Easy Image File Converter

April 17, 2024

Alexander Thompson

Bachelors in Computer Science

Professor Michael O’Neill

1. Statement of Purpose

Many popular software and websites don't accept all image file formats, which can slow down and frustrate the user, which can be solved with a program that can easily convert image files to desired formats. One of my hobbies is to edit videos, and images are used quite often during the process. Since some of these editing programs, like Adobe's Premiere Pro, have not caught up with current standards of storing images, the "WebP" format is not supported. WebP is used often on the internet due to it being efficient for storing images, so when saving an image from the internet to use in the program, an error message pops up showing it is not supported. The problem is that there are many image formats different programs and devices create that may not be supported on every platform where you need images, so people need a converter of sorts to turn their image into an accepted file format.

The reason why this problem is a big deal is because when someone, like me, is editing videos, editing photos, or even posting on social media platforms, a large amount of file formats are not accepted. To back up my assertions, according to Adobe's website, Adobe Premiere Pro does not accept the file format of "Webp". (<https://helpx.adobe.com/premiere-pro/using/supported-file-formats.html>) For another example, according to the social media platform Discord, the platform does not accept the file format of "Tiff" (<https://discord.com/developers/docs/reference#:~:text=Make%20sure%20that%20you're,file%20types%20are%20not%20supported>).

My solution is to create a desktop program that is an image file converter that allows a user to import and export the image to a desired file format. A benefit of the solution is that having an "all-in-one" program on a user's desktop allows for quick and efficient converting file formats without needing to spend time researching about formats and worrying if a format is accepted or not. There are websites that allow for the conversion of image files, but to open up the browser and find a website for the specific file format is a long process. My program will solve the problem of having an image file type that is not supported by a piece of software or social media website. The program will allow a user to import any image file type and convert the image to a desired file format.

1. Research & Background

With this project, I approached the idea with the mindset of both using knowledge I was comfortable with and knowledge I needed to research upon. During my years of university, I had knowledge on the C++ programming language and the many core concepts in Computer Science, so by choosing C++ as my main programming language for the project, I was able to focus my research on concepts directly related to the functionality of the program, which include:

1. QT Framework for C++: This framework allowed me to easily create a user interface for my C++ code, which I only knew how to create a desktop application with a user interface using the Java programming language. Since there were specific aspects different from the C++ version I was used to (e.g. variable types and certain libraries), there was a hefty amount of research along the way of development.
2. OpenCV for C++: This library for C++ was incredibly important for my program, as this library allows for my program to easily import, adjust, display, and export image files with ease. A lot of research was involved to install the library for the QT framework and how to use the library for my specific use cases within QT.
3. Creating an Installer: A key part of an easy-to-use program is to make access to the program as simple as possible for most people. An installer that will install all the required files and create a shortcut to access the program with ease involved research to figure out the best software to create an installer and to make sure the installer is working as intended.
4. Project Language, Software, and Hardware

The project’s language, software, and hardware requirements are as listed:

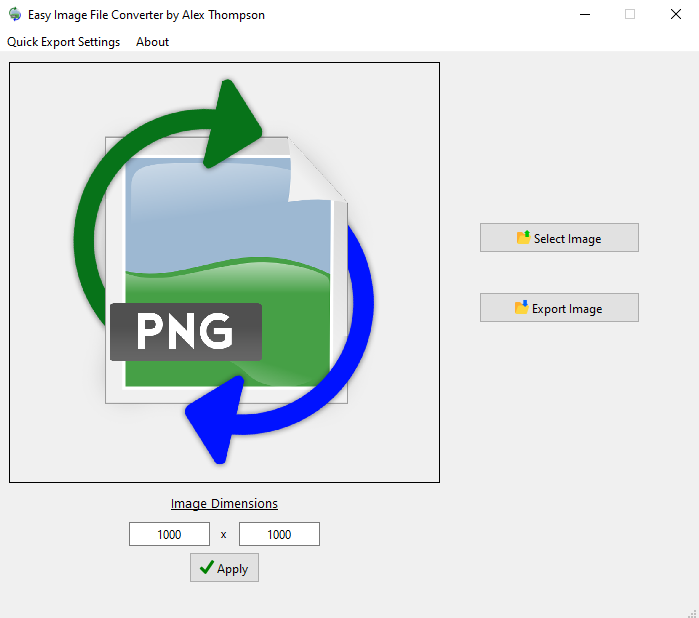
* Language: C++ with the OpenCV library using the QT framework
* Software: Windows 10 Operating System
* Hardware: Any computer running Windows 10 or above

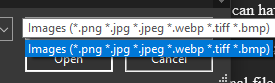
1. Project Requirements

This project’s requirements are located in a separate document here:

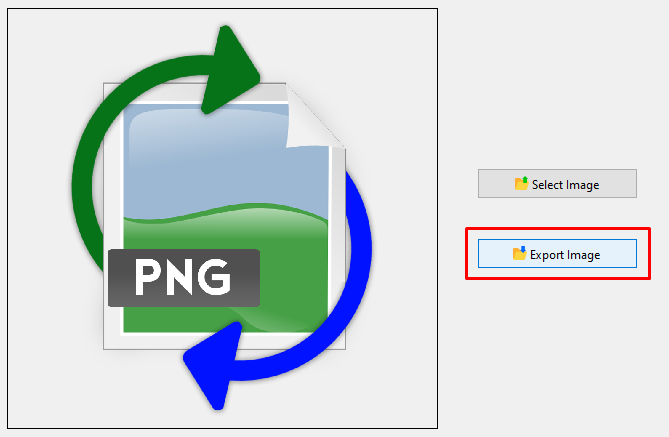
[Easy Image File Converter’s Requirements Document](https://github.com/AlexThomp1/Easy-Image-File-Type-Converter/blob/master/docs/Senior_Project_Requirements_Document.docx)

1. Project Implementation Description & Explanation

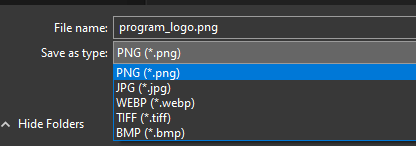
Once the program is installed, the user is greeted with a simple user-interface with different options, but a lot of the options are currently greyed out, as seen in Figure 1. The text “No Image Selected” appears at the left to help guide the user to select an image, which the only option is at the right, a button that says “Select Image”. A user-interface was very important to this project, as I wanted the program to be as simple to use as possible for everyday people.  
  
  
(Figure 1: The main user-interface of the Easy Image File Converter)

When the user presses “Select Image”, a Windows file explorer window appears and asks the user to find the image file they want to convert. All of the supported image types are displayed to the user at the bottom right of the window, as shown in Figure 2. The file explorer will only show files that match the accepted types to prevent users from attempting to import unsupported image file types.  
  
(Figure 2: All image file types the program is currently able to import)

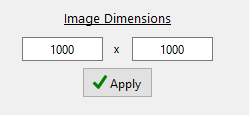
Once the user selects an image, the image is displayed in the box at the left, so the user has feedback that the program knows what image the user wants to convert, as seen in Figure 3. The option to “Export Image” is now available along with the option to change the image’s dimensions.

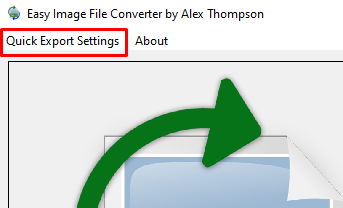
  
(Figure 3: The “Export Image” button is now able to be clicked on after selecting an image)

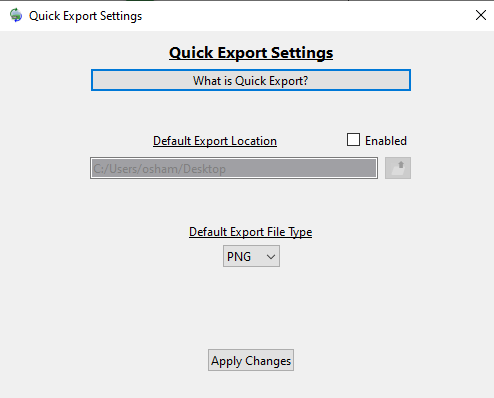
When the user presses “Export Image”, a Windows file explorer window appears and asks the user to export the image file they want to convert. The user can search for a desired file location to export the file, change the file name, and change the image file type to any of the supported image file types as seen in Figure 4. Once the image is saved, the program gives feedback to the user stating the image has been exported.



(Figure 4: The image file formats the user is able to export to)

If the user wants to change the dimensions of the exported image, the section titled “Image Dimensions” at the bottom left of the user interface has two boxes that show the current pixel dimensions of the photo, as shown in Figure 5. The user can enter their desired dimensions, and click on the “Apply” button, which the displayed image will reflect the changes. Once the image is exported, the changed dimensions will be correctly applied.  
  
(Figure 5: The dimensions of the image are displayed and able to be changed)

To convert multiple images in a rapid fashion, the user can have the program open, and drag and drop multiple images anywhere on the program, as seen in Figure 6. The image files will automatically be exported to the same location as the original file and be exported to the image file type of PNG. In order to change the default export location and default export file type, the user is able to change them in the “Quick Export Settings” at the top left of the program, as seen in Figure 7.  
  
(Figure 6: Drag and dropping 20 image files on top of the program to Quick Export)  
  
(Figure 7: The location of the Quick Export Settings button to access the settings)

If the user clicks on the “Quick Export Settings” at the top left of the program, the “Quick Export Settings” window pops up showing the settings the user can change, as seen in Figure 8. There is a “What is Quick Export?” button that gives more information about the purpose of the settings. If the “Enabled” button to the right of the “Default Export Location” is checked, the user is able to click the open folder icon to find a location on their computer that the Quick Export feature will export to automatically. If the “Enabled” button is not checked, the default location will be where the original image is located. The “Default Export File Type” drop-down box shows all the file types the program can export to, and the selected option will be the image file type the Quick Export automatically exports the image as. To apply the changed settings, the user must click on the “Apply Changes” button at the bottom of the window.  
  
(Figure 8: The Quick Export Settings window with all the options)

1. Test Plan

|  |  |  |
| --- | --- | --- |
| **ID** | **Requirement** | **Pass/Fail** |
| 1 | The program shall allow the user to import any image of the accepted image file formats. | Pass |
| 2 | The program shall convert the imported image file format into a desired image file format. | Pass |
| 3 | The program shall give the user multiple image file formats to export the imported file. | Pass |
| 4 | The program shall display the image the user imported into the program. | Pass |
| 5 | The program shall use a user-interface the user can click on to navigate. | Pass |
| 6 | The program shall be simple to navigate for users without a guide or tutorial. | Pass |
| 7 | The program shall be fast and efficient in the process of converting images. | Pass |
| 8 | The program shall have an icon next to each functionality to describe the user process. | Pass |
| 9 | The program shall be reliable to use for image file format conversion for the supported image file types. | Pass |
| 10 | The program shall stop the import process if the user selects an unsupported file type. | Pass |
| 11 | The program shall use images and icons for the user to visually understand the user process. | Pass |
| 12 | The program shall hide the details of its construction from the user. | Pass |
| 13 | The program shall convert images one at a time. | Pass |
| 14 | The program shall not ask the user for personal information. | Pass |
| 15 | The program shall not store the images uploaded by the user. | Pass |
| 16 | The program shall not be offensive to religious or ethnic groups. | Pass |
| 17 | The program shall continue to be used if no internet connection is available. | Pass |
| 18 | The program shall retain the user’s default export image file type. | Pass |
| 19 | The program shall be installed through an installer program. | Pass |
| 20 | The program shall help the user to avoid making mistakes. | Pass |

1. Test Results

This project’s test results are located in separate documents here:

[Easy Image File Converter’s Test Results Documents](https://github.com/AlexThomp1/Easy-Image-File-Type-Converter/tree/master/tests)

1. Challenges Overcome

There was a variety of challenges I had to overcome during the entire process of

constructing this project, which include:

- Time Frame: Since I decided to do both construction and defense within the same semester, I needed to build at least the prototype during my winter break between my two final semesters. I was able to successfully construct a prototype during that time frame, and was able to add all the features I wanted and bug test during my final semester and in a reasonable time frame.

- Installing OpenCV for QT: The installation of this library for use within the QT framework was surprisingly a very convoluted process. After following a video tutorial and written guide involving many steps, there were many times I was unsure I installed the library properly. Once I was able to properly display an image through a test program in the QT framework, this challenge had been overcome.

- Displaying Images on the UI: With the OpenCV library, in order to display images   
within the QT user interface, there was a specific type the framework needed. I had to research and find a way to convert the imported image with OpenCV to an image type QT allowed to be displayed on the user interface.

- Creating Persistent Settings: Since the Quick Export feature does everything automatically, a place where users can adjust these settings was important. I wanted it where the settings the user changed would still be the same if they closed and opened the program, so I had to do research and learn how to do so. I was able to overcome this challenge by using an INI file that I was able to save information and pull information from on the program opening.

- User Interface Design: QT has a built-in user interface designer, which allows for me to easily create a user interface with C++ code. The issue with the designer is the complications of trying to make a user interface that scales to different window sizes and looks the way you want. I decided to make it look simple and clean, as the scaling created a layout that looked not as appealing.

1. Future Enhancements

Due to the program being all about images and conversion of images, there are quite a few enhancements that I could add to make the program more useful for multiple types of situations, which include:

- More Image File Types: In future versions, I would want to include more potential image file types, like ICO, SVG, and maybe even PDF files.

- Color Adjuster: This would allow users to adjust the color values of the image they select, like the saturation, hue, and more.

- Adjustments of the User Interface: I would like to make the user interface look more modern and unique, as the user interface has a basic visual look to it.

- Image Link Importing: Being able to copy the direct URL from an image online, which the program would download and import would save a few extra steps for people who want to download an image online and convert the image to the desired image type.

1. Defense Presentation Slides

* Need to discuss about how I should approach the slides.