Image Hunter

Raster Search tool for the rest of us

Written by Jean Lalande

August 27th 2010

Introduction

*ImageHunter* is an image search tool. It is very useful when you need to find a raster of a given format with specific characteristics such as histogram support, block type of a restricted size, minimal resolution, etc. It can search recursively through as many folders as needed and supports network computers.

When executing a search, the application gives instant feedback to the user and offers to cancel the action whenever the user requests it. As soon as an image fits the given search criteria, it is added to the list of results and is immediately available to do work with it throughout the interface while the search goes on. The user can right click on an image or multiple images and choose among the list of actions available. It is possible to copy the images to a given directory, export the list to the clipboard and many others. External tools can also be added to improve the workflow of the user.

Inside ImageHunter

#### Search criterias

Here is the list of currently supported search criterias:

* Block type
* Compression Codec
* Geocoding
* Histogram
* Image format
* Image size
* Multiple pages
* Multiple resolutions
* Pixel type
* Scanline orientation
* Tag
* Transformation model

To create another criteria, a new class needs to inherit from the interface ICriteria.h. This interface specifies two required classes for a new criteria: ICriteria, which is the criteria itself, and ICriteriaBuilder that knows how to build this very criteria. The new builder will be registered just like the other ones in MainUI::MainUI\_Load and will be automatically added in the search criterias list.

In the list above, some of the criterias are additive and others are not. As an example, it is possible to specify several Pixel Type and if one the criteria applies to the image, it will be listed in the result. The way it is handled when choosing Image Size is different since an image must complies with every specified image size to be treated as a valid image.

#### The Hunter

*The Hunter* class runs in its own thread once it is launched. It is using a BackgroundWorker to call the UI whenever it needs to be updated. This way, it is possible to have instant feedback from the *Hunter* and a progression percentage that we can then display into the progress bar and in the title of the application.

It’s also in this class that the Raster Capabilities are loaded. When implementing a new criteria, it is more than plausible that one needs to add additional capabilities in Hunter::Init function.

#### External tools

To improve the workflow and make the tool more useful, a list of external tools is customizable. At this time, the default options include ImageInsider, HTiffInfo and ActiveImage. When a new entry is added to the list, it will appear in the contextual menu fired when the user right-clicks on a line result. If the new tool is clicked, a new process is launched using the entered parameters from the External Tools dialog.

#### Options

An Option dialog exists in *ImageHunter* to specify the excluded extensions when doing a search. A predefined list is available at first start. The user can simply add more extensions to the list to make the search function faster. Otherwise, all files with extension different than the one in the list will be opened by the *Hunter*.

Another option in the window is the possibility to include the SendTo menu in the contextual menu of the results list. When checked, all available shortcuts put in the SendTo folder of the user will be added. Not all shortcuts are supported though. For a shortcut to work, it must be a .LNK file and should comply be the standard syntax. Otherwise, it is skipped. The default values in the SendTo menu of Windows are special crafted commands that it is not possible to support at the moment. Further investigation would be required.

Work to do

Some features can be added to this tool to make it even better. First of all, some tweaking can find their ways into the search algorithm to decrease the time required to complete a search. It is not that bad actually but it could certainly be better.

Of course, other criterias could be added to the actual list to make *ImageHunter* a more complete application. Several people asked for advanced research field in the Geocoding criteria. Unfortunately, it is not as easy as the other criterias to implement Geocoding. In fact, a whole new application could be written only to support Geocoding research.