

CAS 741 DEVELOPMENT OF SCIENTIFIC COMPUTING
SOFTWARE
IMPLEMENTATION REPORT

**IMAGE FEATURE DETECTION
SYSTEM**

April 15,2024

Gaofeng Zhou

Department of Computing and Software
McMaster University

Contents

1	Introduction	2
2	Configuration of OpenCV library and Visual Studio	3
2.1	OpenCV setup	3
2.2	Visual Studio Configuration	3
2.3	OpenCV Sample Test	3
3	GUI Design	4
3.1	Layout Design	4
3.2	GUI Components Connection	4
4	Function development	6
5	Test	7
5.1	JPEG Image Test	7
5.2	PNG Image Test	10
5.3	TIFF Image Test	12
5.4	BMP Image Test	14
6	Distribution	16

1 Introduction

Following the instructions of CAS741, this IFDS system has been successfully realized. Different from the initial version. The newest version was developed with MFC of Visual Studio using C++.

Through the OpenCV library, we succeeded in integrating GUI and feature detection algorithms. Finally, we generated the software with .msi format which can be distributed on Windows OS platform.

With this IFDS, we can do image import and grayscale image transformation, corner points detection, edge points detection, BRISK points detection and feature points export.

This IFDS work can be used for people to make typical image features visually sensible, and will facilitate further image processing. Here in this report, I would like to record the implementation of the development of IFDS.

2 Configuration of OpenCV library and Visual Studio

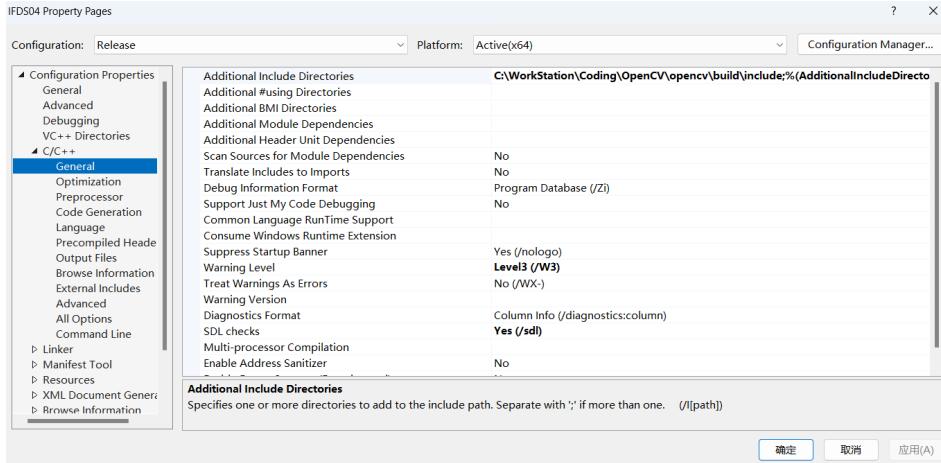


Figure 1: OpenCV Configuration

2.1 OpenCV setup

1. Firstly, we need to setup the environment for OpenCV library. If you haven't use it before, you may need to download it from the website <https://opencv.org/releases/>.
2. Check the include and x64 directory under the build directory to make sure the header files and the libraries existed there.

2.2 Visual Studio Configuration

1. Install Visual Studio as the IDE for this project.
2. Create a MFC application in Visual Studio, with the Dialog mode which can be used to generate a GUI later.
3. open Project–Preferences–Configuration, in the C++ section, add the include directory to the corresponding locale.
4. In the link section, add opencv_world490.lib and opencv_world490d.lib to the release and debug respectively just as Fig 1 showed as above.

2.3 OpenCV Sample Test

To make sure our configuration works well, we need to do a sample test. Here we can use the displayImage sample to do this test, if this sample worked well, then it means the configuration has been done successfully.

3 GUI Design

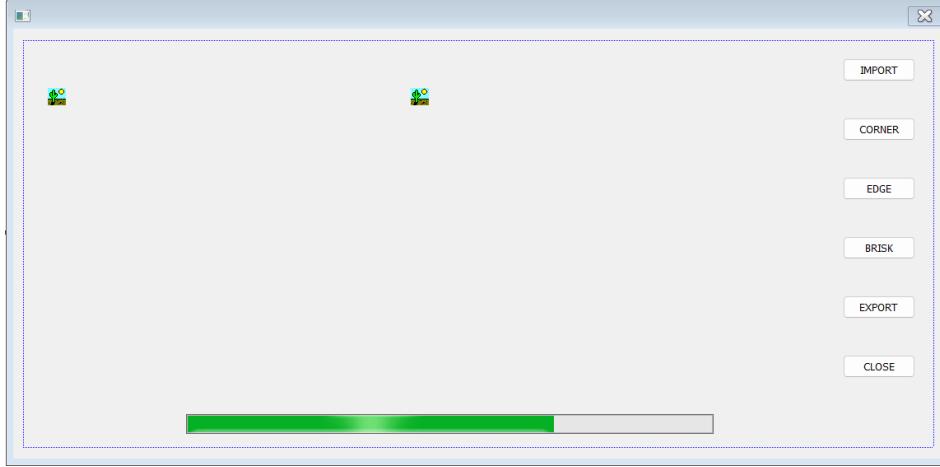


Figure 2: GUI for IFDS through MFC

3.1 Layout Design

- Button Design: Here we set 6 buttons as IMPORT, CORNER, EDGE, BRISK, EXPORT, CLOSE in the right side of the GUI. Meanwhile, we need to set their ID in the MFC framework. For example, we set IMPORT button as IDC_STATIC_BUTTON_IMPORT. and so on so forth.
- Display Windows Design: we set two windows in the GUI to display the original image and grayscale image with feature points. the left one is for original RGB image, and the right one is for grayscale image and feature points. Here, we need to set the image format to bitmap for display.
- Progress Control Bar Design: this progress bar can be used to monitoring the progress of each step.

3.2 GUI Components Connection

- We should define these actions in the header file.
 - void OnClickedButtonImport();
 - void OnClickedButtonCorner();
 - void OnClickedButtonEdge();
 - void OnClickedButtonBrisk();
 - void OnClickedButtonExport();
 - void OnClickedButtonClose();
 - void UpdateProgressBar(int value)

```
ON_BN_CLICKED(IDC_BUTTON_IMPORT, &CIFDS04Dlg::OnClickedButtonImport)
ON_BN_CLICKED(IDC_BUTTON_CORNER, &CIFDS04Dlg::OnClickedButtonCorner)
ON_BN_CLICKED(IDC_BUTTON_EDGE, &CIFDS04Dlg::OnClickedButtonEdge)
ON_BN_CLICKED(IDC_BUTTON_BRISK, &CIFDS04Dlg::OnClickedButtonBrisk)
ON_BN_CLICKED(IDC_BUTTON_EXPORT, &CIFDS04Dlg::OnClickedButtonExport)
ON_BN_CLICKED(IDC_BUTTON_CLOSE, &CIFDS04Dlg::OnClickedButtonClose)
```

Figure 3: Mapping between GUI and actions

- We should map these GUI components with actions and functions.

4 Function development

After the GUI design and Mapping between GUI and actions, we should development those functions using OpenCV.

I have uploaded the source code in GitHub <https://github.com/Zhou4truth/imageFeatureDetection/blob/main/main.cpp>. In this .cpp file, those feature detection functions have been developed.

With the import button, one RGB image with format as JPEG, PNG, BMP, TIFF will be imported into this system and be displayed in the left window of the GUI. Meanwhile, a corresponding grayscale image will be displayed in the right side window of the GUI. With the Corner button, the corner points will be displayed in the right side window of the GUI.

With the Edge button, the edge points will be displayed in the right side window of the GUI.

With the Export button, the feature points image will be saved to the device.

With the Close button, the IFDS system will be closed.

5 Test

5.1 JPEG Image Test

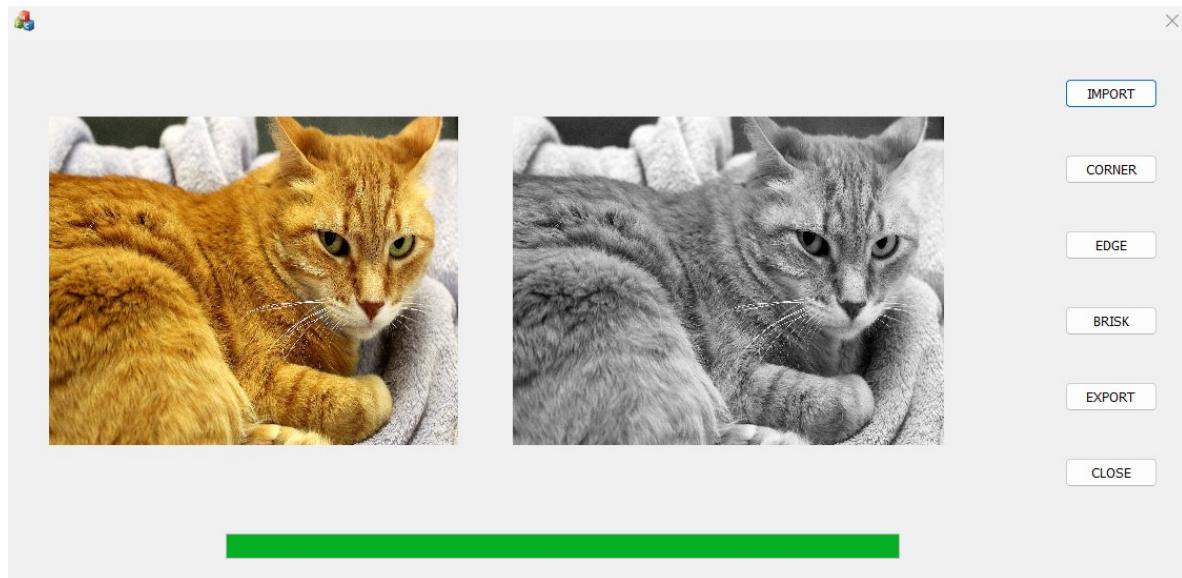


Figure 4: IMPORT

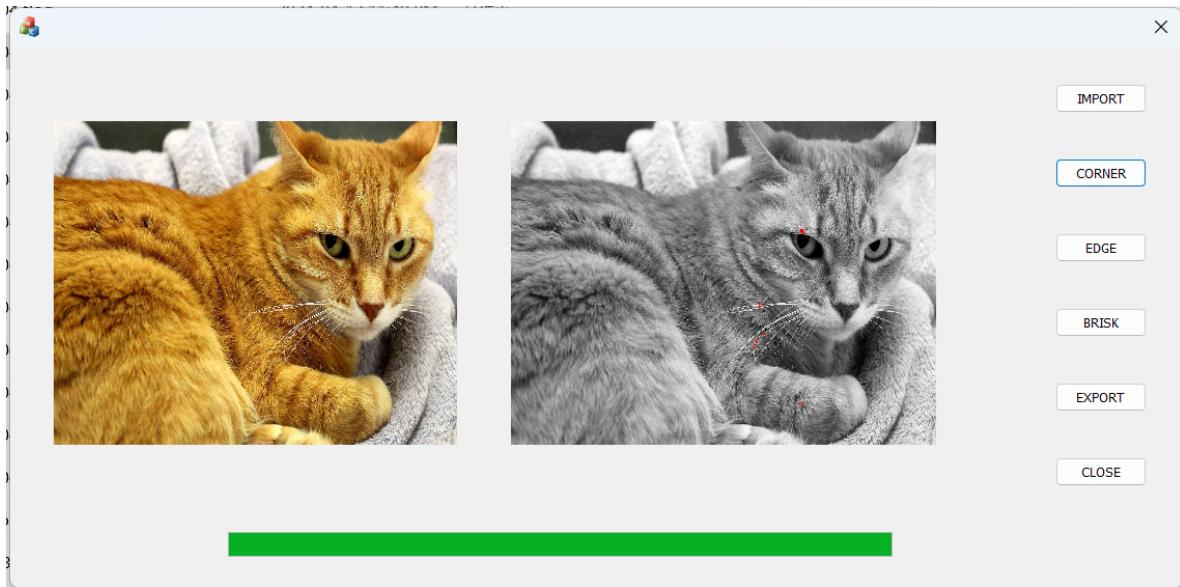


Figure 5: CORNER



Figure 6: EDGE

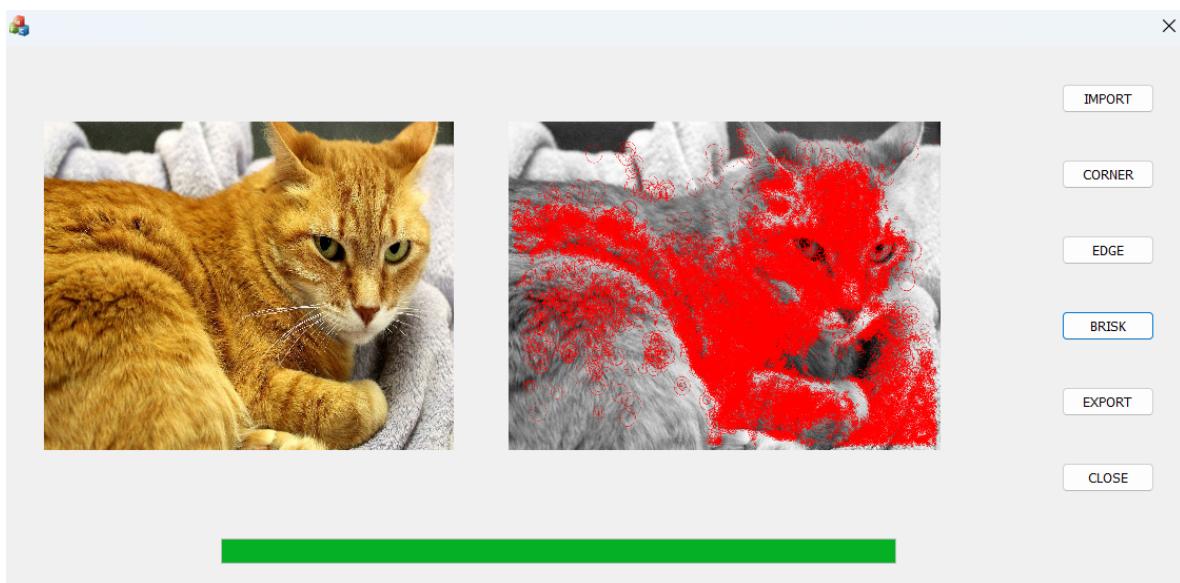


Figure 7: BRISK

5.2 PNG Image Test

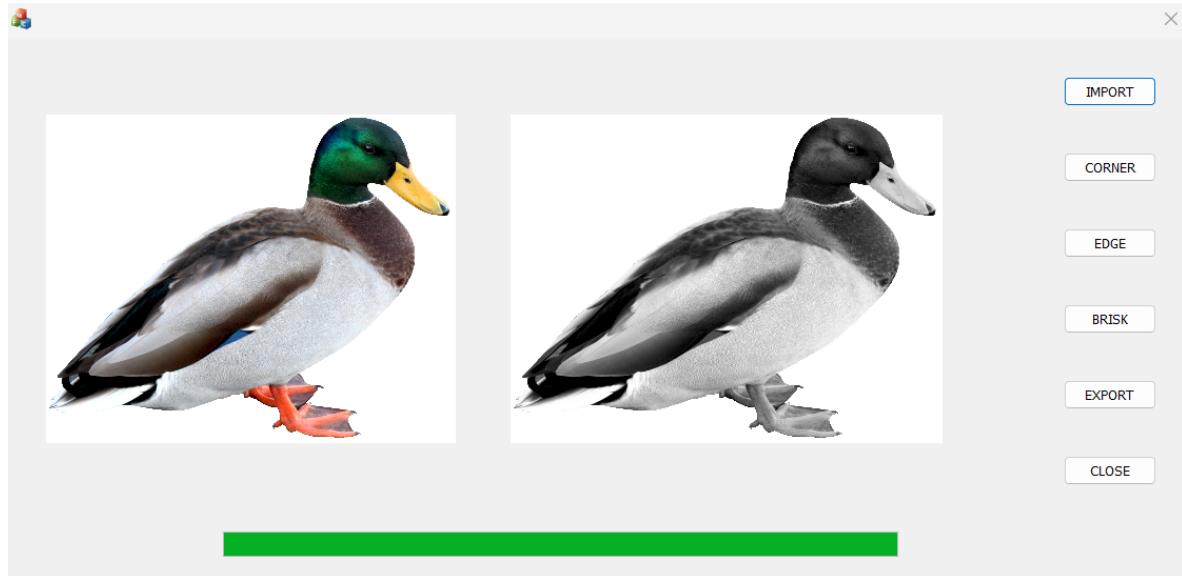


Figure 8: IMPORT

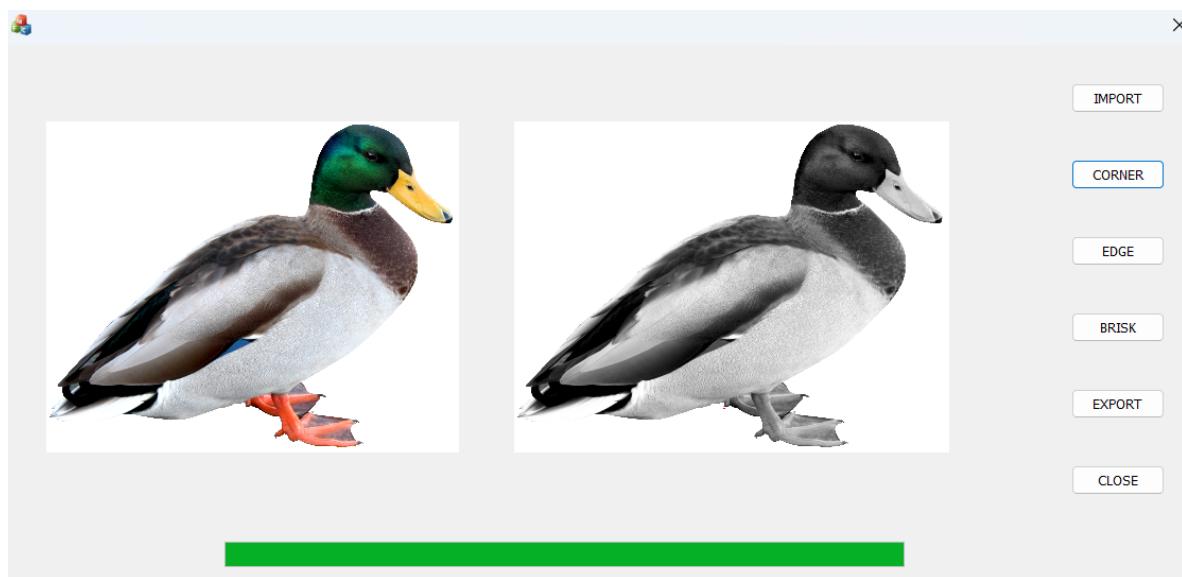


Figure 9: CORNER

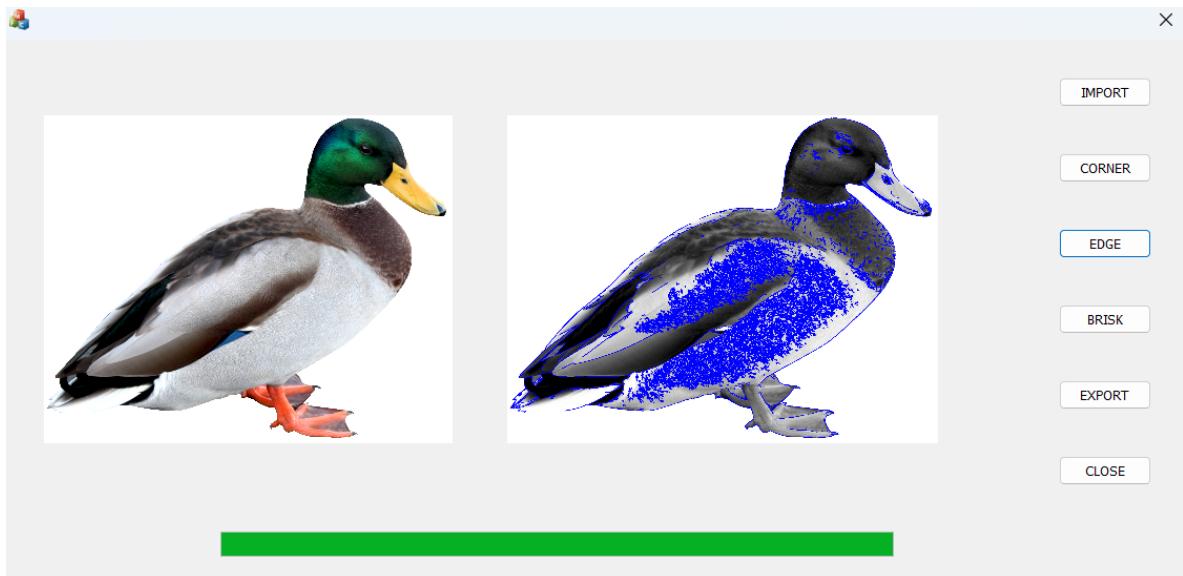


Figure 10: EDGE

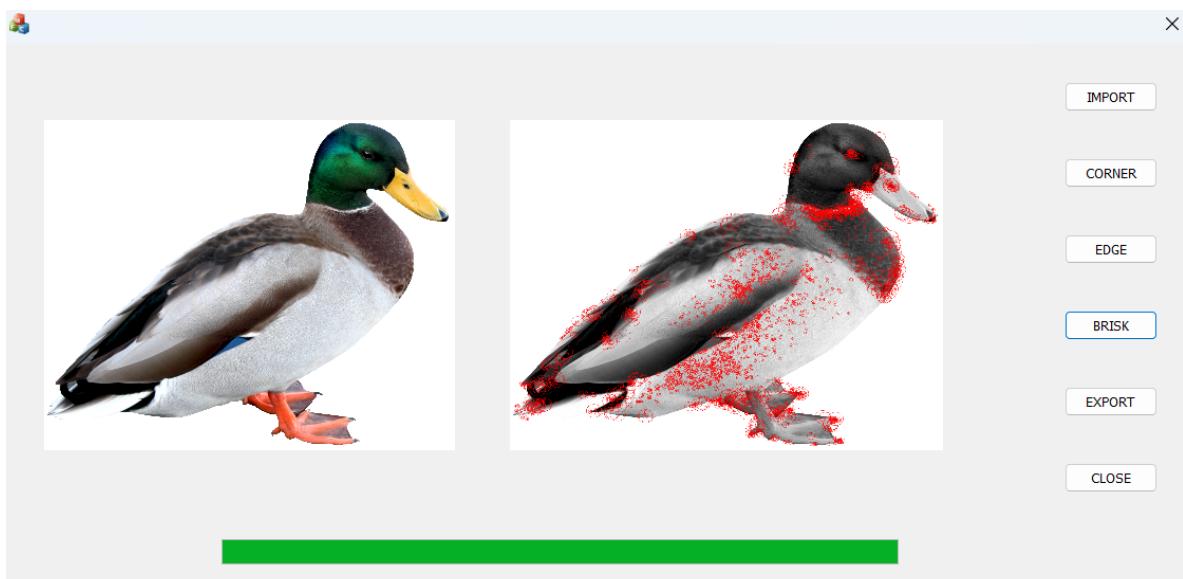


Figure 11: BRISK

5.3 TIFF Image Test



Figure 12: IMPORT

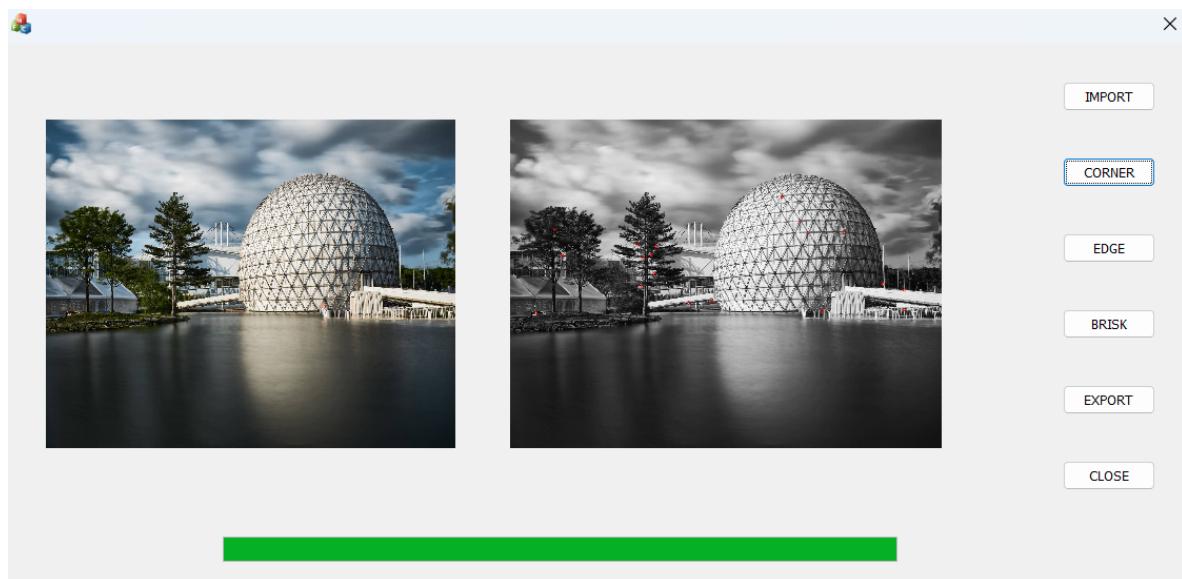


Figure 13: CORNER

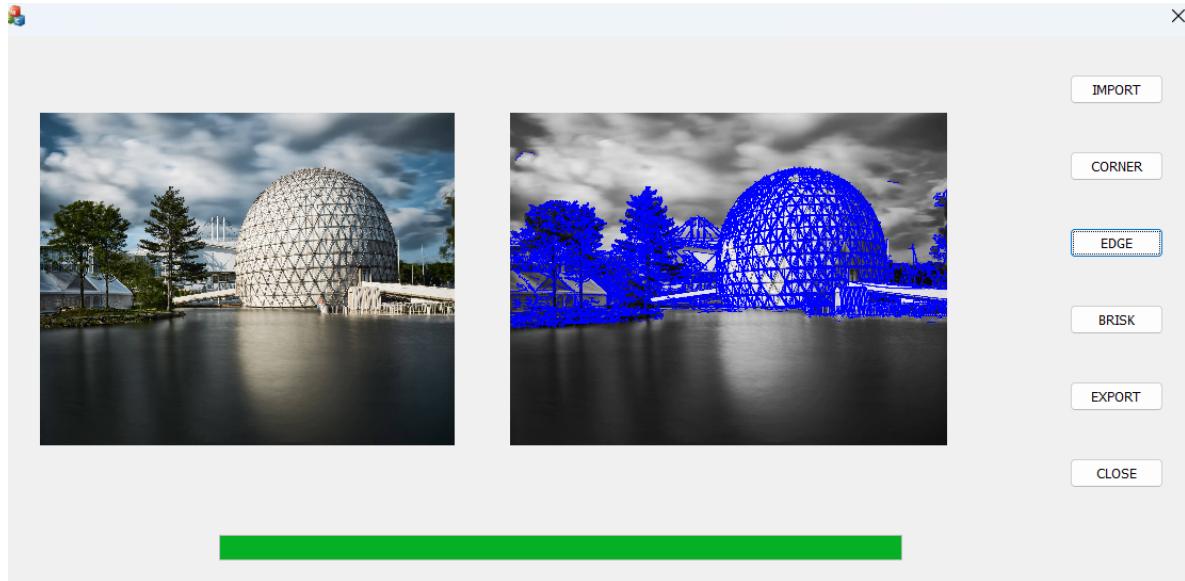


Figure 14: EDGE

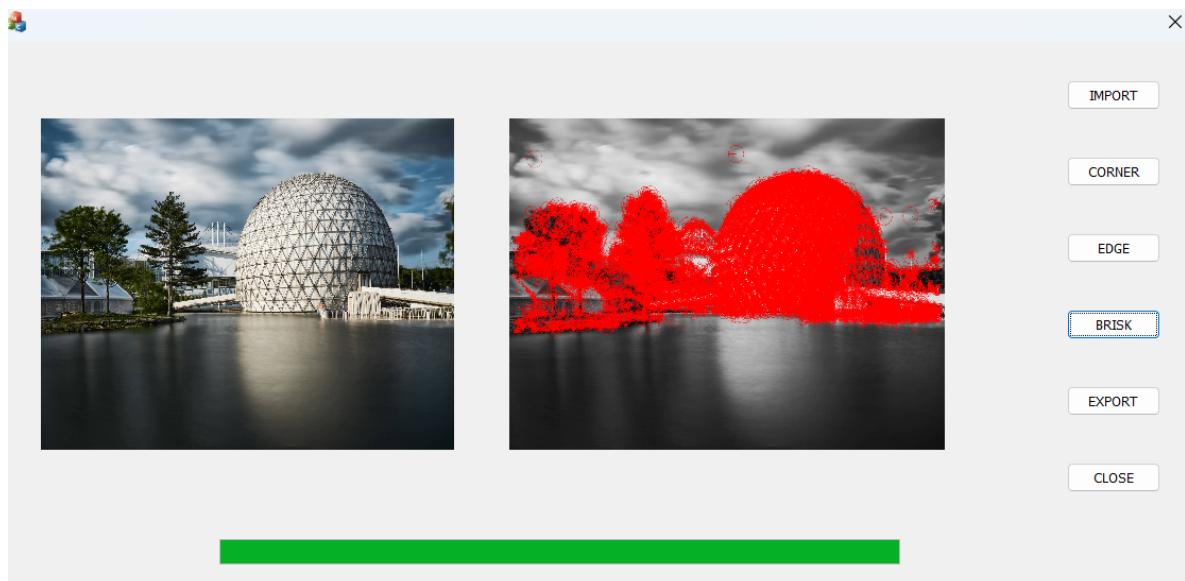


Figure 15: BRISK

5.4 BMP Image Test



Figure 16: IMPORT

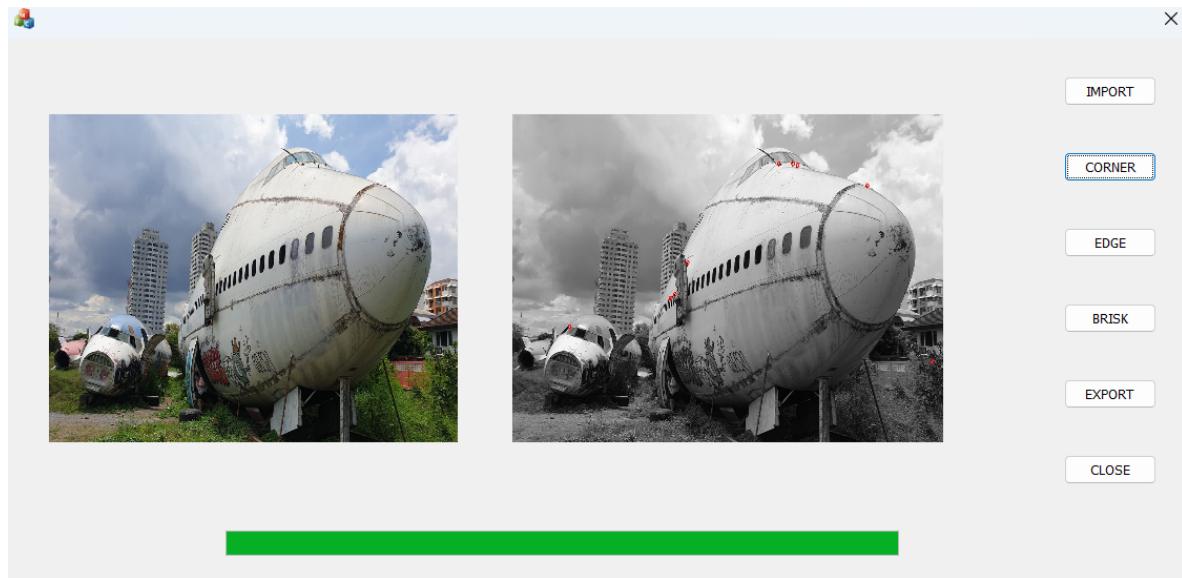


Figure 17: CORNER



Figure 18: EDGE

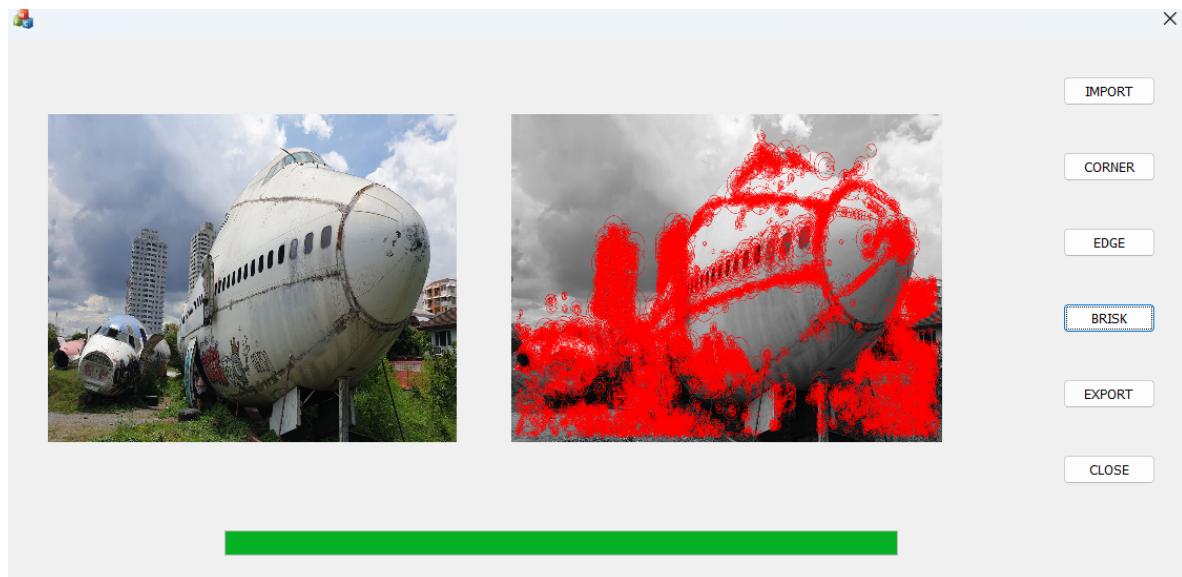


Figure 19: BRISK

6 Distribution

In order to make IFDS installable on Windows OS platform, we need to package all the needed files together including those .dll files.

Here, we use the build management function of Visual Studio to generate a .msi file which can be installed on different terminals of Windows OS.

1. Firstly, we need to install the "Visual Studio Installer Projects" in the Online section, and then download and install this extension.
2. Right click the Solution Explorer and select Add-New project. Under the Installed —visual studio installer, choose Setup Project.
3. Add—Project Output, and then View—File System, open the File System Editor.
4. Fill those Properties such as Manufacture, Product Name, Version, and then build the project. After that, the .msi of IFDS has been generated. Now we can distribute it to different users of Windows OS Platform.