

User Documentation

MoveImagesToFrame

Version 0.11.0

Overview

If you have more photos / images than will fit on your Pix-Star Picture Frame at one time you may find this program helpful. It will “rotate” files from your Home Computer / Laptop to your Pix-Star frame “automatically”. It uses the frame’s Samba server to access memory of your Pix-Star frame. I choose to use Samba instead of UPnP/DLNA since I already the Serviio UPnP/DLNA server (that doesn’t work with the frame) and adding other UPnP/DLNA servers resulted in instability of the connection to the frame. Samba has been rock solid.

Preprocessing of photos helps

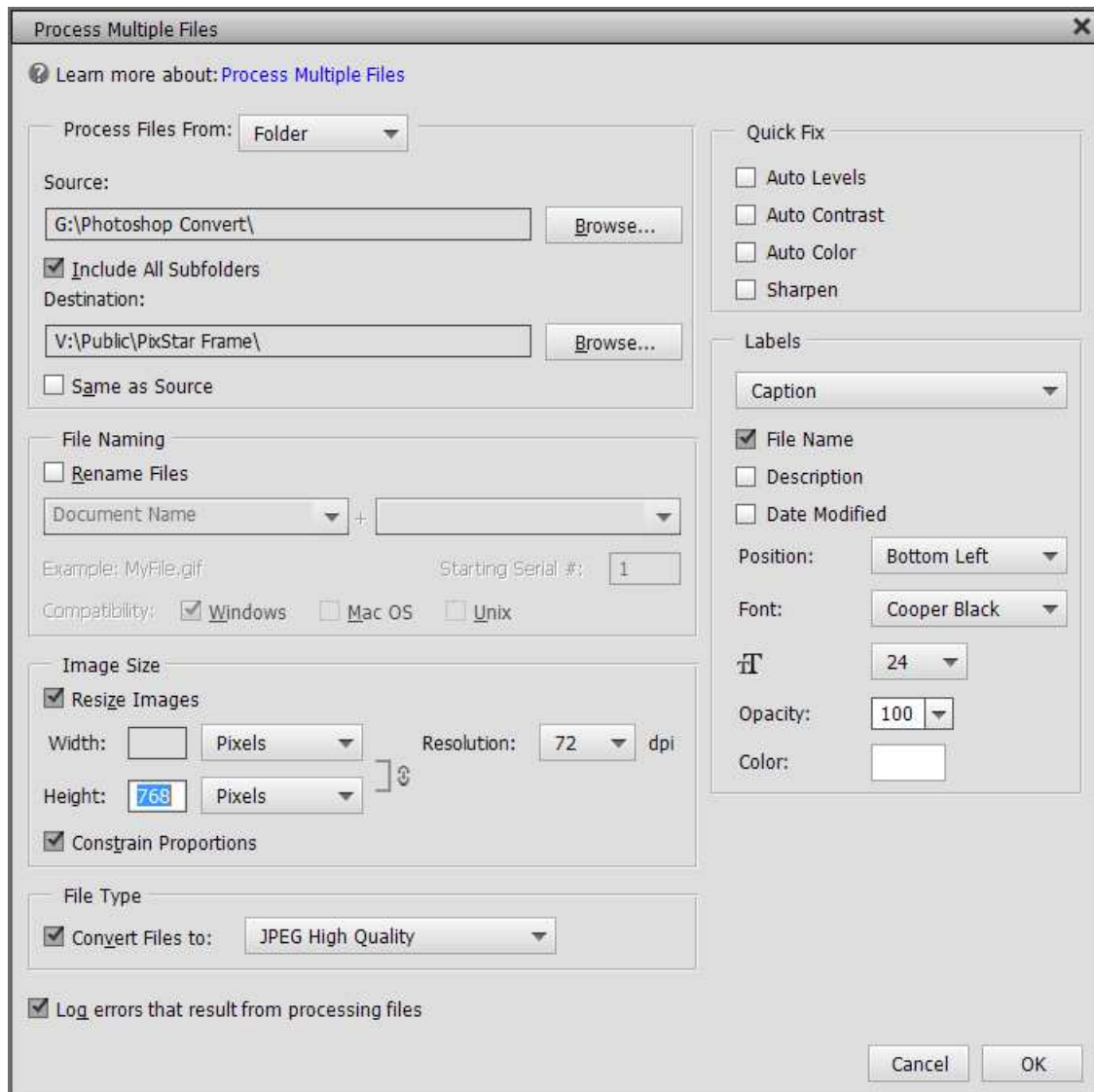
Preprocessing of the photos is not required; however, you will be able to hold a lot more photos on your frame if you do. This is something that Pix-Star does for you automatically if you use Web Albums.

Recommended Preprocessing:

1. Resize your photos to the frame’s native resolution; see the Pix-Star Knowledge Base article [When sending pictures by email to my Pix-Star which size is the most optimized?](#)

How to Resize your photos

- a. Create multiple Web Albums on <https://www.pix-star.com/albums/list/>. Why multiple Web Albums? Pix-Star recommends no more than 3000 photos in a Web Album. For me that would be 9 Web Albums.
- b. After all the files have uploaded to your Web Albums then download them back to your computer into a directory (also sub-directories) used just for photos to be sent to your frame. You now have all of your photos in your frame’s native resolution.
- c. Or use a photo processing application such as Photoshop Elements that supports batch processing of images. For Photoshop Elements that is: File -> Process Multiple Files...



2. I love having a caption on my photos (with 25,000+ I don't remember all of them). When I download photos from a camera or cellphone I create a sub-directory whose name is descriptive of the photo contents ("Ireland London 2006" for example). If I am really industrious I add a description to each photo (no really, I do, sometimes, every once in a while, maybe). I don't want to alter the original image (except for correcting problems in the photo) so I have a program that changes the photo's file name to have the directory structure added to it ("[Ireland London 2006 ~Christina's Canon] P1000427 #181180.JPG" for example). If you look at the screen shot above for "Process Multiple Files" you can see that under "Labels" I selected "Captions" and checked "File Name" (and more options). This adds a caption to the photo that explains it was from Ireland and London trip in 2006 and was taken by Christina's Canon. If anyone is interested in the file name modifying program I can make it available or integrate into MoveImagesToFrame (see [Enhancements Planned](#)).

Description of the command line parameters

As of version 0.11.0 both the original positional command line parameters and the new Keyword command line parameters are supported.

Keyword command line parameters

Below is the output from “-?” or “-help”, most commands have a short form that starts with a dash “-“ and is one character (case sensitive). Most commands also have a long form starting with 2 dashes “--“ and a more descriptive keyword. The long form does exactly what the short form does. Some commands require a single argument, this is designated by “<arg>”. Those arguments that are file paths can contain spaces, if they do contain spaces they must be surrounded by double-quotes, i.e., “d:\MoveImagesToFrame Database”.

Some of the commands are required, they have “(Required)” in their description.

usage: MoveImagesToFrame

-?	print this message
-D,--Debug	print debugging information
-d,--database-dir <arg>	Database directory (Required)
-f,--frame-dir <arg>	Picture Frame Directory (Required)
-help	print this message
-l,--list-files-only	list files only (frame & source), don't rotate files
-log <arg>	log file to append all output to
-m,--megabytes-free <arg>	number of Megabytes to leave free on frame (Required)
-p,--percent <arg>	% of frame images to change (1-100) (Required)
-q,--quiet	quiet mode, most output not displayed
-s,--source-dir <arg>	image Source Directory (Required)
-v,--version	print the version information and exit

Positional command line parameters

The original command line parameters were positional, i.e., you must enter them in order. For the two optional parameters if you wish to enter the second optional parameter you must also enter the first.

Parameters and order:

1. <Picture Frame Dir> = The fully qualified path to the directory on the Picture Frame to copy images. Note: enclose in double quotes.
2. <Source Dir> = The fully qualified path to the directory containing the images to rotate to the Picture Frame, Note: enclose in double quotes.
3. <% to Change> = The percentage of the photos on the Picture Frame to replace, a number between 1 and 100.

4. <Database Dir> = The fully qualified path to the directory where the program can store information about the files in <Picture Frame Dir> and <Source Dir>. Note: enclose in double quotes.
5. <Megabytes to leave free> = Number of Megabytes (1,048,576 bytes) to leave free on the frame for other pictures such as Email.
6. <Log File> = (Optional) The Log File, (default: console (DOS box)) all output will be appended to this file. Note: enclose in double quotes.
7. <List Files Only> = (Optional, default: false) Only list the files on frame and Source Dir (true or false). If true no files are deleted or copied.

Installation on a PC

Note: The instructions below are for Windows 10, earlier versions of Windows also have this capability, however, the steps may be different.

Setting up Samba on your Pix-Star frame

MoveImagesToFrame accesses your Pix-Star Picture Frame memory by using the frame's built-in Samba server.

See the Pix-Star Knowledge Base articles below on Samba configuration:

[How to change SAMBA password \(Connection from PC password\)?](#)

[How to manage the photos stored inside my Pix-Star Photo frame wirelessly from my computer \(using SAMBA\)?](#)

[How to access pictures stored inside my Pix-Star over local network \(SAMBA\) with Windows 8.x?](#)

Installing MoveImagesToFrame

MoveImagesToFrame is written in Java and needs a JRE (Java Runtime Environment) to be installed. It needs version 1.7 or above. If you do not have Java installed on your PC you will need to install one. If you are not sure if you have Java or what version it is see the select in this document: [Determine if the JRE \(Java Runtime Environment\) is in Command Path](#)

Place the executable jar file “execMoveImagesToFrame<Version>.jar” in a directory that will be available when you wish to run MoveImagesToFrame. **Note:** <Version> is the version of MoveImagesToFrame you are using, currently “0_10”. The directory you place the executable jar in is referenced below as <Install Location>.

Determine if the JRE (Java Runtime Environment) is in Command Path

1. Open a “Command Prompt” (DOS Box) by clicking “Start”, click “All Apps”, scroll down to “Windows System” and click on it, then click on “Command Prompt”. This opens the “Command Prompt” window.
2. Type in the following:
java -version<Press Enter>
If you get an error message such as “'java' is not recognized as an internal or external command,” you don’t have the JRE in the Command Path (or you don’t have Java at all).

Determine if you have Java Installed

Go to java.com/ and click on “Do I have Java?”, this will take you to “Verify Java Version”. Click on the “Verify Java Version” button and follow the prompts.

Finding Java if installed but not in the Command Path

Open the “Configure Java” App by clicking “Start”, click “All Apps”, scroll down to “Java” and click on it, then click on “Configure Java” this opens the Java Control Panel. Click on the “Java” Tab and then the “View...” button. Make sure the “User” Tab is selected and that at least one JRE is displayed (Platform must be 1.7 or above). Double-click the “Path” entry of the JRE you want to use. Do “Ctrl-A” (Select All) and then a “Ctrl-C” (Copy) and paste it into a Text Editor (such as Notepad), you should have something like this:

C:\Program Files (x86)\Java\jre1.8.0_73\bin\javaw.exe

Change javaw.exe to java.exe then add double quotes around the line above then add (without the single quotes):

‘ -jar ‘

The result should look like something this:

“C:\Program Files (x86)\Java\jre1.8.0_73\bin\java.exe” -jar

Once you have a line customized like the one above you will need to save it and add it in places that I will refer to as <Execute Jar>

Examples of running the program on a PC

Run in a "Command Prompt"

1. Remember, if you determined that Java is in the Command Path then <Execute Jar> is ignored below,
2. "Start", click "All Apps", scroll down to "Windows System" and click on it, then click on "Command Prompt". This opens the "Command Prompt" window.
3. Type the following text (modified for your environment) after the ">" prompt.
<Execute Jar> "<Install Location>\execMoveImagesToFrame<Version>.jar"
<Picture Frame Dir> <Source Dir> <% to Change> <Database Dir> <Megabytes to leave free> <Log File> <List Files Only>
4. Press the "Enter" key. This starts execution of MoveImagesToFrame.
5. If MoveImagesToFrame did not find any errors on the command line and your Pix-Star frame is online you will see output in the Command Prompt Window (or the Log File if you entered <Log File>). It will tell you what it is doing and any problems it finds.

Run from a Shortcut on Desktop

1. Create a new Shortcut (right-click on Desktop->New->Shortcut)
2. Place the following text (modified for your environment) in the box labeled "Type the location of the item:"
<Execute Jar> "<Install Location>\execMoveImagesToFrame<Version>.jar"
<Picture Frame Dir> <Source Dir> <% to Change> <Database Dir>
<Megabytes to leave free> <Log File> <List Files Only>
3. Click "Next"
4. Modify the name of the Shortcut if you wish and then click "Finish".
5. By double-clicking the Shortcut you just created you run MoveImagesToFrame.

More detailed description of the program internals

1. Processes the command line parameters and performs validity checks.
2. Starts process of "rotating" files (delete from frame, copy new files from source). It reads the 2 files in the Database directory. The 1st file ("lastModDB.fdb") contains the "Last Modified Times" of the Frame directory and The Source directory. The 2nd file ("fileDB.fdb") contains information on all files that were in the Frame and Source directories after the last run. The current "Last Modified Times" of both directories are compared with those saved in "lastModDB.fdb", if they are the same we have all information we need on the files on the frame and in the Source directory. Any directory that has been modified since the last run is re-scanned and the database files are updated. If one or both files are missing from the Database directory that information is created.

3. Once we have an up-to-date list of all files on the frame and in the source we sort the list of files on the frame from longest time on the frame to shortest time on the frame. We also sort the list of files in the source from the longest time since they were on the frame to shortest time since they were on the frame.
4. We then calculate the number of files to delete from the frame, this is “total number of files on the frame * <% to Change>”.
5. Starting with the file that has been on the frame the longest we delete the number of files calculated above.
6. Once we have deleted the correct number of files on the frame we copy files from the source to the frame, starting with the source file that has been off the frame for the longest time.
7. We continue copying from source to frame until the remaining frame disk space has reached the <Megabytes to leave free> from the command line.
8. As we are deleting files and copying files we output what files are being processed. We also update the database records to indicate that a file is no longer on the frame or that a file is now on the frame.
9. At the end of the process we update the 2 database files. This is done since scanning the frame memory remotely is much slower than scanning the local Source Directory (unless it is really huge).

Enhancements Planned

1. Keyword based command line parameters so that you can enter command line parameters in any order, easier until a GUI is available. Will be backward compatible with positional parameters so that you don't need to change parameter entry style unless you want to.
2. A GUI to enter, save and verify execution parameters, this would also display the program execution messages and allow saving them to a Log File.
3. Install, need to automate that whole install section. I didn't realize what a mess installing Java and executing Java Programs has become since I have had Java on all of my machines for years.
4. Random deletion from frame and random selection from Source. Currently the order of source files moved to the frame stays the same even though files are moved to and from the frame. By this I mean that if pictures A and B were originally copied to the frame together the 1st time they will most likely always be on the frame together, this may not be what you want.
5. Explicit list of source directories to process instead one directory and all of its sub-directories. This is something I need myself since I have 500+ folders and not all of the folders have pictures that need to be displayed. Currently I copy the folders I want displayed to a temporary directory, a long process.
6. Image Processing built-in:
 - a. Resize image to frame's native resolution. On-the-Fly (if fast enough) or resized the 1st time and saved in another directory on the PC for later runs.
 - b. Optional captioning using:
 - i. File name or
 - ii. Directory Structure - File name

iii. Fixed Caption

Source Availability

Built with Eclipse IDE (Mars.1) with Java 1.8_73 (Compiled at 1.7 compliance level).

Source available from:

- Pix-Star Support
- GitHub
 - <https://github.com/delbSJ/MoveImagesToFrame>