NVC v1 Application Document 11.10.24

Overview

The NVC utility monitors a source folder for .mov video files and convert them to .mp4 format automatically and saves them in a destination folder. It uses FFMPEG to handle video conversions and allows the user to set source and destination folders, and start/stop the monitoring and conversion process.

Key Features

- **Continuous Monitoring**: Once started, the application will keep checking for new .mov files in the source folder until it is stopped.
- **FFMPEG Integration**: Uses FFMPEG to ensure efficient video encoding.
- **Interface**: Allows users to select folders, view status, and start or stop monitoring and converstion.
- **Threaded Operation**: Runs in a separate thread to keep the interface responsive while converting videos.
- **No File Deletion**: The application does not delete any files from either the source or the destination folder.
- **Settings Persistence**: The selected source and destination folders are saved, so they remain the same the next time you run the application.

How to Use the Application

1. **Select Source Folder**: Click on the "Select Source Folder" button to choose the folder containing .mov files to be converted.
2. **Select Destination Folder**: Click on the "Select Destination Folder" button to choose where the converted .mp4 files will be saved.
3. **Run/Stop Conversion**: Press the "Run" button to start monitoring the source folder and converting files. The button will display "On" when running. Press it again to stop the process.

FFMPEG Integration

FFMPEG is packaged with the executable version of this application, so no additional download is necessary if you are using the compiled .exe. If you are running the Python script, you will need to have FFMPEG installed. You can download the required build of FFMPEG from the following link:

- Download FFMPEG

Source Code and Building Your Own Version

This application was written in Python. The source code for the application can be found at the following GitHub repository:

- Video Converter Source Code

If you wish to create your own build of the application, I used PyInstaller. Below is the command that worked best for compiling the script into an executable:

pyinstaller --onefile --windowed --icon=icon.ico --add-data "icon.ico;." --add-data "logo.png;." --add-data "ffmpeg.exe;." VCN.py

This command ensures that all necessary resources, including the icon and FFMPEG executable, are included in the build.

Troubleshooting

- **FFMPEG Not Found**: If you see the message "ffmpeg not found. Please check the path.", ensure that ffmpeg.exe is in the same directory as the script or executable.
- **Application Freezes**: If the application appears to freeze, it may be waiting for a background process to complete. Wait a few seconds for the thread to finish cleanly.
- **Add FFMPEG to System PATH**: If you are running the script version and encountering issues with FFMPEG not being found, you can add FFMPEG to your system's PATH environment variable. This will make it accessible globally. To add FFMPEG to the PATH:
    - Download FFMPEG from https://www.gyan.dev/ffmpeg/builds/.
    - Extract the downloaded folder and locate the bin directory (which contains ffmpeg.exe).
    - Copy the path to the bin directory.
    - On Windows, open the Start Menu, type "Environment Variables", and select "Edit the system environment variables".
    - Click on "Environment Variables".
    - Under "System variables", find and select "Path", then click "Edit".
    - Click "New" and paste the path to the bin directory.
    - Click "OK" to save all changes.
    - After adding FFMPEG to the PATH, restart your command prompt or IDE, and the script should be able to locate ffmpeg.exe without issues.