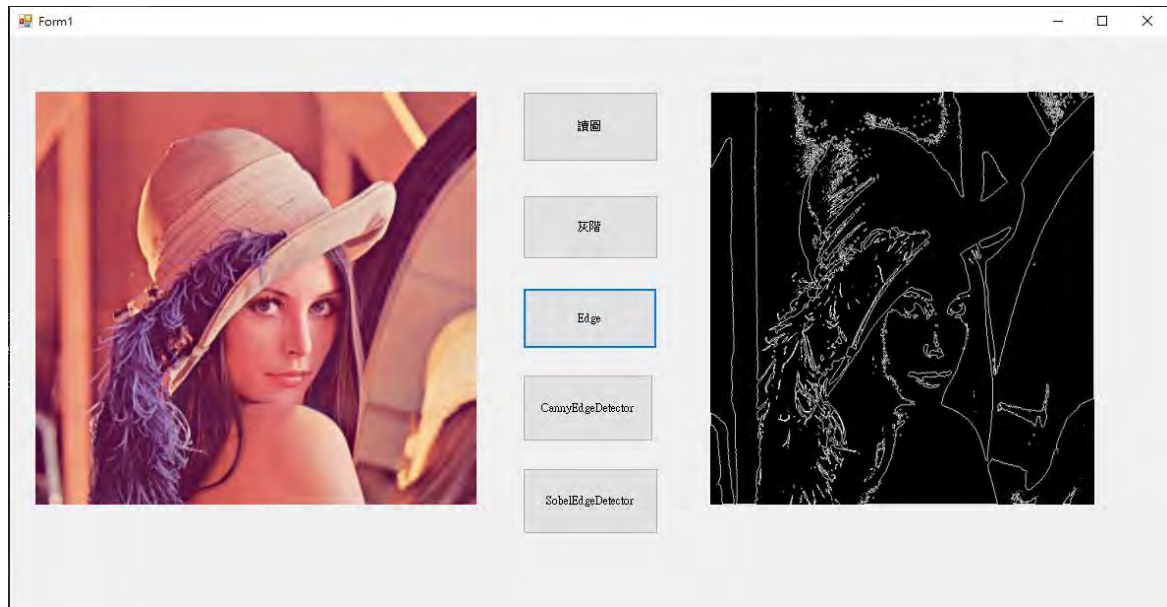
```

Lesson 12

尋找圖形之邊緣與量測

(TP10_灰階找邊緣)

彩色圖變灰階，並邊緣化



```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;  
using AForge.Imaging;  
using AForge.Imaging.Filters;  
using AForge.Imaging.Formats;
```

```
namespace TP_2  
{  
    public partial class Form1 : Form  
    {  
        Bitmap imgone, grayImage, imgtwo;
```

```

public Form1()
{
    InitializeComponent();
}

private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案 (*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        pictureBox1.Image = imgone;
    }
}

private void pictureBox1_Click(object sender, EventArgs e)
{
}

private void button3_Click(object sender, EventArgs e)
{
    // create grayscale filter (BT709)
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    // apply the filter
    grayImage = filter.Apply(imgone);
    pictureBox2.Image = grayImage;
}

```

```

private void button2_Click(object sender, EventArgs e)
{

    OtsuThreshold filter = new OtsuThreshold();
    // apply the filter
    imgtwo = filter.Apply(grayImage);

    //imgtwo = new Bitmap(imgone);
    SobelEdgeDetector filter2 = new SobelEdgeDetector();
    // apply the filter
    filter2.ApplyInPlace(imgtwo);

    pictureBox2.Image = imgtwo;
}

private void Form1_Load(object sender, EventArgs e)
{

}

private void label1_Click(object sender, EventArgs e)
{

}

private void pictureBox2_Click(object sender, EventArgs e)
{

}

private void button4_Click(object sender, EventArgs e)
{

```

```

        // create filter
        OtsuThreshold filter = new OtsuThreshold();
        // apply the filter
        imgtwo=filter.Apply(grayImage);

        //imgtwo = new Bitmap(imgone);
        Edges filter2 = new Edges();
        // apply the filter
        filter2.ApplyInPlace(imgtwo);

        pictureBox2.Image = imgtwo;
    }

    private void button5_Click(object sender, EventArgs e)
    {
        // create filter
        OtsuThreshold filter = new OtsuThreshold();
        // apply the filter
        imgtwo = filter.Apply(grayImage);

        //imgtwo = new Bitmap(imgone);
        CannyEdgeDetector filter2 = new CannyEdgeDetector();
        // apply the filter
        filter2.ApplyInPlace(imgtwo);

        pictureBox2.Image = imgtwo;
    }
}

```

量測

TP61_ToStudent

TP61_

簡單的量測



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
using AForge.Imaging;
using AForge.Imaging.Filters;
using AForge.Imaging.Formats;
```

```
namespace TP61
{
    public partial class Form1 : Form
    {
        Bitmap srcImage, tempImage, bwImage;

        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            srcImage = ImageDecoder.DecodeFromFile("Bracket.bmp");
            pictureBox1.Image = srcImage;
            tempImage = srcImage;
        }
    }
}
```

```

private void button4_Click(object sender, EventArgs e)
{

}

private void button5_Click(object sender, EventArgs e)
{
    Int32 thex = 450;
    Int32 yinitial = 150;
    Int32 yfinal = 240;
    using (Graphics g = Graphics.FromImage(tempImage))
    { g.DrawLine(new Pen(Color.Red), new Point(thex, yinitial), new
Point(thex, yfinal)); }

    int ans = 0;
    for (int j = yinitial; j < yfinal; j++)
    {
        Color pixelColor = bwImage.GetPixel(thex, j);
        if (pixelColor.R == 0) { ans = ans + 1; }
    }

    textBox1.Text = ans.ToString();

    pictureBox1.Image = tempImage;
}

private void button6_Click(object sender, EventArgs e)
{

}

private void button2_Click(object sender, EventArgs e)
{
    // create grayscale filter (BT709)
    Grayscale gfilter = new Grayscale(0.2125, 0.7154, 0.0721);
    // apply the filter
    bwImage = gfilter.Apply(srcImage);
}

```

```

        //create filter
        OtsuThreshold filter = new OtsuThreshold();
        // apply the filter
        bwImage = filter.Apply(bwImage);
        pictureBox2.Image = bwImage;

    }

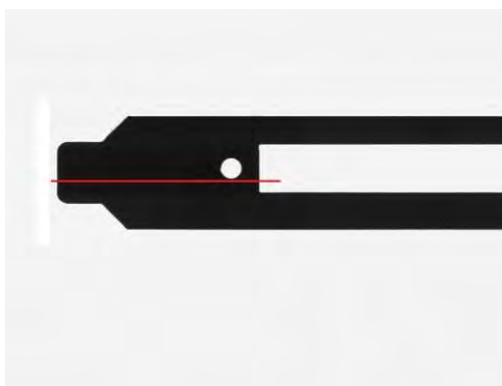
    private void button3_Click(object sender, EventArgs e)
    {
        Int32 thex = 300;
        Int32 yinitial = 100;
        Int32 yfinal = 220;
        using (Graphics g = Graphics.FromImage(tempImage))
        { g.DrawLine(new Pen(Color.Red), new Point(thex, yinitial), new
Point(thex, yfinal));}

        pictureBox1.Image = tempImage;

    }
}

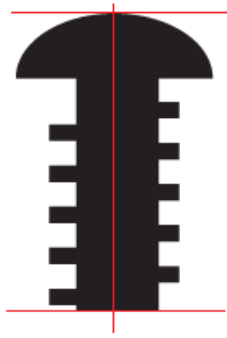
```

1. 找出上臂、中間、下臂的寬度



2. 找出 長度

3. 找出Bolt_0 長度，重點如何找到頂點的(x,y)座標
(TP_61_螺絲_ToStudent)



```
int ans=0;
```

```
for (int j = 100; j < 250; j++)
{
    Color pixelColor = bwImage.GetPixel(400, j);

    if (pixelColor.R < 20)
    {
        ans = ans + 1;
    }
}
```

```
label1.Text = ans.ToString();
```

```
Int32 thex = 600; // 一個良好的習慣
Int32 yinitial = 100;
Int32 yfinal = 400;
```

```
using (Graphics g = Graphics.FromImage(tempImage))
{ g.DrawLine(new Pen(Color.Red), new Point(thex, yinitial), new Point(thex,
yfinal));}
```

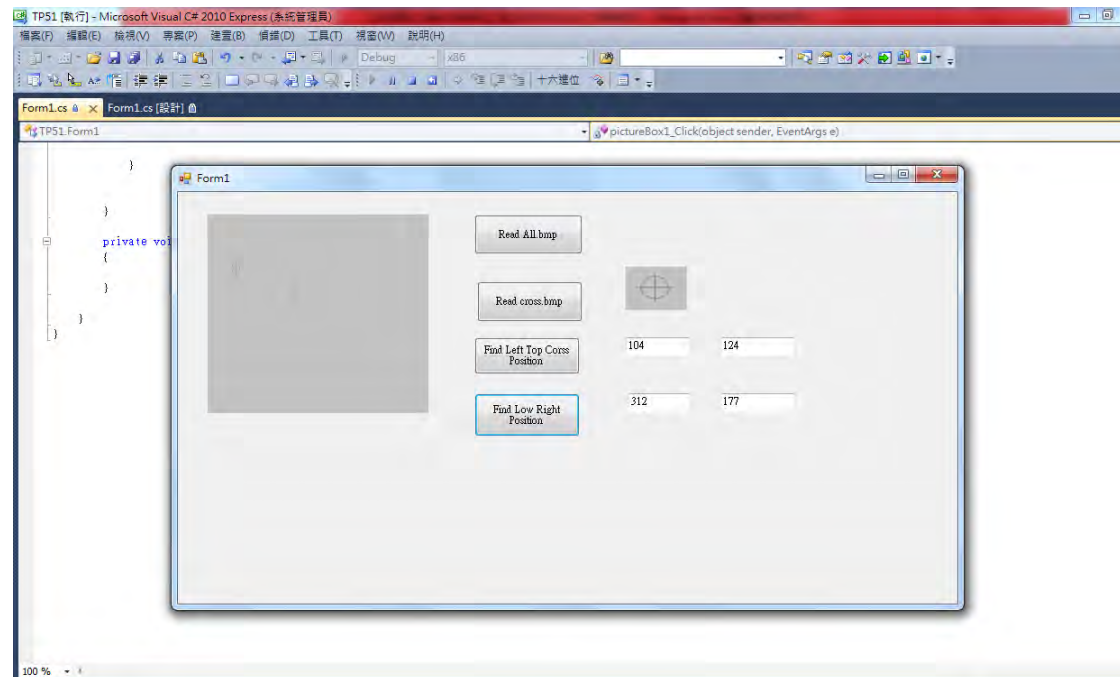
```
int ans=0;
for (int j = yinitial; j < yfinal; j++)
{
    Color pixelColor = bwImage.GetPixel(thex, j);
```

```
        if (pixelColor.R == 0) { ans = ans + 1; }  
    }
```

```
label1.Text = ans.ToString();
```

```
pictureBox1.Image = tempImage;
```

TP51(先跳過)



```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;
```

```
using AForge.Imaging;  
using AForge.Imaging.Filters;  
using AForge.Imaging.Formats;  
using AForge.Math;
```

```
// 完成  
// cross 大小(62,56) 中心 (29,26)
```

```
namespace TP51  
{  
    public partial class Form1 : Form
```

```

{
    Bitmap Allimage, crossimage;
    int ah, aw, ch, cw;

    public Form1()
    {
        InitializeComponent();
    }

    private void button1_Click(object sender, EventArgs e)
    {
        Bitmap temp;
        temp= ImageDecoder.DecodeFromFile("All.bmp");
        // create grayscale filter (BT709)
        Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
        // apply the filter
        Allimage = filter.Apply(temp);
        pictureBox1.Image = Allimage;
        ah = Allimage.Height;
        aw = Allimage.Width;
    }

    private void button2_Click(object sender, EventArgs e)
    {
        Bitmap temp = ImageDecoder.DecodeFromFile("cross.bmp");
        Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
        // apply the filter
        crossimage = filter.Apply(temp);
        pictureBox2.Image = crossimage;
        ch = crossimage.Height;
        cw = crossimage.Width;
    }

    private void button3_Click(object sender, EventArgs e)
    {
        int ii,jj;
        Bitmap temp,result;

```

```

for (ii = 1; ii < aw - cw; ii = ii + 1)
{
    for (jj = 1; jj < ah - ch; jj = jj + 1)
    {
        Crop crfilter = new Crop( new Rectangle( ii, jj, cw, ch ) );
        temp = crfilter.Apply( Allimage );

        Difference difilter = new Difference(crossimage);
        result= difilter.Apply(temp);

        // gather statistics
        ImageStatistics stat = new ImageStatistics(result);
        // get red channel's histogram
        int result_not_black = stat.PixelsCountWithoutBlack;
        // check mean value of red channel
        if (result_not_black < 5)
        {
            int ansii = ii + 29;
            int ansjj = jj + 26;
            textBox1.Text = ansii.ToString();
            textBox2.Text = ansjj.ToString();
            ii = aw - cw;
            jj = ah - ch;
        }
    }
}

private void button4_Click(object sender, EventArgs e)
{
    int ii, jj;
    Bitmap temp, result;

    for (ii = aw - cw; ii > 1; ii = ii - 1)
    {

```

```

for (jj = ah - ch; jj > 1; jj = jj - 1)
{
    Crop crfilter = new Crop(new Rectangle(ii, jj, cw, ch));
    temp = crfilter.Apply(Allimage);

    Difference difilter = new Difference(crossimage);
    result = difilter.Apply(temp);

    // gather statistics
    ImageStatistics stat = new ImageStatistics(result);
    // get red channel's histogram
    int result_not_black = stat.PixelsCountWithoutBlack;
    // check mean value of red channel
    if (result_not_black < 5)
    {
        int ansii = ii + 29;
        int ansjj = jj + 26;
        textBox3.Text = ansii.ToString();
        textBox4.Text = ansjj.ToString();
        ii = 1;
        jj = 1;
    }
}

}

private void pictureBox1_Click(object sender, EventArgs e)
{
}

}
}

```

Lesson 13

Crop and Match

✓ Algorithms

corr2 computes the correlation coefficient using

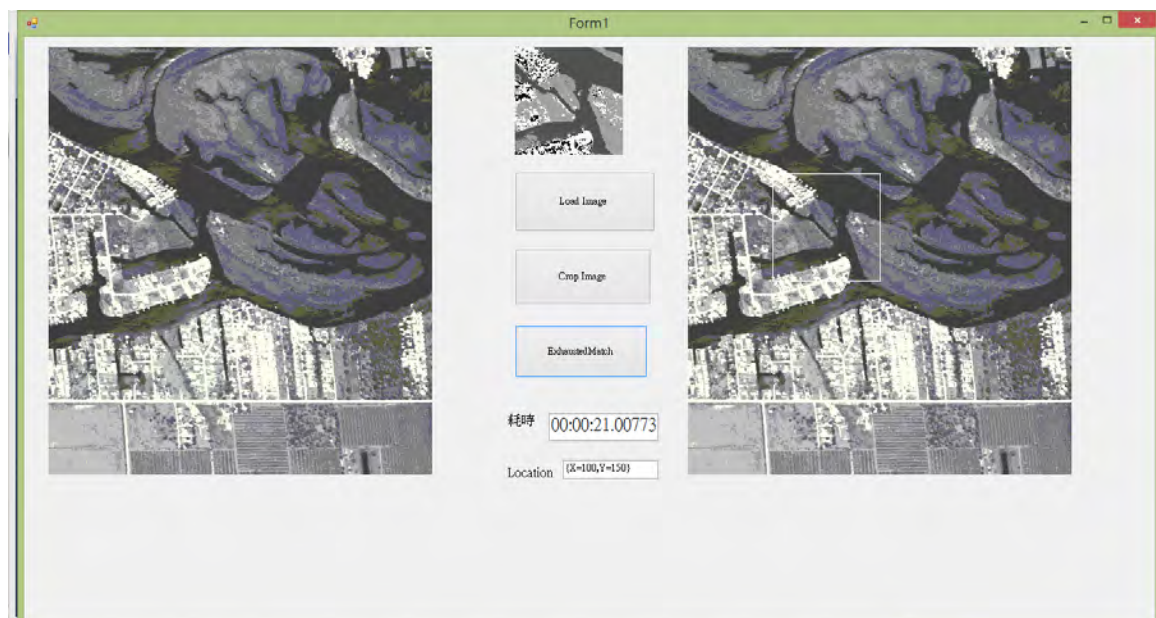
$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \bar{A})^2\right) \left(\sum_m \sum_n (B_{mn} - \bar{B})^2\right)}}$$

where $\bar{A} = \text{mean2}(A)$, and $\bar{B} = \text{mean2}(B)$.

使用 R10.bmp 或 IC_1.bmp

程式 CropAndMatch 原始版

程式 CropAndMatch 原始版_ToStudent



using System;

using System.Collections.Generic;

```

using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

using System.Drawing.Imaging;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;
using System.Diagnostics;

namespace ImageRotate
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            Bitmap sourceImage, templateImage;
            Stopwatch stopWatch = new Stopwatch();

            private void pictureBox1_Click(object sender, EventArgs e)
            {

            }

            private void button1_Click(object sender, EventArgs e)
            {

                openFileDialog1.Filter = "所有檔案(*.*)|*.*";
                if (openFileDialog1.ShowDialog() == DialogResult.OK)
                {
                    sourceImage =

```

```

ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        pictureBox1.Image = sourceImage;
    }

}

private void button2_Click(object sender, EventArgs e)
{
    // create filter
    Crop filter = new Crop( new Rectangle( 100, 150, 128, 128 ) );
    // apply the filter
    templateImage = filter.Apply( sourceImage );
    pictureBox2.Image=templateImage;

}

private void button3_Click(object sender, EventArgs e)
{
    stopWatch.Reset();

    stopWatch.Start();

    // create template matching algorithm's instance
    ExhaustiveTemplateMatching tm = new
ExhaustiveTemplateMatching(0.99f);
    // find all matchings with specified above similarity
    TemplateMatch[] matchings = tm.ProcessImage(sourceImage,
templateImage);

    // highlight found matchings
    BitmapData data = sourceImage.LockBits(
        new Rectangle(0, 0, sourceImage.Width, sourceImage.Height),
        ImageLockMode.ReadWrite, sourceImage.PixelFormat);
    foreach (TemplateMatch m in matchings)
    {
        Drawing.Rectangle(data, m.Rectangle, Color.White);
    }
}

```

```

        textBox2.Text = m.Rectangle.Location.ToString();
        // do something else with matching
    }
    sourceImage.UnlockBits(data);
    pictureBox3.Image = sourceImage;

    stopwatch.Stop();
    // Get the elapsed time as a TimeSpan value.
    TimeSpan ts = stopwatch.Elapsed;
    textBox1.Text = ts.ToString();

}

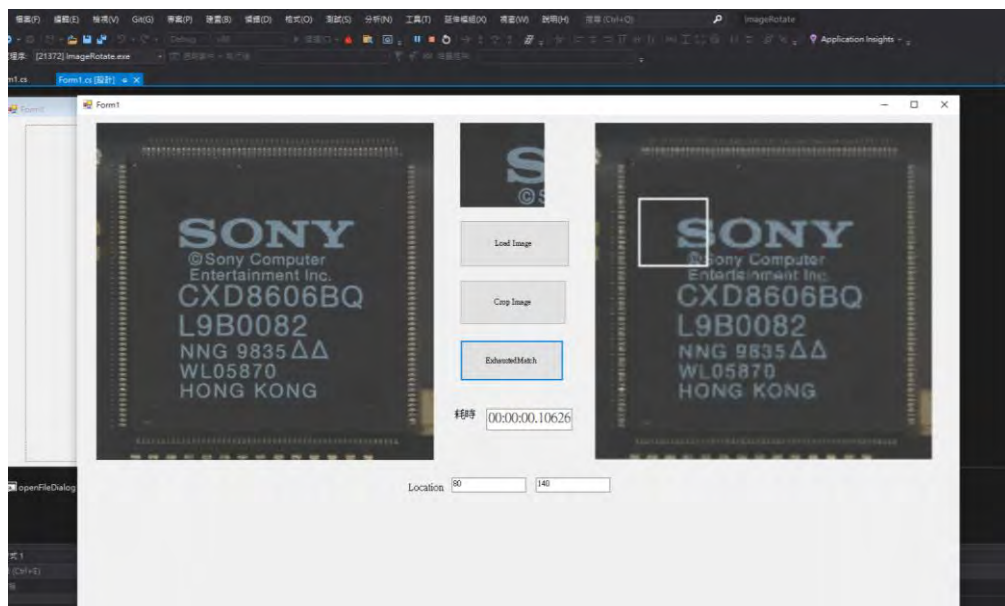
private void pictureBox2_Click(object sender, EventArgs e)
{

}

}
}

```

1. 記錄比對時間
2. 使用 `resize` 將圖形(`sourceImage`, `templateImage`) 縮小 1/4，記錄比對時間，並與未縮圖之情況比較 (程式 `CropAndMatch_快速版_color`，使用 `teaset` 圖)



```

using System;
using System.Collections.Generic;
using System.ComponentModel;

```

```

using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;


using System.Drawing.Imaging;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;
using System.Diagnostics;

namespace ImageRotate
{
    public partial class Form1 : Form
    {
        //最好使用 teaset

        public Form1()
        {
            InitializeComponent();

            Bitmap srcimg, sourceImage, templateImage;
            Stopwatch stopWatch = new Stopwatch();

            double scale=0.25;  /////    縮小 1/5 檢 查//////////

            private void pictureBox1_Click(object sender, EventArgs e)
            {

            }

            private void button1_Click(object sender, EventArgs e)
            {

                openFileDialog1.Filter = "所有檔案(*.*)|*.*";
            }
        }
    }
}

```

```

        if (openFileDialog1.ShowDialog() == DialogResult.OK)
        {
            srcimg =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);

            // create grayscale filter (BT709)
            // Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
            sourceImage = srcimg;
            pictureBox1.Image = sourceImage;

        }

    }

private void button2_Click(object sender, EventArgs e)
{
    // create filter
    Crop filter = new Crop( new Rectangle( 80, 140, 128, 128 ) );
    // apply the filter
    templateImage = filter.Apply( sourceImage );
    pictureBox2.Image=templateImage;

}

private void button3_Click(object sender, EventArgs e)
{
    stopWatch.Reset();
    stopWatch.Start();

    // create filter, Resize
    Int16 s_width = (Int16)(sourceImage.Width * scale);
    Int16 s_height = (Int16)(sourceImage.Height * scale);
    ResizeBilinear filter1 = new ResizeBilinear(s_width, s_height);
    sourceImage = filter1.Apply(sourceImage);

```

```

// create filter, Resize
Int16 t_width = Convert.ToInt16(templateImage.Width * scale);
Int16 t_height = Convert.ToInt16(templateImage.Height * scale);
ResizeBilinear filter2 = new ResizeBilinear(t_width, t_height);
templateImage = filter2.Apply(templateImage);

// create template matching algorithm's instance
ExhaustiveTemplateMatching tm = new
ExhaustiveTemplateMatching(0.97f);
// find all matchings with specified above similarity
TemplateMatch[] matchings = tm.ProcessImage(sourceImage,
templateImage);

// highlight found matchings
BitmapData data = sourceImage.LockBits(
    new Rectangle(0, 0, sourceImage.Width, sourceImage.Height),
    ImageLockMode.ReadWrite, sourceImage.PixelFormat);
foreach (TemplateMatch m in matchings)
{
    Drawing.Rectangle(data, m.Rectangle, Color.White);
    textBox2.Text = Convert.ToInt32
(m.Rectangle.Location.X/scale).ToString();
    textBox3.Text
=Convert.ToInt32( m.Rectangle.Location.Y/scale).ToString();

}
sourceImage.UnlockBits(data);
pictureBox3.Image = sourceImage;

stopWatch.Stop();
TimeSpan ts = stopWatch.Elapsed;
textBox1.Text = ts.ToString();
}

private void pictureBox2_Click(object sender, EventArgs e)
{

```

```

    }

}
}

```

3. 使用 crop、image difference、ImageStastics 之 Gray.mean (程式 CropAndMatch_UseDifference)

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

```

```

using System.Drawing.Imaging;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;
using System.Diagnostics;

```

```

using AForge.Math;
// 使用 lena////////////////////////////////////

```

```

namespace ImageRotate
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
    }
}

```

```

    Bitmap reading, sourceImage, templateImage;
    Stopwatch stopWatch = new Stopwatch();
    int tsize=128;

```

```

private void pictureBox1_Click(object sender, EventArgs e)
{

}

private void button1_Click(object sender, EventArgs e)
{

    openFileDialog1.Filter = "所有檔案(*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        readimg =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);

        Grayscale G_filter = new Grayscale(0.2125, 0.7154, 0.0721);

        sourceImage = G_filter.Apply(readimg);
        pictureBox1.Image = sourceImage;

    }

}

private void button2_Click(object sender, EventArgs e)
{

    Crop filter = new Crop( new Rectangle( 100, 250, tsize, tsize ) );

    templateImage = filter.Apply( sourceImage );
    pictureBox2.Image=templateImage;

}

private void button3_Click(object sender, EventArgs e)
{

```

```

stopWatch.Reset();
stopWatch.Start();

int i, j, ansi, ansj;
ansi = 1;
ansj = 1;
double ans;
ans=999.0;
Bitmap srcimg, result;

for (i = 0; i < sourceImage.Width-tsize; i++)
{
    for (j = 0; j <sourceImage.Height-tsize; j++)
    {

        Crop filter = new Crop(new Rectangle(i, j, tsize, tsize));
        srcimg = filter.Apply(sourceImage);

        Difference D_filter = new Difference(templateImage);
        result = D_filter.Apply(srcimg);

        ImageStatistics stat = new ImageStatistics(result);

        Histogram gray = stat.Gray;
        if (gray.Mean < ans)
        {
            ansi = i;
            ansj = j;
            ans = gray.Mean;
        }
        srcimg.Dispose();
        result.Dispose();
    }
}

Bitmap imgone = new Bitmap(sourceImage);

```

```

using (Graphics g = Graphics.FromImage(imgone))
{
    g.DrawRectangle(new Pen(Color.Red), ansi, ansj, tsize, tsize);
}
pictureBox3.Image = imgone;

stopWatch.Stop();
textBox1.Text = stopWatch.Elapsed.ToString();
textBox2.Text = ansi.ToString() + "    " + ansj.ToString();

}

private void pictureBox2_Click(object sender, EventArgs e)
{
}

}
}

```

有記憶體爆掉的問題所以先取小塊處理

4. 尋找 the_mark 在 F_mark1、F_mark4 的位置 (CropAndMatch_快速版_diff_#定標)

交通標誌比對 (程式：比對_使用 Difference, 比對_使用 Difference_ToStudent))



```
using System;
```

```

using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

```

```

using AForge.Math;
using System.Drawing.Imaging;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;
using System.Diagnostics;
using System.Threading;
using System.IO;

```

```

namespace 比對_使用 Difference

```

```

{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            Bitmap[] temp = new Bitmap[21];

            Bitmap src, srcimg;
            int srch, srcw, noif;

            Stopwatch stp = new Stopwatch();
            private void button1_Click(object sender, EventArgs e)
            {
                // Bitmap srcimg ;
                if (openFileDialog1.ShowDialog() == DialogResult.OK)
                {

```

```

        srcimg =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
    }
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    src = filter.Apply(srcimg);
    srch = src.Height;
    srcw = src.Width;
    pictureBox1.Image = src;
}
private void button2_Click(object sender, EventArgs e)
{
    stp.Reset();
    stp.Start();
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    ResizeBicubic filter_s = new ResizeBicubic(srcw, srch);
    string[] imageFiles = Directory.GetFiles(@"C:\TT" , "*.jpg");
    temp = new Bitmap[imageFiles.Length];
    for (int i = 0; i < imageFiles.Length; i++)
    {
        Bitmap tempimage =
ImageDecoder.DecodeFromFile(imageFiles[i]);
        temp[i] = filter.Apply(tempimage);
        temp[i] = filter_s.Apply(temp[i]);
        // filter_T.ApplyInPlace(temp[i]);
    }
    pictureBox2.Image = temp[20];
    noif = imageFiles.Length;
    textBox1.Text = stp.Elapsed.ToString();
}

private void button3_Click(object sender, EventArgs e)
{
    stp.Reset();
    stp.Start();

    int ansi = 0;
    double ans = 999.0;
    Difference D_filter = new Difference(src);

```

```

        Bitmap result;

        for (int i = 0; i < noif; i++)
        {
            result = D_filter.Apply(temp[i]);
            ImageStatistics stat = new ImageStatistics(result);
            Histogram gray = stat.Gray;
            if (gray.Mean < ans)
            {
                ansi = i;
                ans = gray.Mean;
            }

            result.Dispose();
        }
        pictureBox3.Image = temp[ansi];
        textBox2.Text = stp.Elapsed.ToString();
    }
}
}

```

以下副程式為類似Matlab 的 corr2
 利用它做 交通號誌比對
 (比對_使用 Corr, 比對_使用 Corr_ToStudent)

```

double corr2(Bitmap img1, Bitmap img2)
{
    ImageStatistics stat1 = new ImageStatistics(img1);
    double mean1 = stat1.Gray.Mean;
    ImageStatistics stat2 = new ImageStatistics(img2);
    double mean2 = stat2.Gray.Mean;

    double cor = 0.0;
    double n1 = 0.0;
    double n2 = 0.0;
    double n3 = 0.0;

```

```

for (int i=0;i<srcw;i++)
{
    for(int j=0;j<srch;j++)
    {
        double amn = img1.GetPixel(i, j).G;
        double bmn = img2.GetPixel(i, j).G;
        n1 = n1 + (amn - mean1) * (bmn - mean2);
        n2 = n2 + (amn - mean1) * (amn - mean1);
        n3 = n3 + (bmn - mean2) * (bmn - mean2);
    }
}
cor = n1 / Math.Sqrt(n2 * n3);
return cor;
}

```

▼ Algorithms

corr2 computes the correlation coefficient using

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \bar{A})^2\right) \left(\sum_m \sum_n (B_{mn} - \bar{B})^2\right)}}$$

where $\bar{A} = \text{mean2}(A)$, and $\bar{B} = \text{mean2}(B)$.

使用 EmguCV_Matching

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Diagnostics;
using Emgu.CV;

```

```

using Emgu.Util;
using Emgu.CV.Structure;

namespace Emgu_1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            //Bitmap myBitmap;
            Image<Bgr, Byte> sourceImage;
            Image<Bgr, Byte> templateImage;
            Stopwatch stp=new Stopwatch();

            private void button1_Click(object sender, EventArgs e)
            {
                //OpenFileDialog Openfile = new OpenFileDialog();
                //if (Openfile.ShowDialog() == DialogResult.OK)
                {
                    sourceImage = new Image<Bgr, byte>("IC_1.bmp");
                    //myBitmap = sourceImage.ToBitmap();
                    //pictureBox1.Image = myBitmap;
                    //Image<Bgr, Byte> srcimg = new Image<Bgr,
Byte>(myBitmap);
                    pictureBox1.Image = sourceImage.ToBitmap();
                }
            }

            private void pictureBox1_Click(object sender, EventArgs e)
            {
            }
        }
    }
}

```

```

private void button4_Click(object sender, EventArgs e)
{
    // OpenFileDialog Openfile = new OpenFileDialog();
    //if (Openfile.ShowDialog() == DialogResult.OK)
    {
        templateImage = new Image<Bgr, byte>("IC_1_part.bmp");
        pictureBox2.Image = templateImage.ToBitmap();
    }
}

private void Form1_Load(object sender, EventArgs e)
{
}

private void button2_Click(object sender, EventArgs e)
{
    stp.Reset();
    stp.Start();

    Image<Gray, float> resultImage =
sourceImage.MatchTemplate(templateImage,
Emgu.CV.CvEnum.TM_TYPE.CV_TM_CCOEFF_NORMED);

    float[, ] matches = resultImage.Data;
    for (int x = 0; x < matches.GetLength(1); x++)
    {
        for (int y = 0; y < matches.GetLength(0); y++)
        {
            double matchScore = matches[y, x, 0];
            if (matchScore > 0.95)
            {
                Rectangle rect = new Rectangle(new Point(x, y), new
Size(templateImage.MiplImage.width, templateImage.Height));
                sourceImage.Draw(rect, new Bgr(Color.Blue), 1);
            }
        }
    }
}

```

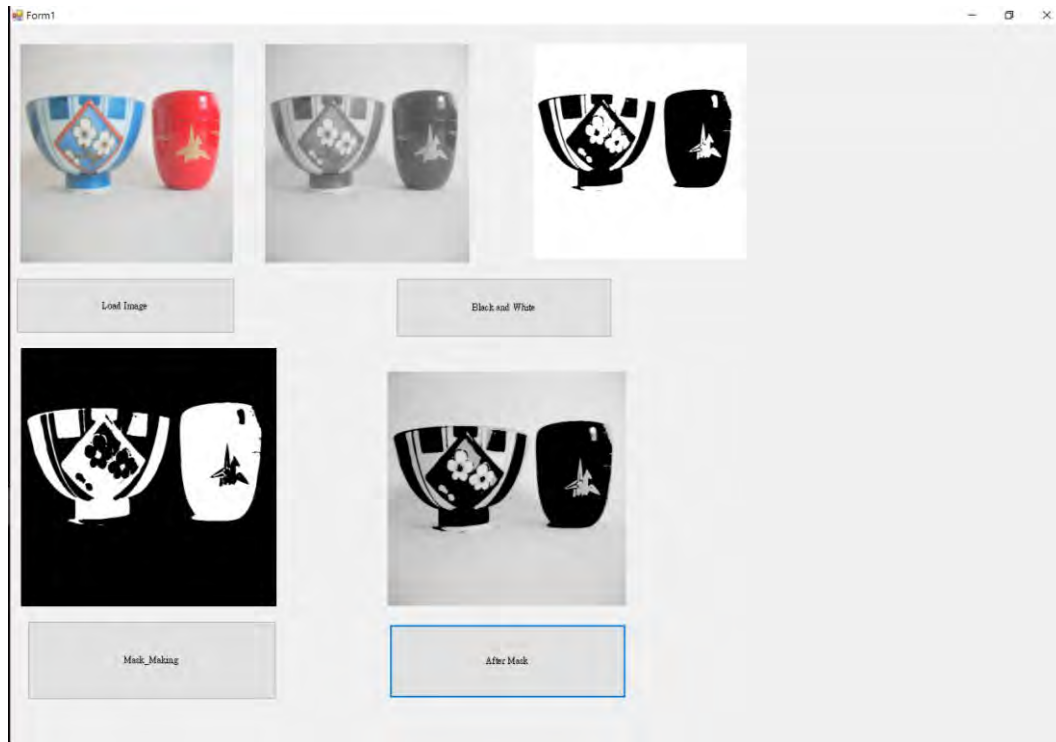
```
    }  
    stp.Stop();  
    label1.Text = stp.Elapsed.ToString();  
    pictureBox3.Image = sourceImage.ToBitmap();  
    }  
    }  
}
```

Lesson 14

Mask 與 Blob 的應用

(Mask 先跳過，直接處理 Blob)

製造 Mask (未完成，只適用 gray image)



```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;  
using AForge;  
using AForge.Imaging;  
using AForge.Imaging.Formats;  
using AForge.Imaging.Filters;
```

```

namespace Mask_Making
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            Bitmap srcimg,grayimg,bwimg,maskimg,dstimg;

            private void button1_Click(object sender, EventArgs e)
            {
                openFileDialog1.Filter = "所有檔案(*.*)|*.*";
                if (openFileDialog1.ShowDialog() == DialogResult.OK)
                {
                    srcimg = ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
                }
                pictureBox1.Image = srcimg;
            }

            private void button2_Click(object sender, EventArgs e)
            {
                // create grayscale filter (BT709)
                Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
                grayimg = filter.Apply(srcimg);
                pictureBox2.Image = grayimg;

                OtsuThreshold filter_2 = new OtsuThreshold();
                bwimg= filter_2.Apply(grayimg);
                pictureBox3.Image = bwimg;
            }

            private void button3_Click(object sender, EventArgs e)
            {
                Invert filterinv = new Invert();
                maskimg = filterinv.Apply(bwimg);
                pictureBox4.Image = maskimg;
            }
        }
    }
}

```

```

    }

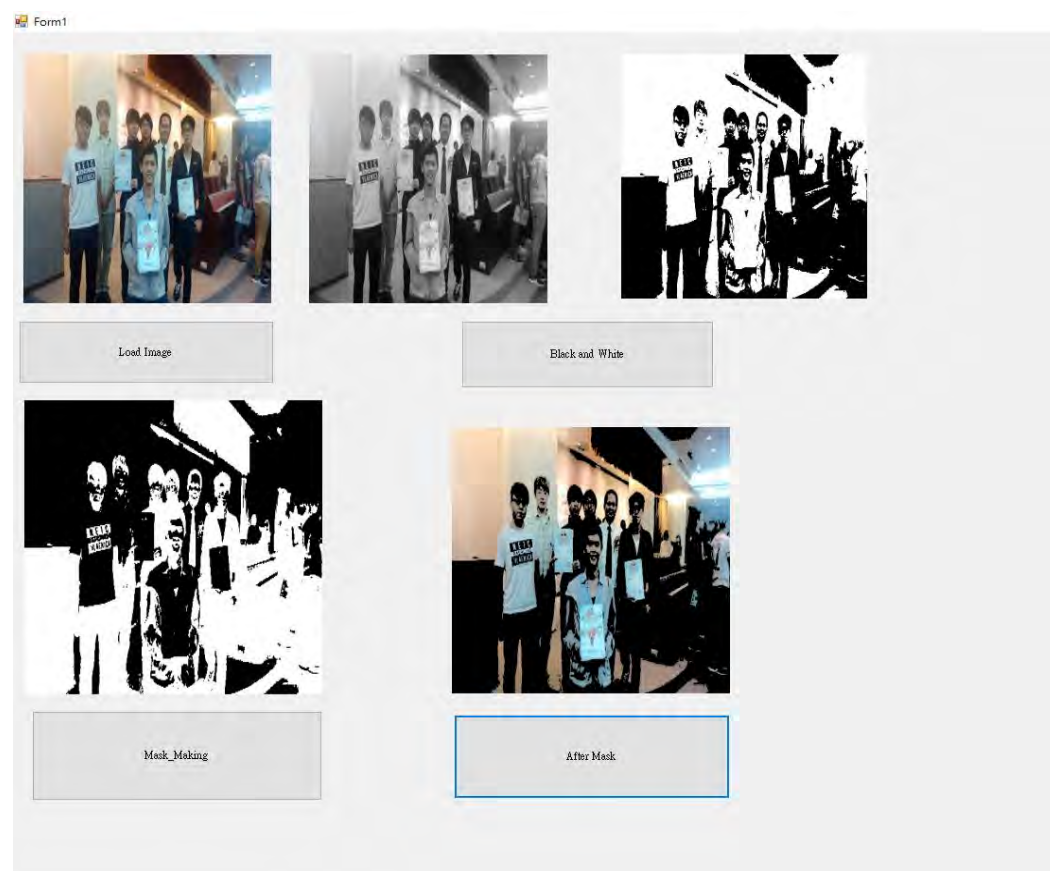
    private void button4_Click(object sender, EventArgs e)
    {

        Subtract filtersub = new Subtract(maskimg);
        // apply the filter
        dstimg= filtersub.Apply(grayimg);
        pictureBox5.Image = dstimg;

    }
}
}_

```

Mask Making (利用相減法，Image Format 要一致)



```

using System;
using System.Collections.Generic;

```

```

using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using AForge;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;

namespace Mask_Making
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            Bitmap srcimg,grayimg,bwimg,maskimg,dstimg;

            private void button1_Click(object sender, EventArgs e)
            {
                openFileDialog1.Filter = "所有檔案(*.*)|*.*";
                if (openFileDialog1.ShowDialog() == DialogResult.OK)
                {
                    srcimg = ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
                }
                pictureBox1.Image = srcimg;
            }

            private void button2_Click(object sender, EventArgs e)
            {
                // create grayscale filter (BT709)
                Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
                grayimg = filter.Apply(srcimg);
            }
        }
    }
}

```

```

        pictureBox2.Image = grayimg;

        OtsuThreshold filter_2 = new OtsuThreshold();
        bwimg= filter_2.Apply(grayimg);
        pictureBox3.Image = bwimg;
    }

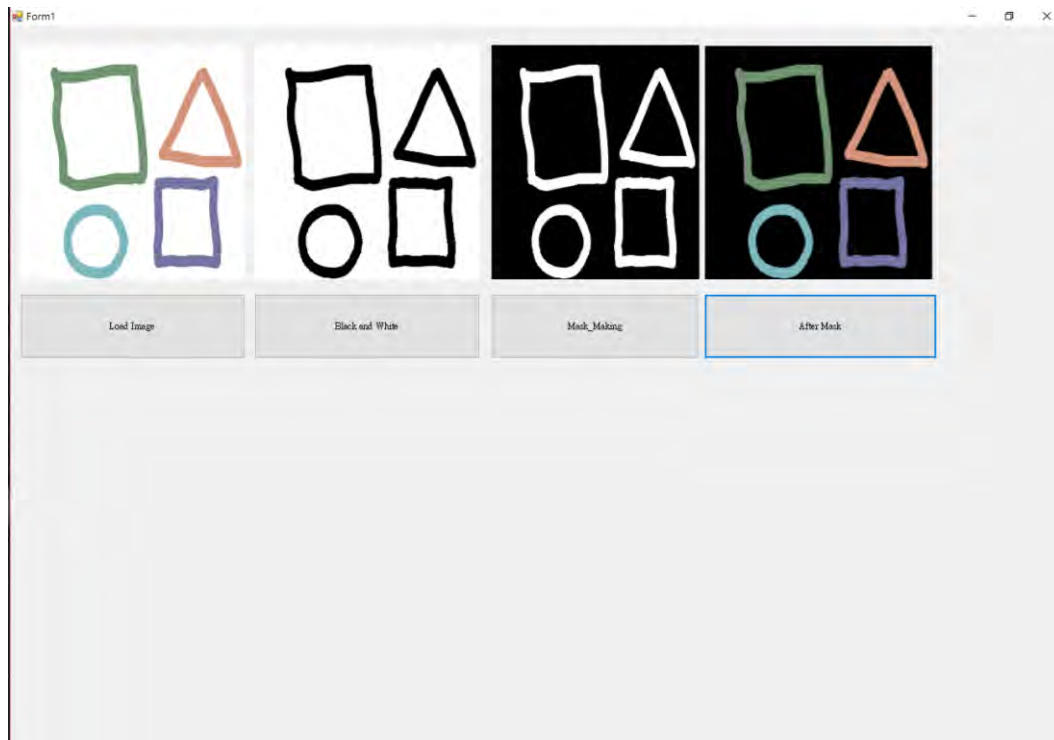
    private void button3_Click(object sender, EventArgs e)
    {
        Invert filterinv = new Invert();
        masking = filterinv.Apply(bwimg);
        pictureBox4.Image = masking;
    }

    private void button4_Click(object sender, EventArgs e)
    {
        // create filter
        GrayscaleToRGB colorfilter = new GrayscaleToRGB();
        // apply the filter
        Bitmap colormasking = colorfilter.Apply(masking);

        Subtract filtersub = new Subtract(colormasking);
        // apply the filter
        dstimg= filtersub.Apply(srcimg);
        pictureBox5.Image = dstimg;
    }
}

```

另一法 製造 Mask (OK) 程式 MaskMakingOK



```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using AForge;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;

namespace Mask_Making
{
    public partial class Form1 : Form
    {
        public Form1()
        {

```

```

        InitializeComponent();
    }
    Bitmap srcimg,grayimg,bwimg,maskimg;

    private void button1_Click(object sender, EventArgs e)
    {
        openFileDialog1.Filter = "所有檔案(*.*)|*. *";
        if (openFileDialog1.ShowDialog() == DialogResult.OK)
        {
            srcimg = ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        }
        pictureBox1.Image = srcimg;
    }

    private void button2_Click(object sender, EventArgs e)
    {
        // create grayscale filter (BT709)
        Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
        grayimg = filter.Apply(srcimg);
        pictureBox2.Image = grayimg;

        // create filter
        OtsuThreshold filter_2 = new OtsuThreshold();
        // apply the filter
        bwimg= filter_2.Apply(grayimg);
        pictureBox2.Image = bwimg;
    }

    private void button3_Click(object sender, EventArgs e)
    {
        Invert filterinv = new Invert();
        maskimg = filterinv.Apply(bwimg);
        pictureBox3.Image = maskimg;
    }

    private void button4_Click(object sender, EventArgs e)
    {

```

```

for (int i=1;i<srcimg.Width;i++)
{
    for(int j=1; j<srcimg.Height;j++)
    {

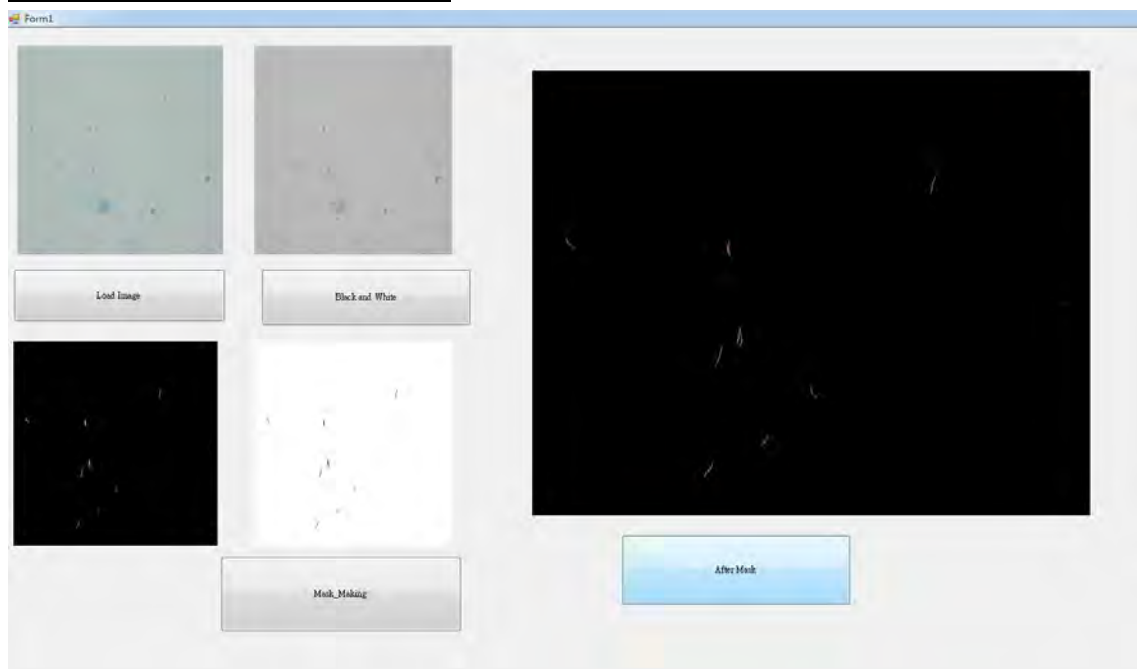
        if (maskimg.GetPixel(i, j).R<10)
        {
            srcimg.SetPixel(i, j, Color.Black);
        }
    }
}

pictureBox4.Image = srcimg;

    }
}
}

```

練習 : "b21.jpg" TB 結核桿菌 的 MASK



```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using AForge;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;

```

```

namespace Mask_Making
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            Bitmap srcimg,grayimg,bwimg,maskimg,dstimg;

            private void openFileDialog2_FileOk(object sender, CancelEventArgs e)
            {

            }

            private void button1_Click(object sender, EventArgs e)
            {

                srcimg = new Bitmap("b21.jpg");
                pictureBox1.Image = srcimg;

            }

```

```

private void button2_Click(object sender, EventArgs e)
{
    // 轉 gray image
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    grayimg = filter.Apply(srcimg);
    pictureBox2.Image = grayimg;

    // 取紅色
    ExtractChannel filterR = new ExtractChannel(RGB.R);
    // apply the filter
    Bitmap srcimgR = filterR.Apply(srcimg);

    //紅色減灰階
    Subtract filtersub = new Subtract(grayimg);
    // apply the filter
    Bitmap pre_mask = filtersub.Apply(srcimgR);

    OtsuThreshold filter_2 = new OtsuThreshold();
    bwimg= filter_2.Apply(pre_mask);
    pictureBox3.Image = bwimg;
}

private void button3_Click(object sender, EventArgs e)
{
    Invert filterinv = new Invert();
    maskimg = filterinv.Apply(bwimg);
    pictureBox4.Image = maskimg;
}

private void button4_Click(object sender, EventArgs e)
{
    // create filter
    GrayscaleToRGB colorfilter = new GrayscaleToRGB();
    // apply the filter

```

```
Bitmap colormasking = colorfilter.Apply(maskimg);
```

```
Subtract filtersub = new Subtract(colormasking);
```

```
// apply the filter
```

```
dstimg= filtersub.Apply(srcimg);
```

```
pictureBox5.Image = dstimg;
```

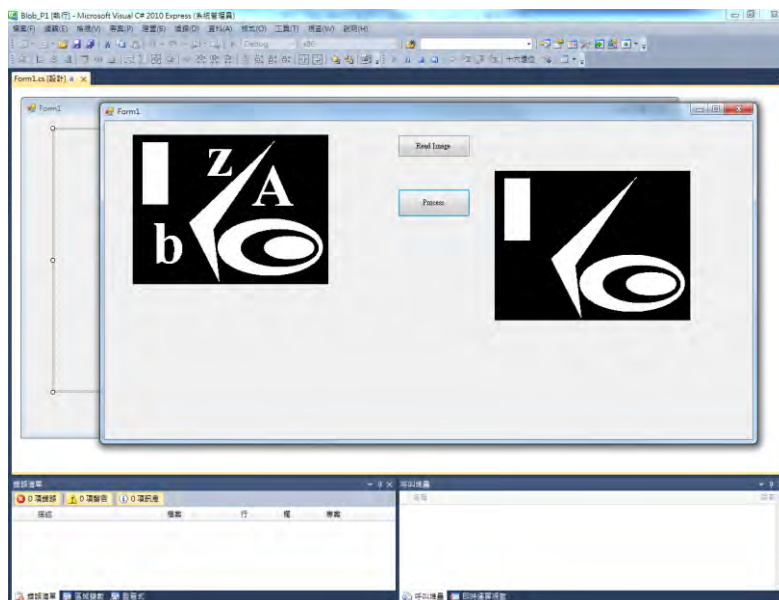
```
}
```

```
}
```

```
}
```

程式 Blob_P1

Blob 的應用



```
using System;
```

```
using System.Collections.Generic;
```

```
using System.ComponentModel;
```

```
using System.Data;
```

```
using System.Drawing;
```

```
using System.Linq;
```

```
using System.Text;
```

```
using System.Windows.Forms;
```

```
using AForge.Imaging;
```

```
using AForge.Imaging.Filters;
```

```

using AForge.Imaging.Formats;
using AForge.Math;

namespace Blob_P1
{
    public partial class Form1 : Form
    {
        Bitmap image;
        public Form1()
        {
            InitializeComponent();
        }

        private void ReadImage_Click(object sender, EventArgs e)
        {
            Bitmap temp;
            temp = ImageDecoder.DecodeFromFile("BlobSample.jpg");
            // create grayscale filter (BT709)
            Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
            // apply the filter
            image = filter.Apply(temp);
            Threshold filter1 = new Threshold(100);
            // apply the filter
            filter1.ApplyInPlace(image);
            pictureBox1.Image = image;
        }

        private void Process_Click(object sender, EventArgs e)
        {
            // create filter
            BlobsFiltering filter = new BlobsFiltering();
            // configure filter
            filter.CoupledSizeFiltering = true;
            filter.MinWidth = 70;
            filter.MinHeight = 70;
            // apply the filter

```

```

        Bitmap dstImage=filter.Apply(image);
        pictureBox2.Image = dstImage;

    }
}

```

習題：求各個 **Blob** 面積與其對應之中心點座標(程式 **Blob_P1_計算面積**)

註：

// create an instance of blob counter algorithm

BlobCounterBase **bc** = new BlobCounter();

//The class is abstract and serves as a base for different blob counting algorithms.

// set filtering options

bc.FilterBlobs = true;

bc.MinWidth = 5;

bc.MinHeight = 5;

// set ordering options

bc.ObjectsOrder = ObjectsOrder.Area;

bc.ProcessImage(image); // process binary image

Blob[] blobs = bc.GetObjectsInformation();

textBox1.Text = blobs[0].Area.ToString();// extract the biggest blob

textBox2.Text = bc.ObjectsCount.ToString();//總個數

for (Int ii = 0; ii < **bc.ObjectsCount**; ii++)

{

listBox1.Items.Add(blobs[ii].Area.ToString());

listBox2.-----重心

listBox3-----Rectangles

}

習題: 找螺帽之面積，孔洞面積，半徑

(程式:Blob_P1_計算面積_針對螺帽_Nut_0)

(程式:Blob_P1_計算面積_針對螺帽_Nut_0_ToStudent)

(程式:Blob_P1_計算面積_針對螺帽_Nut_0_ToStudent_2)



```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;
```

```
using AForge.Imaging;  
using AForge.Imaging.Filters;  
using AForge.Imaging.Formats;  
using AForge.Math;
```

```
namespace Blob_P1  
{  
    public partial class Form1 : Form  
    {  
        Bitmap image,temp, bwimg, dstimg;  
        public Form1()  
        {  
            InitializeComponent();  
        }  
  
        private void ReadImage_Click(object sender, EventArgs e)
```

```

    {

        openFileDialog1.Filter = "所有檔案(*.*)|*.*";
        if (openFileDialog1.ShowDialog() == DialogResult.OK)
        {
            temp =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        }
        pictureBox1.Image = temp;

    }

private void Process_Click(object sender, EventArgs e)
{

    Grayscale filter_0 = new Grayscale(0.2125, 0.7154, 0.0721);
    image = filter_0.Apply(temp);

    Threshold filter1 = new Threshold(200);
    filter1.ApplyInPlace(image);

    Invert filter_2 = new Invert();
    filter_2.ApplyInPlace(image);

    BlobsFiltering filter = new BlobsFiltering();
    filter.CoupledSizeFiltering = true;
    filter.MinWidth = 5;
    filter.MinHeight = 5;
    Bitmap dstImage=filter.Apply(image);
    pictureBox2.Image = dstImage;

}

private void button1_Click(object sender, EventArgs e)

```

```

    {
        // create an instance of blob counter algorithm
        BlobCounterBase bc = new BlobCounter();
        //The class is abstract and serves as a base for different blob counting
        algorithms.

        // set filtering options
        bc.FilterBlobs = true;
        bc.MinWidth = 5;
        bc.MinHeight = 5;
        // set ordering options
        bc.ObjectsOrder = ObjectsOrder.Area;

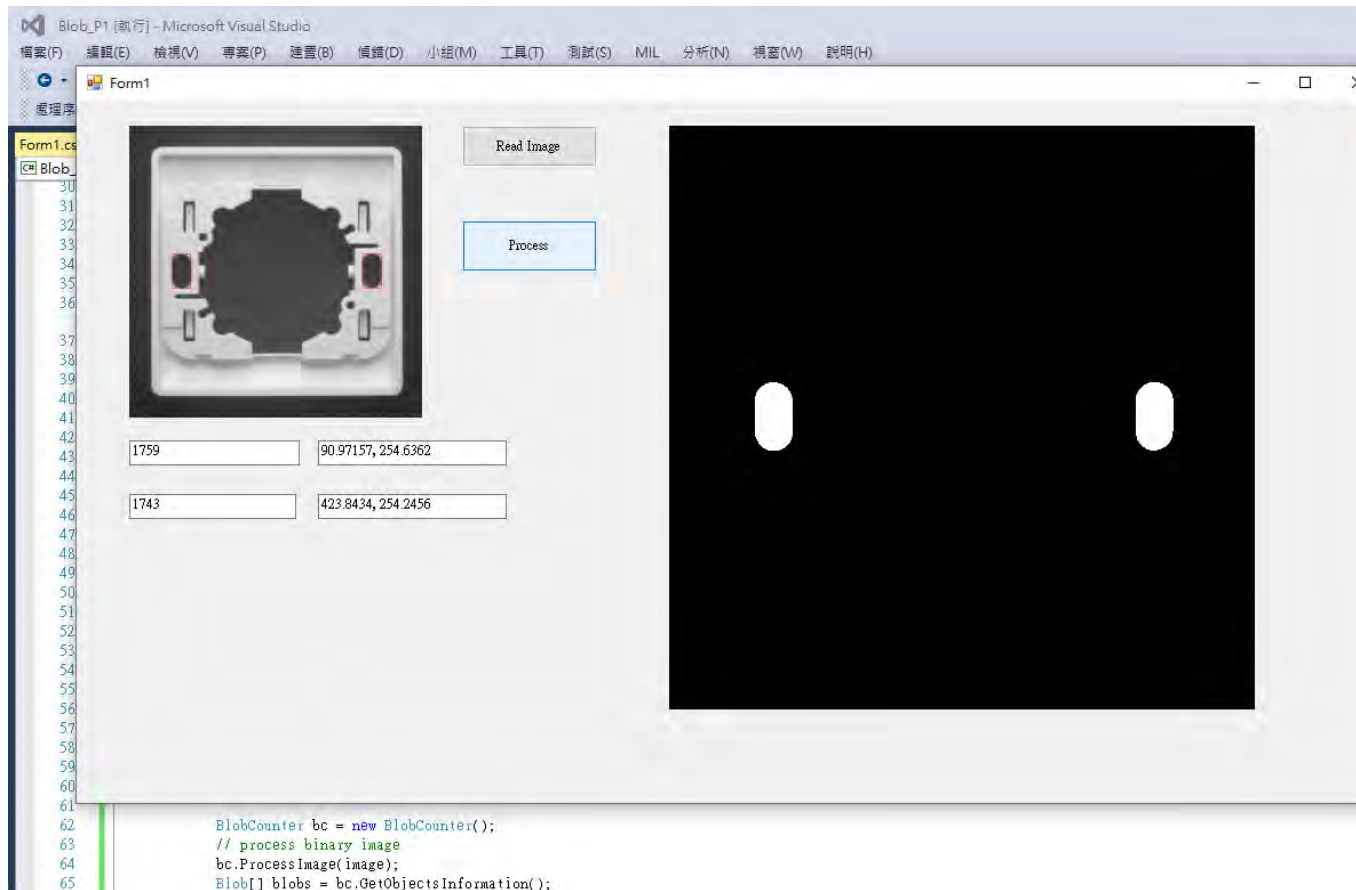
        bc.ProcessImage(image); // process binary image
        Blob[] blobs = bc.GetObjectsInformation();

        textBox1.Text = blobs[0].Area.ToString(); // extract the biggest blob
        Crop cropfilter = new Crop(blobs[0].Rectangle);
        dstimg = cropfilter.Apply(image);
        pictureBox2.Image = dstimg;

        using (Graphics g = Graphics.FromImage(temp))
        {
            g.DrawRectangle(new Pen(Color.Red),blobs[0].Rectangle);
        }

        pictureBox1.Image = temp;
    }
}

```



找出特定目標之座標面積

(程式 Blob_Switch1)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
using AForge;
using System.Drawing.Imaging;
using AForge.Imaging;
using AForge.Imaging.Filters;
```

```

using AForge.Imaging.Formats;
using AForge.Math;

namespace Blob_P1
{
    public partial class Form1 : Form
    {
        Bitmap temp,image;
        public Form1()
        {
            InitializeComponent();
        }

        private void ReadImage_Click(object sender, EventArgs e)
        {
            //Bitmap temp;
            temp = ImageDecoder.DecodeFromFile("Switch1.bmp");

            pictureBox1.Image = temp;
        }

        private void Process_Click(object sender, EventArgs e)
        {
            // create grayscale filter (BT709)
            Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
            // apply the filter
            image = filter.Apply(temp);
            Threshold filter1 = new Threshold(100);
            // apply the filter
            filter1.ApplyInPlace(image);

            Invert filter2 = new Invert();
            filter2.ApplyInPlace(image);

            // create filter
            BlobsFiltering filter3 = new BlobsFiltering();
            // configure filter

```

```

filter3.CoupledSizeFiltering = true;
filter3.MinWidth = 60;
filter3.MinHeight = 60;
filter3.MaxHeight = 200;
filter3.MaxWidth = 200;
filter3.ApplyInPlace(image);
pictureBox2.Image = image;

```

```

BlobCounter bc = new BlobCounter();
bc.ObjectsOrder = ObjectsOrder.Area;
bc.ProcessImage(image);
Blob[] blobs = bc.GetObjectsInformation();

```

```

textBox1.Text = blobs[0].Area.ToString();// extract the biggest blob
textBox2.Text = blobs[1].Area.ToString();// extract the second biggest

```

blob

```

textBox3.Text = blobs[0].CenterOfGravity.ToString();// extract the

```

biggest blob

```

textBox4.Text = blobs[1].CenterOfGravity.ToString();// extract the

```

second biggest blob

```

BitmapData data = temp.LockBits(
    new Rectangle(0, 0, temp.Width, temp.Height),
    ImageLockMode.ReadWrite, temp.PixelFormat);

```

```

Drawing.Rectangle(data, blobs[0].Rectangle, Color.Red);
Drawing.Rectangle(data, blobs[1].Rectangle, Color.Red);

```

```

temp.UnlockBits(data);
pictureBox1.Image = temp;

```

```

    }

```

```

}

```

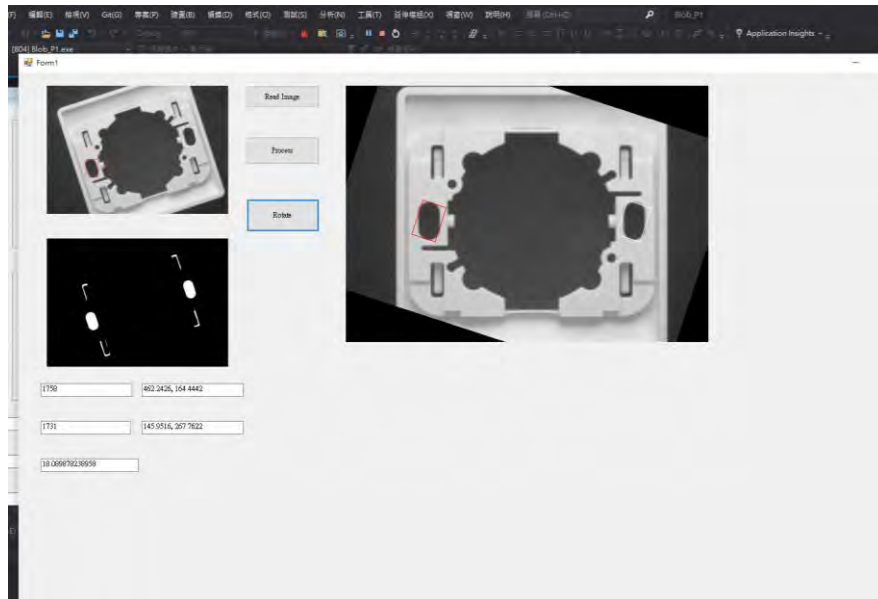
```

}

```

1. 練習： 找出最大孔洞面積、座標(修改 程式 Blob_Switch1)

2. 練習： 利用孔洞的座標，將 Switch2 轉正
(程式 Blob_Switch2_Rotate)



```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;
```

```
using AForge;  
using System.Drawing.Imaging;  
using AForge.Imaging;  
using AForge.Imaging.Filters;  
using AForge.Imaging.Formats;  
using AForge.Math;
```

```
namespace Blob_P1  
{  
    public partial class Form1 : Form  
    {  
        Bitmap temp,image;
```

```

double angle;
public Form1()
{
    InitializeComponent();
}

private void ReadImage_Click(object sender, EventArgs e)
{
    //Bitmap temp;
    temp = ImageDecoder.DecodeFromFile("Switch2.bmp");
    pictureBox1.Image = temp;
    pictureBox3.Image = temp;
}

private void Process_Click(object sender, EventArgs e)
{
    // create grayscale filter (BT709)
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    // apply the filter
    image = filter.Apply(temp);
    Threshold filter1 = new Threshold(100);
    // apply the filter
    filter1.ApplyInPlace(image);

    Invert filter2 = new Invert();
    filter2.ApplyInPlace(image);

    // create filter
    BlobsFiltering filter3 = new BlobsFiltering();
    // configure filter
    filter3.CoupledSizeFiltering = true;
    filter3.MinWidth = 20;
    filter3.MinHeight = 20;
    filter3.MaxHeight = 100;
    filter3.MaxWidth = 100;
    filter3.ApplyInPlace(image);
}

```

```

pictureBox2.Image = image;

BlobCounter bc = new BlobCounter();
bc.ObjectsOrder = ObjectsOrder.Area;
bc.ProcessImage(image);
Blob[] blobs = bc.GetObjectsInformation();

textBox1.Text = blobs[0].Area.ToString();// extract the biggest blob
textBox2.Text = blobs[1].Area.ToString();// extract the second biggest
blob

textBox3.Text = blobs[0].CenterOfGravity.ToString();// extract the
biggest blob
textBox4.Text = blobs[1].CenterOfGravity.ToString();// extract the
second biggest blob

angle = Math.Atan2(blobs[0].CenterOfGravity.Y -
blobs[1].CenterOfGravity.Y, blobs[1].CenterOfGravity.X
-blobs[0].CenterOfGravity.X);
angle = angle * 180 / Math.PI;
textBox5.Text = angle.ToString();

BitmapData data = temp.LockBits(
    new Rectangle(0, 0, temp.Width, temp.Height),
    ImageLockMode.ReadWrite, temp.PixelFormat);

Drawing.Rectangle(data, blobs[0].Rectangle, Color.Red);
Drawing.Rectangle(data, blobs[1].Rectangle, Color.White);

temp.UnlockBits(data);
pictureBox1.Image = temp;

}
private void button1_Click(object sender, EventArgs e)
{
    // create filter - rotate for 30 degrees keeping original image size

```

```

RotateBicubic filter = new RotateBicubic(-angle,true);
// apply the filter
Bitmap newImage = filter.Apply(temp);
pictureBox3.Image = newImage;

```

```

}

```

```

}

```

```

}

```

3.練習: 尋找邊緣，再旋轉

(程式: Edge_Rotate, Edge_Rotate_ToStudent)



```

int the_y = 60;
int x_initial = 30;
int x_final = 200;

int x1 = 0;
int y1 = 0;

for (int i=x_initial;i<x_final;i++)
{
    Color i_color = temp.GetPixel(i, the_y);
    if( i_color.R>120)

```

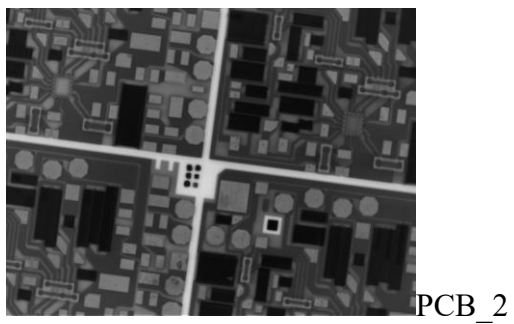
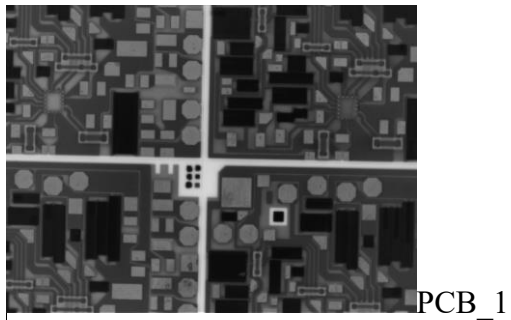
```

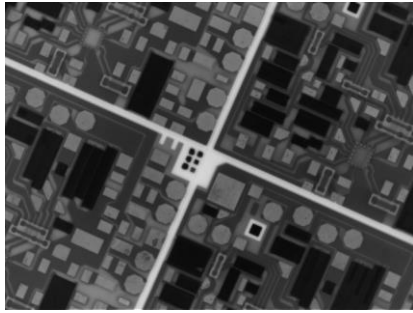
        {
            x1 = i;
            y1 = the_y;
            goto Found_1;
        }
    }
    Found_1:
    textBox1.Text = x1.ToString();
    textBox2.Text = y1.ToString();

    using (Graphics g = Graphics.FromImage(temp))
    { g.DrawLine(new Pen(Color.Red), new
System.Drawing.Point(x_initial, the_y), new System.Drawing.Point(x_final,
the_y)); }

    pictureBox3.Image = temp;

```





PCB_3

(1) 將 PCB_2 與 PCB_3 轉正，如 PCB_1

(2) 求 BlobSample.jpg 的 Rectangle ?

(3)求下圖板手之 (a) 中間部位之寬度

(b)總面積(可能需用到 FillHoles filter, and invert filter)

(c) Bounding Box

(程式:Blob_計算面積_讀圖_針對板手 p5)

(利用 Blob_計算面積_讀圖_針對螺帽_ToStudent 改寫)



計算 Blob ，可以輸出界定的範圍

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

using AForge.Imaging;
using AForge.Imaging.Filters;
using AForge.Imaging.Formats;
using AForge.Math;

namespace Blob_P1
{
    public partial class Form1 : Form

    {
        Bitmap image;
        public Form1()
        {
            InitializeComponent();

        }

        private void ReadImage_Click(object sender, EventArgs e)
        {
            Bitmap temp;
            temp = ImageDecoder.DecodeFromFile("BlobSample.jpg");
            // create grayscale filter (BT709)
            Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
            // apply the filter
            image = filter.Apply(temp);
            Threshold filter1 = new Threshold(100);
            // apply the filter
            filter1.ApplyInPlace(image);
            pictureBox1.Image = image;
        }
    }
}

```

```

private void Process_Click(object sender, EventArgs e)
{
    // create filter
    BlobsFiltering filter = new BlobsFiltering();
    // configure filter
    filter.CoupledSizeFiltering = true;
    filter.MinWidth = 70;
    filter.MinHeight = 70;
    // apply the filter
    Bitmap dstImage=filter.Apply(image);
    pictureBox2.Image = dstImage;
}

private void button1_Click(object sender, EventArgs e)
{
    // create an instance of blob counter algorithm
    BlobCounterBase bc = new BlobCounter();
    //The class is abstract and serves as a base for different blob counting
algorithms.
    // set filtering options
    bc.FilterBlobs = true;
    bc.MinWidth = 5;
    bc.MinHeight = 5;
    // set ordering options
    bc.ObjectsOrder = ObjectsOrder.Area;

    bc.ProcessImage(image); // process binary image
    Blob[] blobs = bc.GetObjectsInformation();

    textBox1.Text = blobs[0].Area.ToString();// extract the biggest blob

    textBox2.Text = bc.ObjectsCount.ToString();//總個數

    Int32 totalarea = 0;
    for (int ii = 0; ii < bc.ObjectsCount; ii++)

```

```

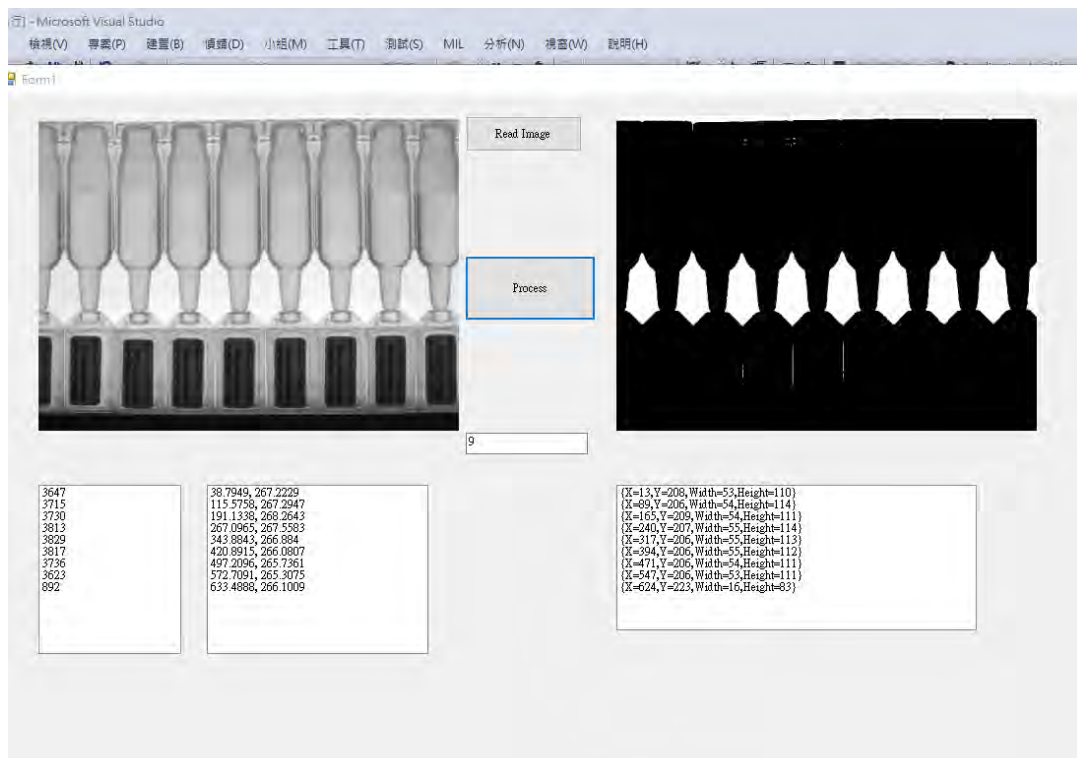
    {
        totalarea = totalarea + blobs[ii].Area;

        listBox1.Items.Add(blobs[ii].Area.ToString());
        listBox2.Items.Add(blobs[ii].CenterOfGravity.ToString());
        listBox3.Items.Add(blobs[ii].Rectangle.ToString());
    }
}
}
}

```

尋找空白處的座標 (程式 Blob_藥水_找空白處)





```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
using AForge.Imaging;
using AForge.Imaging.Filters;
using AForge.Imaging.Formats;
```

```
namespace Blob_P1
{
    public partial class Form1 : Form

    {

        public Form1()
        {
            InitializeComponent();
```

```
}
```

```
Bitmap temp, image;  
private void ReadImage_Click(object sender, EventArgs e)  
{  
    temp = ImageDecoder.DecodeFromFile("ampoules_01.png");  
    pictureBox1.Image = temp;  
  
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);  
    image = filter.Apply(temp);  
    Threshold filter1 = new Threshold(220);  
    filter1.ApplyInPlace(image);  
    pictureBox2.Image = image;  
}
```

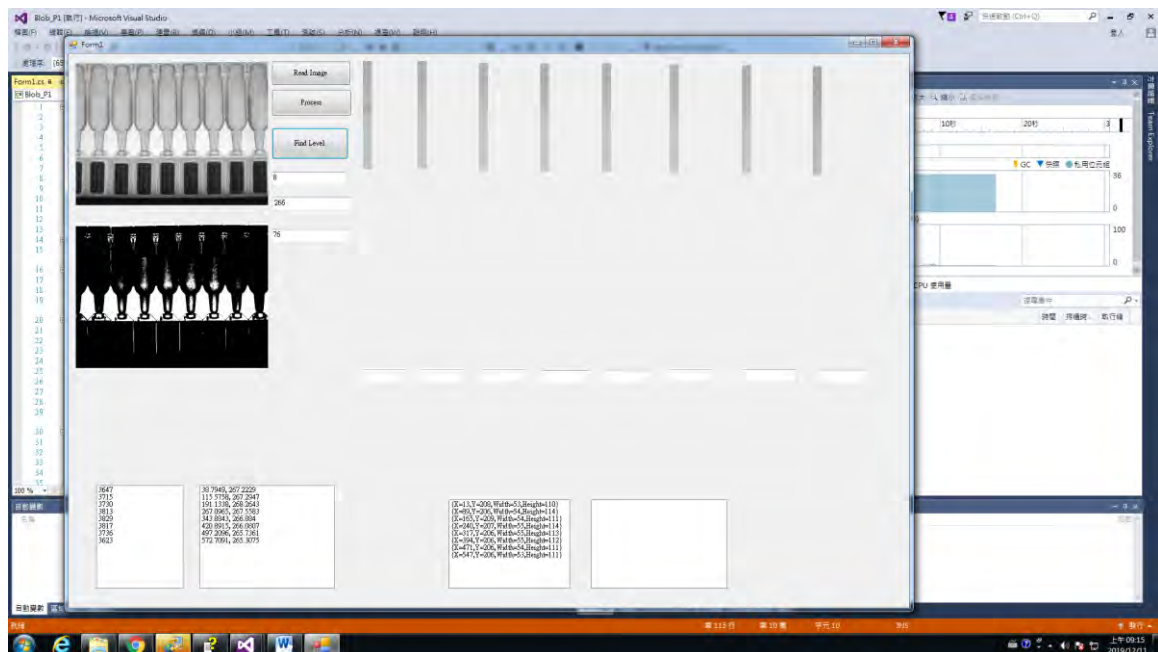
```
private void button1_Click(object sender, EventArgs e)  
{  
    BlobCounterBase bc = new BlobCounter();  
    bc.FilterBlobs = true;  
    bc.MinWidth = 40;  
    //bc.MinHeight = 10;  
    bc.MaxHeight = 200;  
    bc.MaxWidth = 200;  
    // set ordering options  
  
    bc.ObjectsOrder = ObjectsOrder.XY; ;  
  
    bc.ProcessImage(image); // process binary image  
    Blob[] blobs = bc.GetObjectsInformation();  
    textBox1.Text = bc.ObjectsCount.ToString();//總個數  
  
    for (int ii = 0; ii < bc.ObjectsCount; ii++)  
    {  
        listBox1.Items.Add(blobs[ii].Area);  
        listBox2.Items.Add(blobs[ii].CenterOfGravity.ToString());  
        listBox3.Items.Add(blobs[ii].Rectangle.ToString());  
    }  
}
```

```

    }
}
}

```

將獨立藥水 程式 (Blob_P1_藥水_All5_part_ToStudent)



```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

using AForge.Imaging;
using AForge.Imaging.Filters;
using AForge.Imaging.Formats;

```

```

namespace Blob_P1
{
    public partial class Form1 : Form

    {

        public Form1()
        {
            InitializeComponent();

            Bitmap temp, image;

            Bitmap[] sub = new Bitmap[10];
            double[] ans = new double[10];
            int count;

            private void ReadImage_Click(object sender, EventArgs e)
            {
                openFileDialog1.Filter = "所有檔案(*.*)|*.*";
                if (openFileDialog1.ShowDialog() == DialogResult.OK)
                {
                    temp =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
                }
                pictureBox1.Image = temp;

                Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
                image = filter.Apply(temp);
                Threshold filter1 = new Threshold(220);
                filter1.ApplyInPlace(image);
                pictureBox2.Image = image;
            }

            private void button1_Click(object sender, EventArgs e)
            {

```

```

BlobCounterBase bc = new BlobCounter();
bc.FilterBlobs = true;
bc.MinWidth = 30;
bc.MinHeight = 30;
bc.MaxHeight = 150;
bc.MaxWidth = 150;

bc.ObjectsOrder = ObjectsOrder.XY;

bc.ProcessImage(image); // process binary image
Blob[] blobs = bc.GetObjectsInformation();

count = bc.ObjectsCount;
textBox1.Text = count.ToString();//總個數

for (int ii = 0; ii < count; ii++)
{

    listBox1.Items.Add(blobs[ii].Area);
    listBox2.Items.Add(blobs[ii].CenterOfGravity.ToString());
    listBox3.Items.Add(blobs[ii].Rectangle.ToString());
}

```

////////////////////////////////////

```

Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
image = filter.Apply(temp);
Threshold filter1 = new Threshold(220);
filter1.ApplyInPlace(image);
pictureBox2.Image = image;

for (int i=0;i<count;i++)
{

    Crop Cfilter = new Crop(new

```

```

Rectangle((int)blobs[i].CenterOfGravity.X+25, 30, 15, 180));
        sub[i] = Cfilter.Apply(temp);
    }

    pictureBox3.Image = sub[0];
    pictureBox4.Image = sub[1];
    pictureBox5.Image = sub[2];
    pictureBox6.Image = sub[3];
    pictureBox7.Image = sub[4];
    pictureBox8.Image = sub[5];
    pictureBox9.Image = sub[6];
    pictureBox10.Image = sub[7];
}

private void button2_Click(object sender, EventArgs e)
{
}
}
}

```

習題: 量測藥水高度

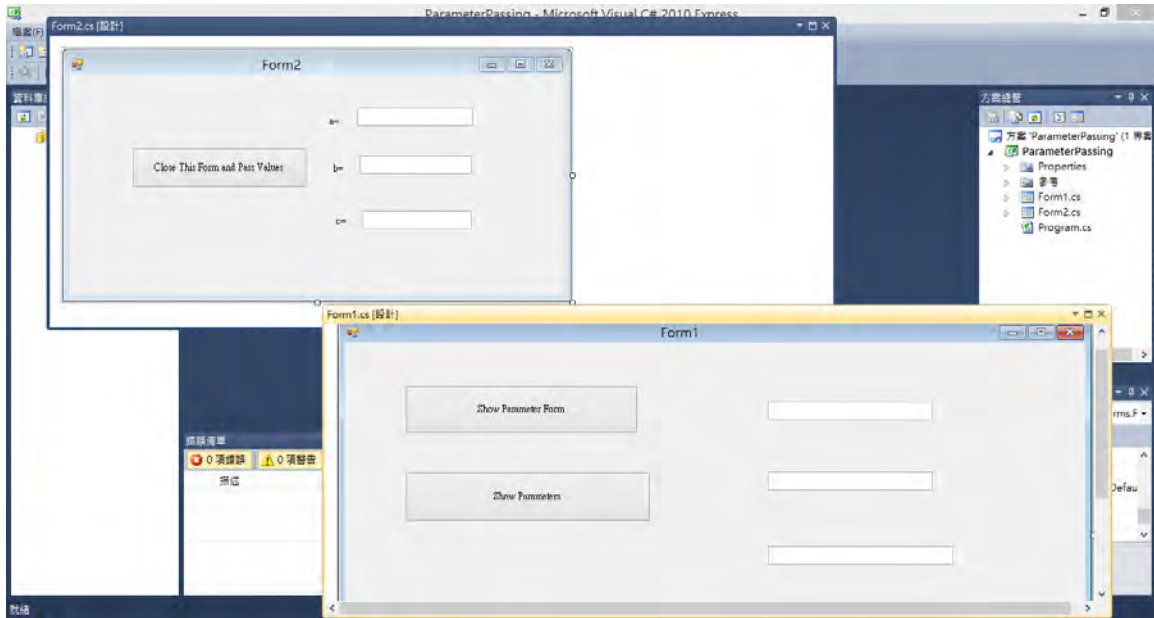
(完整程式 Blob_P1_藥水_All_1, Blob_P1_藥水_All_2)

Lesson 15

在不同的 Form 之間傳遞參數

在不同的 Form 之間傳遞參數

程式 **Passing Parameter**



```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;
```

```
namespace ParameterPassing  
{  
  
    public partial class Form1 : Form  
    {  
  
        Form2 f2=new Form2();  
  
        public Form1()  
        {  
            InitializeComponent();  

```

```

    }

    private void richTextBox1_TextChanged(object sender, EventArgs e)
    {

    }

    private void button1_Click(object sender, EventArgs e)
    {
        //f2.ShowDialog();

        textBox1.Text = f2.aa;
        textBox2.Text = f2.bb;
        textBox3.Text = f2.cc;
    }

    private void button2_Click(object sender, EventArgs e)
    {
        f2.Show();
    }
}

```

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace ParameterPassing
{
    public partial class Form2 : Form
    {
        public string aa, bb, cc;

        public Form2()

```

```

    {
        InitializeComponent();
    }

    private void Form2_Load(object sender, EventArgs e)
    {

    }

    private void button1_Click(object sender, EventArgs e)
    {
        aa = textBox1.Text;
        bb = textBox2.Text;
        cc = textBox3.Text;

        this.Hide();
    }
}

```

-
1. 習題: 製造 1. Form2 輸入 a
 2. Form 3 輸入 b
 3. Form 1 計算 $c=a*b$
 (程式 `ParameterPassing_2`)
 2. 習題: 開新表單(Form)，驗證密碼
(程式 `PasswordPassing`)
-

找圖案最頂點

程式: `Get Colorpixel_and Find the top`

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;

```

```

using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace Get_Colorpixel
{
    public partial class Form_Main : Form
    {
        public Form_Main()
        {
            InitializeComponent();
        }
        Bitmap bmpSourceImage;
        Color colorNowPixel;
        int thetop;

        private void btnLoadImage_Click(object sender, EventArgs e)
        {
            bmpSourceImage = new Bitmap("Bolt_0.bmp");
            picShowImage.Image = bmpSourceImage;
        }
        // pictureBox的滑鼠移動事件
        private void picShowImage_MouseMove(object sender, MouseEventArgs e)
        {
            //bmpNowImage=new Bitmap(picShowImage.Image);
            //if (picShowImage.Image != null )
            {
                if (bmpSourceImage != null)
                {
                    //取得pixel的RGB值
                    colorNowPixel = bmpSourceImage.GetPixel(e.X, e.Y);
                    labR_Value.Text = colorNowPixel.R.ToString();
                    labG_Value.Text = colorNowPixel.G.ToString();
                    labB_Value.Text = colorNowPixel.B.ToString();
                    //顯示滑鼠座標
                    label_X.Text = e.X.ToString();
                    label_Y.Text = e.Y.ToString();
                }
            }
        }
    }
}

```

```

    }
}

private void btnClearR_Click(object sender, EventArgs e)
{
    for (int j = 0; j < bmpSourceImage.Height; j++)
    {
        for (int i = 0; i < bmpSourceImage.Width; i++)
        {
            Color pixelcolor = bmpSourceImage.GetPixel(i, j);

            if (pixelcolor.R < 50)
            {
                thetop = j;
                Bitmap imgone = new Bitmap(bmpSourceImage);
                using (Graphics g = Graphics.FromImage(imgone))
                {
                    g.DrawLine(new Pen(Color.Red), i-50, j, i+50, j);
                }

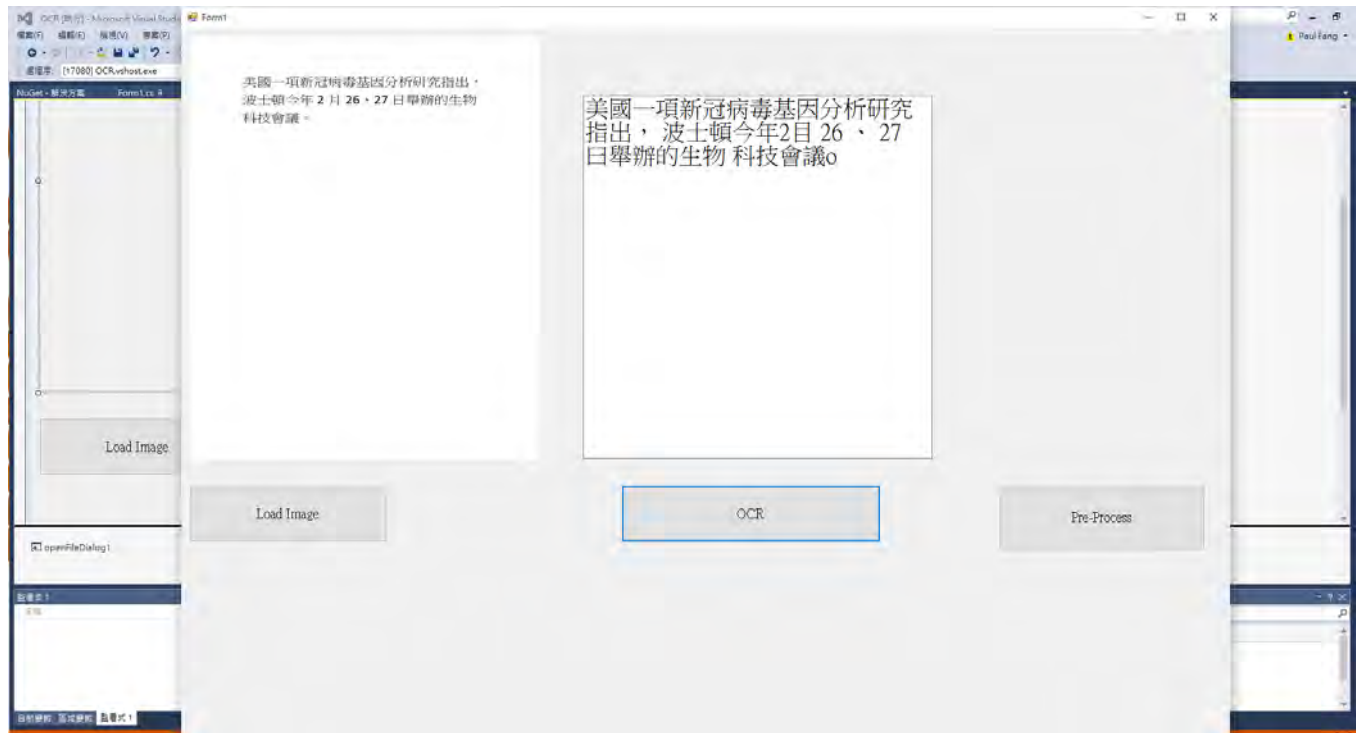
                pictureBox1.Image = imgone;
                j = bmpSourceImage.Height;
                i = bmpSourceImage.Width;
            }
        }
    }
}
}

```

- | |
|---|
| 1. 不利用其他的 Library，找圖案最底部 |
| 2. 不利用其他的 Library，自行劃出 Bounding Box，並給予四個頂點座標 |
| 3. 不利用其他的 Library 計算面積，與質心 Center of Mass |

Lesson 16

Tesseract 與 OCR



程式:iOCR，tessdata

iOCR_ToStudent

Download 中文字庫 https://github.com/tesseract-ocr/tessdata_best

放入 bin/tessdata 資料夾

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;
```

```

using System.Windows.Forms;
using Tesseract;
using AForge;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;

namespace OCR
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        Bitmap srcimg, grayimg, bwimg, maskimg, img_R, img_B, dstimg, img;

        private void button1_Click(object sender, EventArgs e)
        {
            openFileDialog1.Filter = "所有檔案(*.*)|*.*";
            if (openFileDialog1.ShowDialog() == DialogResult.OK)
            {
                srcimg =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
            }

            pictureBox1.Image = srcimg;
            img = srcimg;
        }

        private void button2_Click(object sender, EventArgs e)
        {
            Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
            grayimg = filter.Apply(srcimg);

```

```

        ContrastStretch filter_C = new ContrastStretch();
        filter_C.ApplyInPlace(grayimg);

        pictureBox2.Image = bwimg;
        img = bwimg;

    }

    private void button10_Click(object sender, EventArgs e)
    {
        //img = srcimg;
        TesseractEngine engine = new TesseractEngine(@"./tessdata",
"chi_tra", EngineMode.Default);
        var page1 = engine.Process(img, PageSegMode.Auto);
        textBox9.Text = page1.GetText();

    }

}

}

```



```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using Tesseract;
using Newtonsoft.Json;
using AForge;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;

namespace OCR
{
    public partial class Form1 : Form
    {
        public Form1()
    }

```

```

    {
        InitializeComponent();
    }
    Bitmap srcimg, grayimg, bwimg, img;

private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案 (*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        srcimg =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
    }

    pictureBox1.Image = srcimg;
    img = srcimg;
}

private void button2_Click(object sender, EventArgs e)
{
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    grayimg = filter.Apply(srcimg);

    ContrastStretch filter_C = new ContrastStretch();
    bwimg= filter_C.Apply(grayimg);

    pictureBox1.Image = bwimg;
    img = bwimg;
}

private void button10_Click(object sender, EventArgs e)
{

```

```

        //img = srcimg;
        TesseractEngine engine = new TesseractEngine(@"./tessdata","eng",
EngineMode.Default);
        var page1 = engine.Process(img, PageSegMode.Auto);
        textBox9.Text = page1.GetText();

    }

    private void button3_Click(object sender, EventArgs e)
    {
        //img = srcimg;
        TesseractEngine engine = new TesseractEngine(@"./tessdata", "eng",
EngineMode.Default);
        var page1 = engine.Process(img, PageSegMode.SingleLine);
        textBox9.Text = page1.GetText();
    }

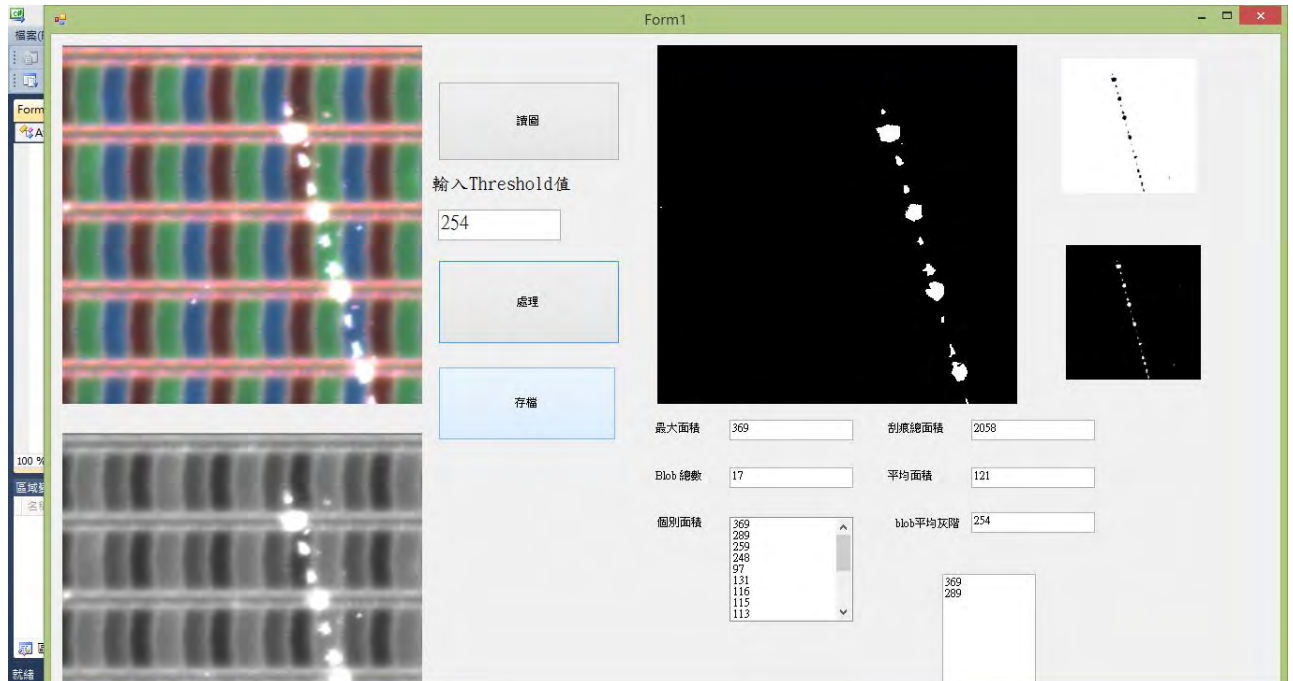
}
}

```

Lesson 17

LCD 玻璃刮痕瑕疵檢測

(LCD 玻璃刮痕瑕疵檢測)



```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using AForge.Imaging;
using AForge.Imaging.Formats;
using AForge.Imaging.Filters;
using AForge.Math;
using AForge.Video;

namespace AforgeBlob_1
{
    public partial class Form1 : Form
    {

```

```

Bitmap imgone;

public Form1()
{
    InitializeComponent();
}

private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案(*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgone =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        pictureBox1.Image = imgone;
    }
}

private void button3_Click(object sender, EventArgs e)
{
    saveFileDialog1.FileName = @"imgone.bmp";
    saveFileDialog1.Filter = "Bitmap file (*.bmp)|*.bmp";
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {
        imgone.Save(saveFileDialog1.FileName); //, ImageFormat.Bmp);
    }
}

private void button2_Click(object sender, EventArgs e)
{
    int largestarea, ii, blobcount, totalarea, avearea;
    //Bitmap image;
    // create grayscale filter (BT709)
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    // apply the filter
    Bitmap grayImage = filter.Apply(imgone);
    pictureBox2.Image = grayImage;
}

```

```

// create filter
int Threshold;
Threshold = int.Parse(textBox3.Text);
Threshold filter2 = new Threshold(Threshold);
// apply the filter
Bitmap bimage=filter2.Apply(grayImage);
pictureBox3.Image = bimage;

// create an instance of blob counter algorithm
BlobCounterBase bc = new BlobCounter();
// set filtering options
bc.FilterBlobs = true;
bc.MinWidth = 5;
bc.MinHeight = 5;
// set ordering options
bc.ObjectsOrder = ObjectsOrder.Size;
// process binary image
bc.ProcessImage(bimage);
Blob[] blobs = bc.GetObjectsInformation();
// extract the biggest blob

largestarea = blobs[0].Area;
textBox1.Text = largestarea.ToString();
blobcount=bc.ObjectsCount;
textBox2.Text = bc.ObjectsCount.ToString();

totalarea = 0;
for (ii = 0; ii < blobcount; ii++)
{
    totalarea = totalarea + blobs[ii].Area;

    listBox1.Items.Add(blobs[ii].Area.ToString());
    textBox8.Text =
blobs[ii].Area.ToString()+Environment.NewLine;

}
textBox8.Text = blobs[0].Area.ToString() + Environment.NewLine +

```

```

blobs[1].Area.ToString();
    textBox4.Text = totalarea.ToString();

    avearea = totalarea / blobcount;
    textBox5.Text = avearea.ToString();
////////////////////////////////////
    Invert filterinv= new Invert();
    Bitmap invimg = filterinv.Apply(bimage);
    pictureBox4.Image = invimg;

    // create filter
    Subtract filtersub = new Subtract(invimg);
    // apply the filter
    Bitmap blobimg = filtersub.Apply(grayImage);
    pictureBox5.Image = blobimg;

    // gather statistics
    ImageStatistics stat = new ImageStatistics(blobimg);
    // get red channel's histogram
    Histogram gray = stat.Gray;
    int blobimgarea = stat.PixelsCountWithoutBlack;
    double blobmean =
gray.Mean*blobimg.Height*blobimg.Width/blobimgarea;
    textBox6.Text = blobmean.ToString();

}

private void label3_Click(object sender, EventArgs e)
{

}

private void textBox4_TextChanged(object sender, EventArgs e)
{

}

```

```

private void pictureBox1_Click(object sender, EventArgs e)
{

}

private void pictureBox3_Click(object sender, EventArgs e)
{

}

private void Form1_Load(object sender, EventArgs e)
{

}

private void textBox2_TextChanged(object sender, EventArgs e)
{

}

private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
{

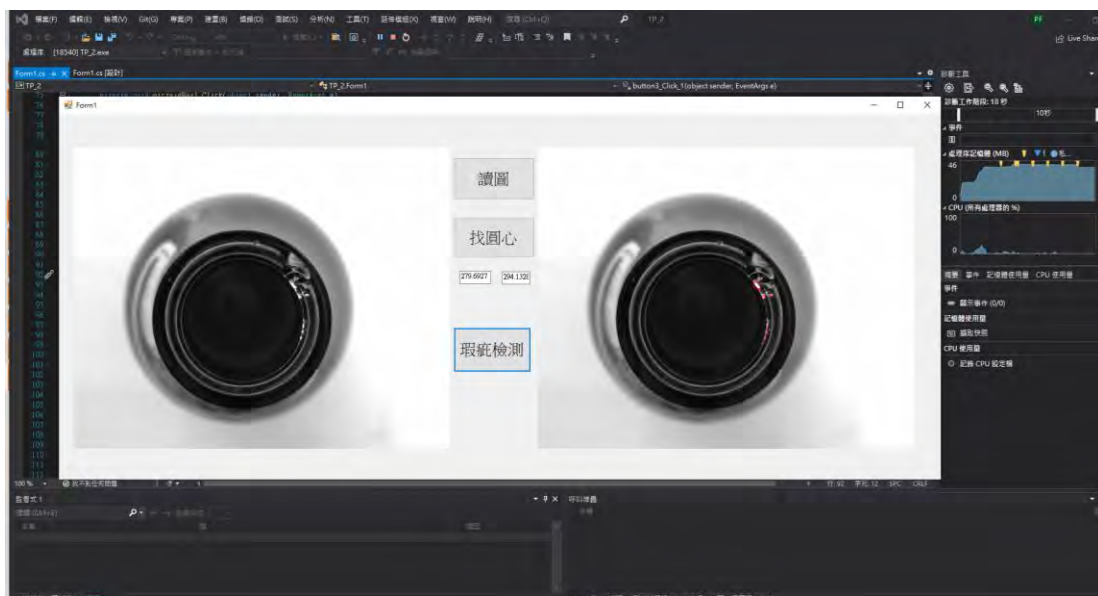
}

private void textBox8_TextChanged(object sender, EventArgs e)
{

}
}
}

```

程式



程式 玻璃瓶口檢測

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
```

```
using System.Windows.Forms;
using AForge.Imaging.Filters;
using AForge.Imaging;
using AForge.Video;
```

```
namespace TP_2
{
    public partial class Form1 : Form
    {

        public Form1()
        {
            InitializeComponent();
        }

        Bitmap srcimg, srcinvert, dstimg;
        float Cx, Cy;
```

```

private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案 (*.*)|*. *";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        //srcimg =
ImageDecoder.DecodeFromFile(openFileDialog1.FileName);
        Bitmap temp = new Bitmap(openFileDialog1.FileName);
        Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
        srcimg = filter.Apply(temp);
        pictureBox1.Image = srcimg;
        dstimg = new Bitmap(srcimg);
    }
}

private void button2_Click(object sender, EventArgs e)
{
    Invert filter = new Invert();
    srcinvert = filter.Apply(srcimg);
    pictureBox2.Image = srcinvert;
    Threshold filter_T = new Threshold(220);
    Bitmap timg = filter_T.Apply(srcinvert);
    pictureBox2.Image = timg;

    BlobCounterBase bc = new BlobCounter();
    //The class is abstract and serves as a base for different blob counting
algorithms.
    // set filtering options
    bc.FilterBlobs = true;
    bc.MinWidth = 5;
    bc.MinHeight = 5;

    // set ordering options

```

```

        bc.ObjectsOrder = ObjectsOrder.Area;

        bc.ProcessImage(timg); // process binary image
        Blob[] blobs = bc.GetObjectsInformation();

        textBox1.Text = blobs[0].CenterOfGravity.X.ToString();// extract the
biggest blob
        textBox2.Text = blobs[0].CenterOfGravity.Y.ToString();
        Cx = blobs[0].CenterOfGravity.X;
        Cy = blobs[0].CenterOfGravity.Y;
    }

    private void pictureBox1_Click(object sender, EventArgs e)
    {

    }

    private void button3_Click_1(object sender, EventArgs e)
    {

    }

    Threshold filter_T = new Threshold(230);
    Bitmap timg = filter_T.Apply(srcimg);
    pictureBox2.Image = timg;

    BlobCounterBase bc = new BlobCounter();
    //The class is abstract and serves as a base for different blob counting
algorithms.
    // set filtering options
    bc.FilterBlobs = true;
    bc.MinWidth = 2;
    bc.MinHeight = 2;
    bc.MaxHeight = 100;
    bc.MaxWidth = 100;
    // set ordering options
    bc.ObjectsOrder = ObjectsOrder.Area;

    bc.ProcessImage(timg); // process binary image

```

```

        Blob[] blobs = bc.GetObjectsInformation();
        for(int i=0; i<bc.ObjectsCount;i++)
        {
            double distance = Math.Sqrt(((blobs[i].CenterOfGravity.X - Cx) *
(blobs[i].CenterOfGravity.X - Cx) + (blobs[i].CenterOfGravity.Y - Cy) *
(blobs[i].CenterOfGravity.Y - Cy)));
            if (distance<150.0)
            {
                using (Graphics g = Graphics.FromImage(dstimg))
                {
                    g.DrawRectangle(new Pen(Color.Red),
blobs[i].Rectangle);
                }
            }
        }

        pictureBox2.Image = dstimg;

    }

    private void button3_Click(object sender, EventArgs e)
    {

    }

}
}

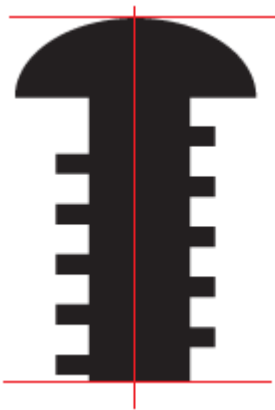
```

習題

1. a. 量測 Bolt_0 的長度

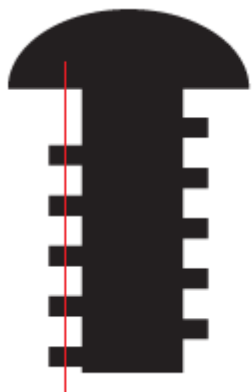


Bolt_0



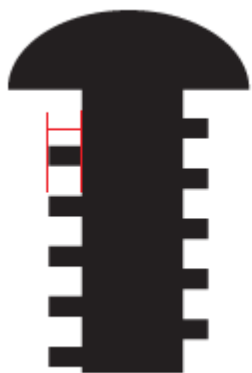
Bolt_10

1 b. 量測 Bolt_0 的左邊螺牙的個數與間距



Bolt_11

1.c. 量測 Bolt_0 的左邊螺牙的深度



Bolt_12

2. 寫一程式，以 Bolt_0 為標準，判讀 Bolt_1~Bolt_8 何者合格，並說明你的判讀方法



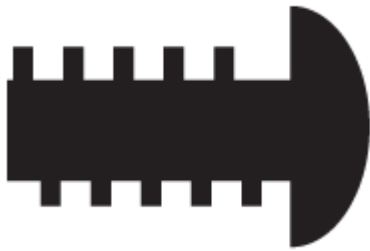
Bolt_1



Bolt_2



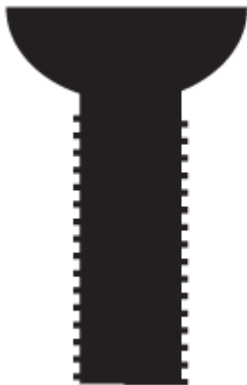
Bolt_3



Bolt_4



Bolt_5



Bolt_6



Bolt_7



Bolt_8

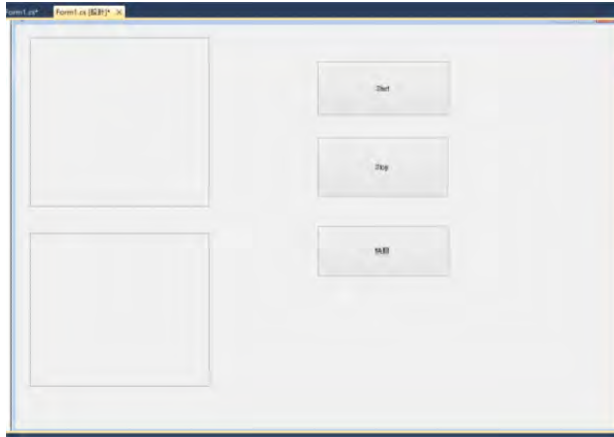
3. 量測 Nut_0 本身的面積、中心之圓面積與直徑



Lesson 18

攝影機的使用攝影機的使用與 Read Video File

最簡單版的攝影機



```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;
```

```
using AForge.Imaging;  
using AForge.Imaging.Filters;  
using AForge.Video;  
using AForge.Video.DirectShow;
```

```
namespace cam_aforge_simple  
{  
    public partial class Form1 : Form  
    {  
  
        private FilterInfoCollection videoDevices;  
        private VideoCaptureDevice videoSource = null;
```

```

        Bitmap image, imageToProcess;

    public Form1()
    {
        InitializeComponent();
    }

    private void video_NewFrame(object sender, NewFrameEventArgs
eventArgs)
    {
        Bitmap img = (Bitmap)eventArgs.Frame.Clone();
        pictureBox1.Image = img;
        image = img;
    }

    private void button1_Click(object sender, EventArgs e)
    {
        videoDevices = new
FilterInfoCollection(FilterCategory.VideoInputDevice);
        videoSource = new
VideoCaptureDevice(videoDevices[0].MonikerString);
        videoSource.NewFrame += new
NewFrameEventHandler(video_NewFrame);
        videoSource.Start();
    }

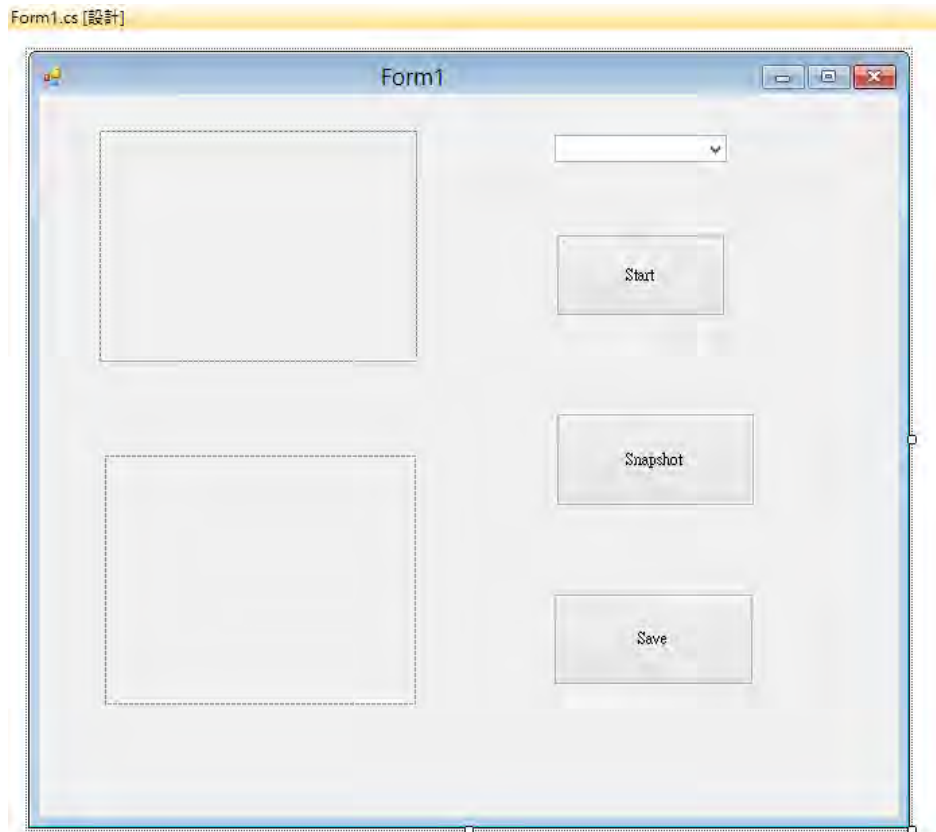
    private void button2_Click(object sender, EventArgs e)
    {
        videoSource.SignalToStop();
    }

    private void button3_Click(object sender, EventArgs e)
    {
        pictureBox2.Image = image;
        imageToProcess = image;
    }
}

```

}

簡單版的攝影機使用



```
using System;  
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;  
using System.IO;  
using System.Drawing.Imaging;  
//AForge.Video dll  
using AForge.Video;  
using AForge.Video.DirectShow;  
using AForge.Imaging;  
using AForge.Imaging.Filters;
```

```
using AForge;
```

```
namespace WebCameraCapture
```

```
{
```

```
    public partial class Form1 : Form
```

```
    {
```

```
        public Form1()
```

```
        {
```

```
            InitializeComponent();
```

```
        }
```

```
        private FilterInfoCollection CaptureDevice; // list of webcam
```

```
        private VideoCaptureDevice FinalFrame;
```

```
        private void Form1_Load(object sender, EventArgs e)
```

```
        {
```

```
            CaptureDevice = new
```

```
FilterInfoCollection(FilterCategory.VideoInputDevice); //constructor
```

```
            foreach (FilterInfo Device in CaptureDevice)
```

```
            {
```

```
                comboBox1.Items.Add(Device.Name);
```

```
            }
```

```
            comboBox1.SelectedIndex = 0; // default
```

```
            FinalFrame = new VideoCaptureDevice();
```

```
        }
```

```
        void FinalFrame_NewFrame(object sender, NewFrameEventArgs  
eventArgs) // must be void so that it can be accessed everywhere.
```

```
        // New Frame Event Args is an constructor of a class
```

```
        {
```

```
            pictureBox1.Image = (Bitmap)eventArgs.Frame.Clone(); // clone the  
bitmap
```

```
        }
```

```
        private void From1_Closing(object sender, EventArgs e)
```

```

    {
        if (FinalFrame.IsRunning == true) FinalFrame.Stop();
    }

    private void button1_Click_1(object sender, System.EventArgs e)
    {
        FinalFrame = new
VideoCaptureDevice(CaptureDevice[comboBox1.SelectedIndex].MonikerString);//
specified web cam and its filter moniker string
        FinalFrame.NewFrame += new
NewFrameEventHandler(FinalFrame_NewFrame);// click button event is fired,
        FinalFrame.Start();
    }

    private void button2_Click(object sender, System.EventArgs e)
    {
        pictureBox2.Image = (Bitmap)pictureBox1.Image.Clone();

    }

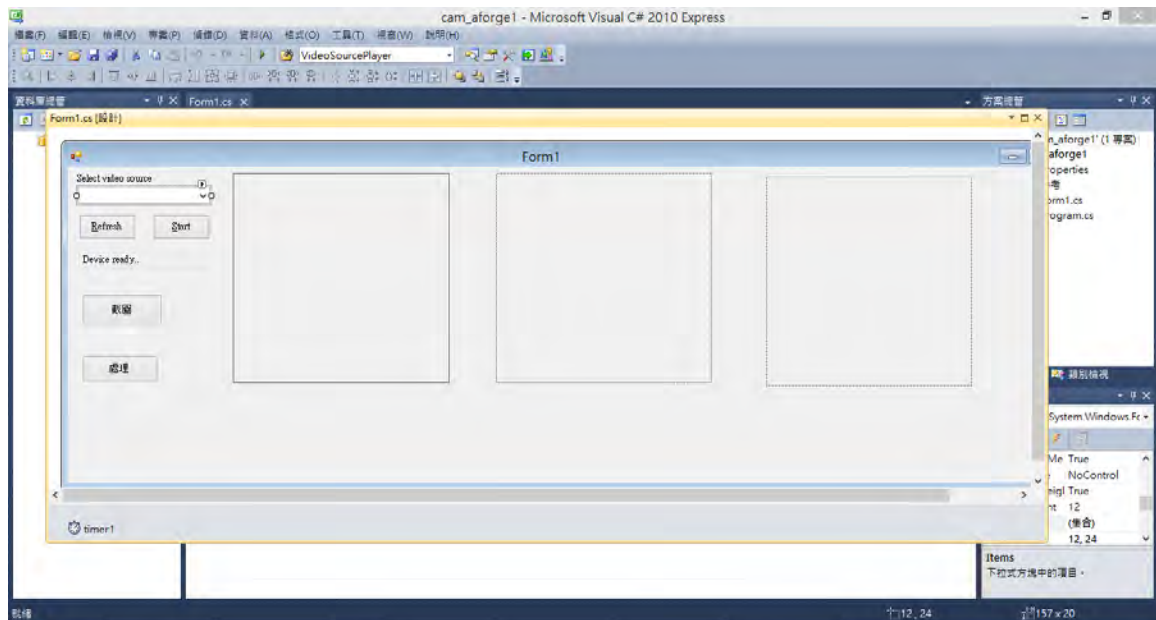
    private void button3_Click(object sender, System.EventArgs e)
    {
        if (pictureBox2.Image != null)
        {
            Bitmap varBmp = new Bitmap(pictureBox2.Image);
            Bitmap newBitmap = new Bitmap(varBmp);

            varBmp.Save(@"D:\a.png", ImageFormat.Png);
            varBmp.Dispose();
            varBmp = null;
        }
        else
        { MessageBox.Show("null exception"); }
    }
}

```

}

較複雜版的攝影機使用



```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Text;  
using System.Windows.Forms;
```

```
using AForge.Imaging;  
using AForge.Imaging.Filters;  
using AForge.Video;  
using AForge.Video.DirectShow;
```

//本例題 OK

```
namespace cam_aforge1  
{  
    public partial class Form1 : Form  
    {  
        private bool DeviceExist = false;  
        private FilterInfoCollection videoDevices;
```

```

private VideoCaptureDevice videoSource = null;
Bitmap image, imageToProcess;

public Form1()
{
    InitializeComponent();
}

private void getCamList()
{
    try
    {
        videoDevices = new
FilterInfoCollection(FilterCategory.VideoInputDevice);
        comboBox1.Items.Clear();
        if (videoDevices.Count == 0)
            throw new ApplicationException();

        DeviceExist = true;
        foreach (FilterInfo device in videoDevices)
        {
            comboBox1.Items.Add(device.Name);
        }
        comboBox1.SelectedIndex = 0; //make default to first cam
    }
    catch (ApplicationException)
    {
        DeviceExist = false;
        comboBox1.Items.Add("No capture device on your system");
    }
}

private void rfsh_Click(object sender, EventArgs e)
{
    getCamList();
}

private void start_Click(object sender, EventArgs e)

```

```

    {
        if (start.Text == "&Start")
        {
            if (DeviceExist)
            {
                videoSource = new
VideoCaptureDevice(videoDevices[comboBox1.SelectedIndex].MonikerString);
                videoSource.NewFrame += new
NewFrameEventHandler(video_NewFrame);
                CloseVideoSource();
                videoSource.DesiredFrameSize = new Size(160, 120);
                //videoSource.DesiredFrameRate = 10;
                videoSource.Start();
                label2.Text = "Device running...";
                start.Text = "&Stop";
                timer1.Enabled = true;
            }
            else
            {
                label2.Text = "Error: No Device selected.";
            }
        }
        else
        {
            if (videoSource.IsRunning)
            {
                timer1.Enabled = false;
                CloseVideoSource();
                label2.Text = "Device stopped.";
                start.Text = "&Start";
            }
        }
    }

private void video_NewFrame(object sender, NewFrameEventArgs
eventArgs)
{
    Bitmap img = (Bitmap)eventArgs.Frame.Clone();

```

```

        // create filter
        //Edges filter = new Edges();
        // apply the filter
        //filter.ApplyInPlace(img);
        pictureBox1.Image = img;
        image = img;
    }

    private void button1_Click(object sender, EventArgs e)
    {
        pictureBox2.Image = image;
        imageToProcess = image;
    }

    private void CloseVideoSource()
    {
        if (!(videoSource == null))
            if (videoSource.IsRunning)
            {
                videoSource.SignalToStop();
                videoSource = null;
            }
    }

    private void timer1_Tick(object sender, EventArgs e)
    {
        label2.Text = "Device running... " +
        videoSource.FramesReceived.ToString() + " FPS";
    }

    private void Form1_FormClosed(object sender, FormClosedEventArgs e)
    {
        CloseVideoSource();
    }

    private void button2_Click(object sender, EventArgs e)

```

```
{  
  
    // create grayscale filter (BT709)  
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);  
    // apply the filter  
    Bitmap grayImage = filter.Apply(imageToProcess);  
    pictureBox3.Image = grayImage;  
}  
  
}  
}
```

Read Video File (Avi)
(VideoFile_Avi_aforge_最簡版)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

using AForge.Imaging;
using AForge.Imaging.Filters;
using AForge.Video;
using AForge.Video.DirectShow;

namespace cam_aforge_simple
{
    public partial class Form1 : Form
    {
        // private FilterInfoCollection videoDevices;
        // private VideoCaptureDevice videoSource = null;
        FileVideoSource videoSource;
        Bitmap image, imageToProcess;

        public Form1()
        {
            InitializeComponent();
        }

        private void video_NewFrame(object sender, NewFrameEventArgs
eventArgs)
```

```

    {
        // get new frame
        image = (Bitmap) eventArgs.Frame.Clone();
        // process the frame
        pictureBox1.Image = image;
    }

private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案 (*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        videoSource = new FileVideoSource(openFileDialog1.FileName);
    }
    // set NewFrame event handler
    videoSource.NewFrame += new
NewFrameEventHandler(video_NewFrame);
    // start the video source
    videoSource.Start();
}

private void button2_Click(object sender, EventArgs e)
{
    videoSource.SignalToStop();
}

private void button3_Click(object sender, EventArgs e)
{
    pictureBox2.Image = image;
    imageToProcess = image;
}
}
}

```

Read Video File (MPG4)
(VideoFile_MPG4_aforge_最簡版)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

using AForge.Imaging;
using AForge.Imaging.Filters;
//using AForge.Video;
using Accord.Video;
using Accord.Video.FFMPEG;

//using AForge.Video.DirectShow;

namespace cam_aforge_simple
{
    public partial class Form1 : Form
    {
        // private FilterInfoCollection videoDevices;
        // private VideoCaptureDevice videoSource = null;
        VideoFileSource videoSource;
        Bitmap image, imageToProcess;

        public Form1()
        {
            InitializeComponent();
        }

        private void video_NewFrame(object sender, NewFrameEventArgs
eventArgs)
        {

```

```

        // get new frame
        image = (Bitmap) eventArgs.Frame.Clone();
        // process the frame
        pictureBox1.Image = image;
    }

private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "所有檔案 (*.*)|*.*";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        videoSource = new VideoFileSource(openFileDialog1.FileName);
    }
    // set NewFrame event handler
    videoSource.NewFrame += new
NewFrameEventHandler(video_NewFrame);
    // start the video source
    videoSource.Start();
}

private void button2_Click(object sender, EventArgs e)
{
    videoSource.SignalToStop();
}

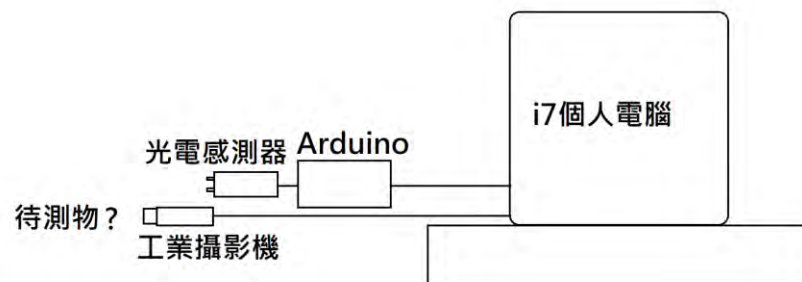
private void button3_Click(object sender, EventArgs e)
{
    pictureBox2.Image = image;
    imageToProcess = image;
}
}
}

```

附錄(一)
期末專題

cam_c#_Arduino_OK

1. 輸入密碼 (使用 multiform)
若密碼是 123，OK 繼續執行
若不是 123，離開程式
2. 利用光電感測器，擷取影像
3. 量測兩鐵片之 a. 面積 b.半徑 c.孔之面積 d.半徑



Arduino 光電感測器與攝影機

Arduino 程式

```
void setup()
{
    Serial.begin(9600);    // 開啟 Serial port, 通訊速率為 9600 bps
    pinMode(4, INPUT);

}
void loop() {
    // 檢查是否有資料可供讀取
    int val = digitalRead(4);
    Serial.println(val);
    delay(500);
}
```

C# 程式

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using System.IO.Ports;

using AForge.Video;
using AForge.Video.DirectShow;

namespace WindowsFormsApplication1
{
    public partial class Form1 : Form
    {

```

```

        FilterInfoCollection videoDevice;
        VideoCaptureDevice videoSource = null;
        int scup1;
        string scup;
        Bitmap img,image;
    public Form1()
    {
        InitializeComponent();
        serialPort1.DataReceived += new
SerialDataReceivedEventHandler(serialPort1_DataReceived);
    }
    private void Form1_Load(object sender, EventArgs e)
    {
        serialPort1.BaudRate = 9600;
        serialPort1.Parity = Parity.None;
        serialPort1.DataBits = 8;
        serialPort1.StopBits = StopBits.One;

        serialPort1.Open();
        videoDevice = new
FilterInfoCollection(FilterCategory.VideoInputDevice);
        videoSource = new
VideoCaptureDevice(videoDevice[0].MonikerString);
        videoSource.NewFrame += new
NewFrameEventHandler(video_NewFrame);

        videoSource.Start();

    }
    private void serialPort1_DataReceived(object sender,
SerialDataReceivedEventArgs e)
    {
        this.Invoke(new EventHandler(value));

    }
    private void value(object sender, EventArgs e)

```

```

    {

        scup = serialPort1.ReadLine();
        textBox1.Text = scup;

        scup1 = Convert.ToInt16(scup);

        if (scup1 == 0) //偵測感光元件
        {
            button1.PerformClick();
        }

    }

    private void button1_Click(object sender, EventArgs e)
    {
        pictureBox2.Image = image;
        //後續的處理

    }

    private void video_NewFrame(object sender, NewFrameEventArgs
eventArgs)
    {
        img = (Bitmap)eventArgs.Frame.Clone();
        pictureBox1.Image = img;
        image = img;

    }
}

```

C# Arduino_Camera

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using System.IO.Ports;

using AForge.Video;
using AForge.Video.DirectShow;

namespace WindowsFormsApplication1
{
    public partial class Form1 : Form
    {
        FilterInfoCollection videoDevice;
        VideoCaptureDevice videoSource = null;
        int scup1;
        string scup;
        Bitmap img,image;
        public Form1()
        {
            InitializeComponent();
            serialPort1.DataReceived += new
SerialDataReceivedEventHandler(serialPort1_DataReceived);
        }
        private void Form1_Load(object sender, EventArgs e)
        {
            serialPort1.BaudRate = 9600;
            serialPort1.Parity = Parity.None;
            serialPort1.DataBits = 8;
```

```

        serialPort1.StopBits = StopBits.One;

        serialPort1.Open();
        videoDevice = new
FilterInfoCollection(FilterCategory.VideoInputDevice);
        videoSource = new
VideoCaptureDevice(videoDevice[0].MonikerString);
        videoSource.NewFrame += new
NewFrameEventHandler(video_NewFrame);

        videoSource.Start();

    }
    private void serialPort1_DataReceived(object sender,
SerialDataReceivedEventArgs e)
    {
        this.Invoke(new EventHandler(value));
    }
    private void value(object sender, EventArgs e)
    {

        scup = serialPort1.ReadLine();

        scup1 = Convert.ToInt16(scup);

        if (scup1 == 0) //偵測感光元件
        {
            button1.PerformClick();
        }

    }

    private void button1_Click(object sender, EventArgs e)
    {

```

```

        pictureBox2.Image = image;
    }

    private void video_NewFrame(object sender, NewFrameEventArgs
eventArgs)
    {
        img = (Bitmap)eventArgs.Frame.Clone();
        pictureBox1.Image = img;
        image = img;
    }
}
}

```

Solution 2

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using AForge.Imaging;
using AForge.Imaging.Filters;
using AForge.Video;
using AForge.Video.DirectShow;
using System.IO.Ports;

namespace WindowsFormsApplication1
{
    public partial class Form1 : Form
    {
        FilterInfoCollection videoDevice;
    }
}

```

```

VideoCaptureDevice videoSource = null;

Bitmap image, imageToProcess;
int test = 1;
string dis, text2;

int scup1;
string scup;
Bitmap img,img2;

public Form1()
{
    InitializeComponent();
    serialPort1.DataReceived += new
SerialDataReceivedEventHandler(serialPort1_DataReceived);
}

private void serialPort1_DataReceived(object sender,
System.IO.Ports.SerialDataReceivedEventArgs e)
{
    this.Invoke(new EventHandler(value));
}

private void video_NewFrame(object sender, NewFrameEventArgs
eventArgs)
{
    Bitmap img = (Bitmap)eventArgs.Frame.Clone();
    pictureBox1.Image = img;
    image = img;
}

private void button2_Click(object sender, EventArgs e)
{
    videoSource.SignalToStop();
}

```

```

private void button3_Click(object sender, EventArgs e)
{

    imageToProcess = img2;

    // create grayscale filter (BT709)
    Grayscale filter = new Grayscale(0.2125, 0.7154, 0.0721);
    imageToProcess = filter.Apply(imageToProcess);

    OtsuThreshold filter_2 = new OtsuThreshold();
    imageToProcess = filter_2.Apply(imageToProcess);

    pictureBox2.Image = imageToProcess;

}

private void pictureBox1_Click(object sender, EventArgs e)
{

}

private void Form1_Load(object sender, EventArgs e)
{
    serialPort1.BaudRate = 9600;
    serialPort1.Parity = Parity.None;
    serialPort1.DataBits = 8;
    serialPort1.StopBits = StopBits.One;

    serialPort1.Open();
    videoDevice = new
FilterInfoCollection(FilterCategory.VideoInputDevice);
    videoSource = new
VideoCaptureDevice(videoDevice[0].MonikerString);
    videoSource.NewFrame += new
NewFrameEventHandler(video_NewFrame);
}

```

```

        videoSource.Start();

    }

    private void value(object sender, EventArgs e)
    {

        scup = serialPort1.ReadLine();

        scup1 = Convert.ToInt16(scup);

        if (scup1 >100) //偵測感光元件
        {
            button1.PerformClick();
        }

    }

    private void button1_Click(object sender, EventArgs e)
    {

        pictureBox2.Image = image;
        image = img2;
    }
}

```

Arduino

```

unsigned long time_reset,time_diff;
int IR1_Objects = 4; // 紅外線接腳 #4
int IR1 = HIGH; // 紅外線初始值為 HIGH，沒東西擋住為 HIGH

void setup(){
    Serial.begin(9600);
    pinMode(4,INPUT); // 定義紅外線為輸入端
}

```

```

void loop(){
    IR1=digitalRead(4); // 定義 IR_Objects=IR 並為讀取紅外線狀態

    if(IR1 == LOW) // 有東西擋住為 LOW
    {
        time_reset=millis(); //紀錄開始時間
        // Serial.println("START ");
        do
        {
            delay(1); //每隔 1ms 檢查 IR1 一次
            IR1=digitalRead(4);
        } while(IR1==LOW);
        time_diff= millis()-time_reset;//時間差現在時間開始時間
        // Serial.print("Time Difference : ");
        Serial.println(time_diff);
    }

}

```