F2EBI: DEVELOPER GUIDE

Ricardo Tapiador Morales

Robotics and technology of computers/ University of Seville

Contenido

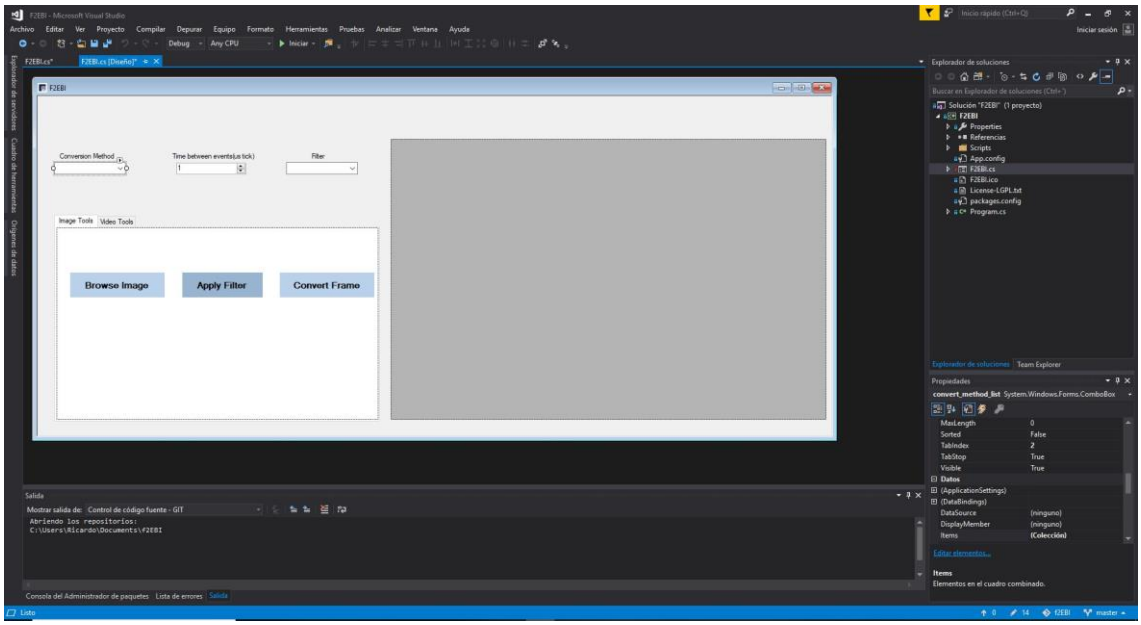
Introduction

F2EBI need Visual Studio community and .Net Framework 4.6.1.

The software has been developed using Windows Form application where each component produces an event.

In this Document it is explained how to develop a custom filter to convert from Frames to AEDAT format.

When repository is downloaded, you should click over F2EBI.sln file and Visual Studio will open the project. With the project opened we are able to explore all files, if you want to see the windows GUI you must click in F2EBI.cs in order to see the form as it shown in next picture.

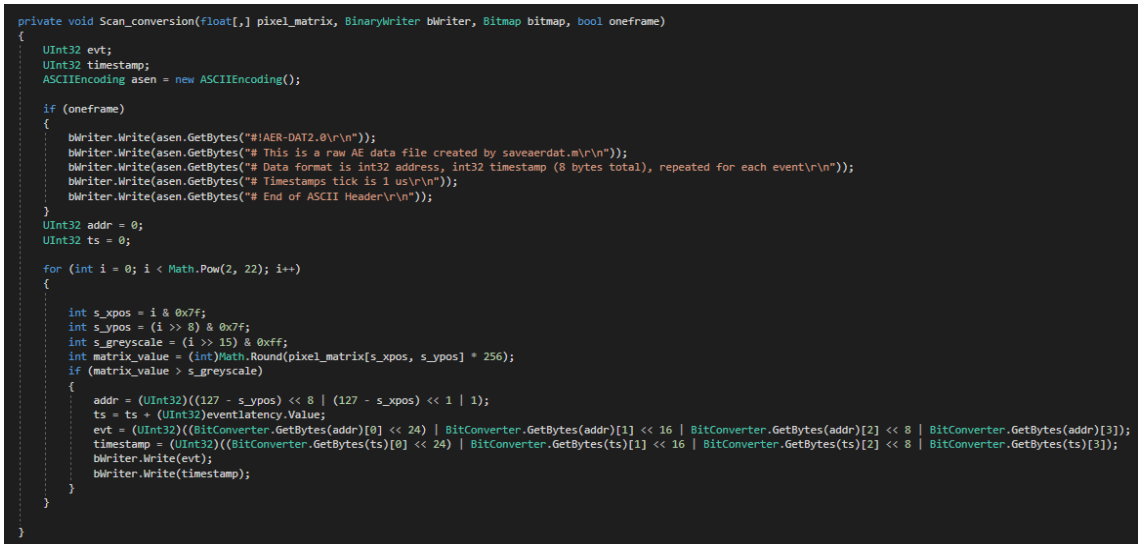


Windows from is composed of different components. Unless your algorithms need another control it is not necessary to add more.

Creating and adding a new filter

In Conversion Methods List, we write which algorithms we will use in order to convert the image. Using the GUI as it shown before you can edit the list to add new methods.

Having a look to the code of F2EBI.cs, you can see all code of the Windows form components and conversion algorithms as independent function as it seen in next image for Scan method.



```
private void Scan_conversion(float[,] pixel_matrix, BinaryWriter bWriter, Bitmap bitmap, bool oneframe)
{
    UInt32 evt;
    UInt32 timestamp;
    ASCIIEncoding asen = new ASCIIEncoding();

    if (oneframe)
    {
        bWriter.Write(asen.GetBytes("#!AER-DAT2.0\r\n"));
        bWriter.Write(asen.GetBytes("# This is a raw AE data file created by saveaerdat.m\r\n"));
        bWriter.Write(asen.GetBytes("# Data format is int32 address, int32 timestamp (8 bytes total), repeated for each event\r\n"));
        bWriter.Write(asen.GetBytes("# Timestamps tick is 1 us\r\n"));
        bWriter.Write(asen.GetBytes("# End of ASCII Header\r\n"));
    }
    UInt32 addr = 0;
    UInt32 ts = 0;

    for (int i = 0; i < Math.Pow(2, 22); i++)
    {

        int s_xpos = i & 0x7f;
        int s_ypos = (i >> 8) & 0x7f;
        int s_greyscale = (i >> 15) & 0xff;
        int matrix_value = (int)Math.Round(pixel_matrix[s_xpos, s_ypos] * 256);
        if (matrix_value > s_greyscale)
        {
            addr = (UInt32)((127 - s_ypos) << 8 | (127 - s_xpos) << 1 | 1);
            ts = ts + (UInt32)eventlatency.Value;
            evt = (UInt32)((BitConverter.GetBytes(addr)[0] << 24) | BitConverter.GetBytes(addr)[1] << 16 | BitConverter.GetBytes(addr)[2] << 8 | BitConverter.GetBytes(addr)[3]);
            timestamp = (UInt32)((BitConverter.GetBytes(ts)[0] << 24) | BitConverter.GetBytes(ts)[1] << 16 | BitConverter.GetBytes(ts)[2] << 8 | BitConverter.GetBytes(ts)[3]);
            bWriter.Write(evt);
            bWriter.Write(timestamp);
        }
    }

}
```

All conversion functions have the same header:

```
private void Scan_conversion(float[,] pixel_matrix, BinaryWriter bWriter, bool oneframe)
```

pixel matrix: float array that has the brightness of all pixel of the loaded image.

bWriter: Binary writer to write the output AEDAT file.

Oneframe: Boolean variable to indicate if it is one frame or multiple for videos.

When new algorithm has been developed we have to add our filter.

Inside the code there is the next method:

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

This method corresponds to convert button in the GUI form, and if a conversion algorithm is selected and image loaded it will apply the algorithm converting the image.

To add a new filter, add a new if condition where convert method list content is compared with the name of the filter added, inside the condition add your method as it shown in next code.

```
if (convert_method_list.Text == "SCAN")
                {
                    Scan_conversion(pixel_matrix, bWriter, true);
                }
                else if (convert_method_list.Text == "Random")
                {
                    Random_Method(pixel_matrix, bWriter, true);

                }
                else if (convert_method_list.Text == "Bitwise")
                {
                    Bitwise_method(pixel_matrix, bWriter, true);
                 }
```

Important: you should add a new entry in the conversion method list, and variables of functions are given it is not necessary new ones unless your conversion method needs.

Adding image filters

It follows the same mechanism as conversion methods we should add a new entry in filters list and add a new condition in Apply_filter_click, that corresponds to apply filter button event.

```
private void ApplyFilter_Click(object sender, EventArgs e)
        {
            if (current_image == null)
             {
                    MessageBox.Show("Please Select an Image");

             }
            else
             {
                    Image<Gray, float> Convolved_Image = new Image<Gray,
                    float>(current_image);


             if (filter_list.Text == "Sobel Vertical")
                {
                    Convolved_Image = ImageFrame.Convert<Gray, Byte>().Sobel(0,
                    1, 3).AbsDiff(new Gray(0.0));

                }

                if (filter_list.Text == "Sobel Horizontal")
                {
                    Convolved_Image = ImageFrame.Convert<Gray, Byte>().Sobel(1,
                    0, 3).AbsDiff(new Gray(0.0));
                }

                if (filter_list.Text == "Laplacian")
                {
                    Convolved_Image = ImageFrame.Convert<Gray,
                    Byte>().Laplace(3).AbsDiff(new Gray(5.0));
                }

                if (filter_list.Text == "Gaussian")
                {
                    Convolved_Image = ImageFrame.Convert<Gray,
                    float>().SmoothGaussian(3);
                }

                current_image = Convolved_Image.Bitmap;

                Get_pixel_bright(current_image, pixel_matrix);

                pictureBox1.Image = current_image;
            }
        }
```

F2EBI use EMGU, an OpenCv wrapper for C#, so you can use this library for your custom image filters.

Video Conversion

To add your algorithm in Video converter just add the same condition added in ConvertFrame_Click in BrowseVideo_Click as it shown in next code:

```
switch (method)
     {
            case "SCAN
            Scan_conversion(pixel_matrix_frame, bWriter, oneframe);
            break;

            case "Random":
            Random_Method(pixel_matrix_frame, bWriter, oneframe);
            break;

            case "Bitwise":
            Bitwise_method(pixel_matrix_frame, bWriter, oneframe);
            break;
     }
```

Contact

I hope this documentation have helped you to add new features. If you still have questions or doubts does not hesitate to contact with me at ricardo@atc.us.es.