



# Manual



## **1. Mode: Easy**

[1.1 Step 1: Collect Files](#)

[1.2 Step 2: Configure Options](#)

[1.2.1 Processing Options](#)

[1.2.1.1 Command  
Options](#)

[1.2.2 File Handling](#)

[1.2.3 Output Directory](#)

[1.2.4 Output Format](#)

## **2. Modus: Command Line Interface (CLI)**

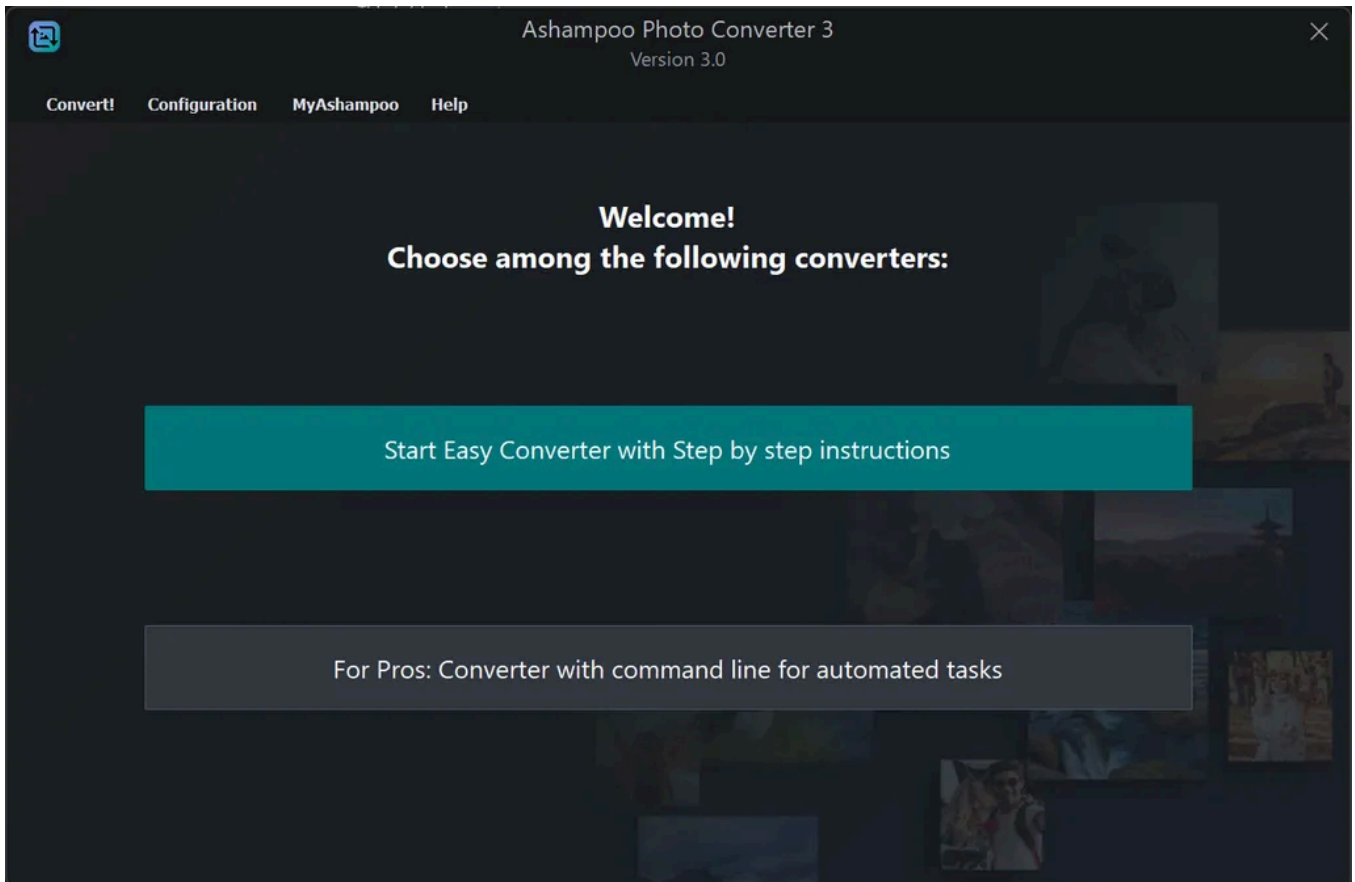
[2.1 Collecting Files](#)

[2.2 Output Format](#)

[2.3 Output Directory \(Optional\)](#)

[2.4 Command Options \(Optional\)](#)

[2.5 Flags \(Optional\)](#)

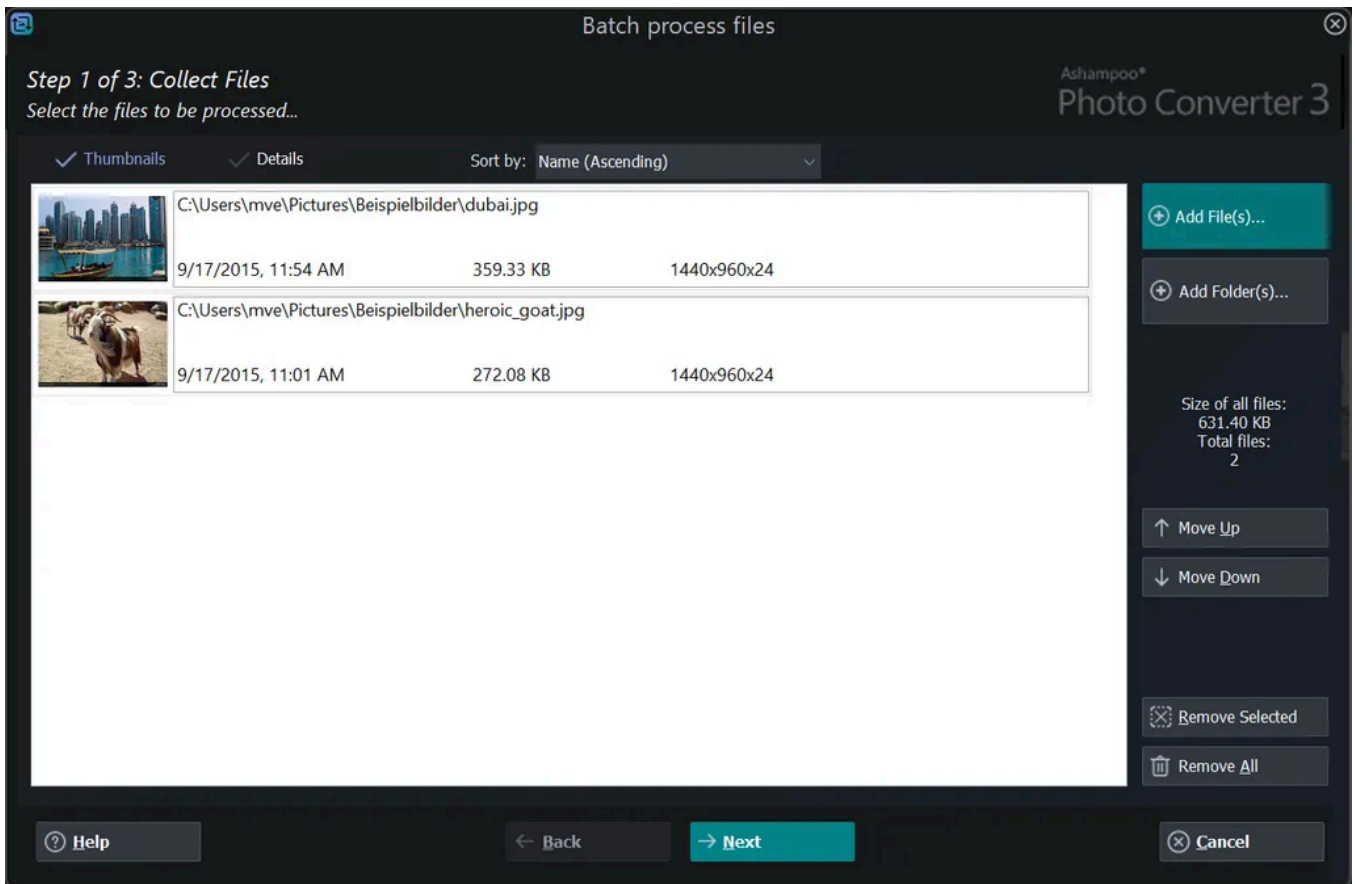


The program offers an easy mode with a graphical user interface and a command line interface.

# 1. Mode: Easy

This mode uses a graphical interface optimized for mouse operation.

## 1.1 Step 1: Collect Files

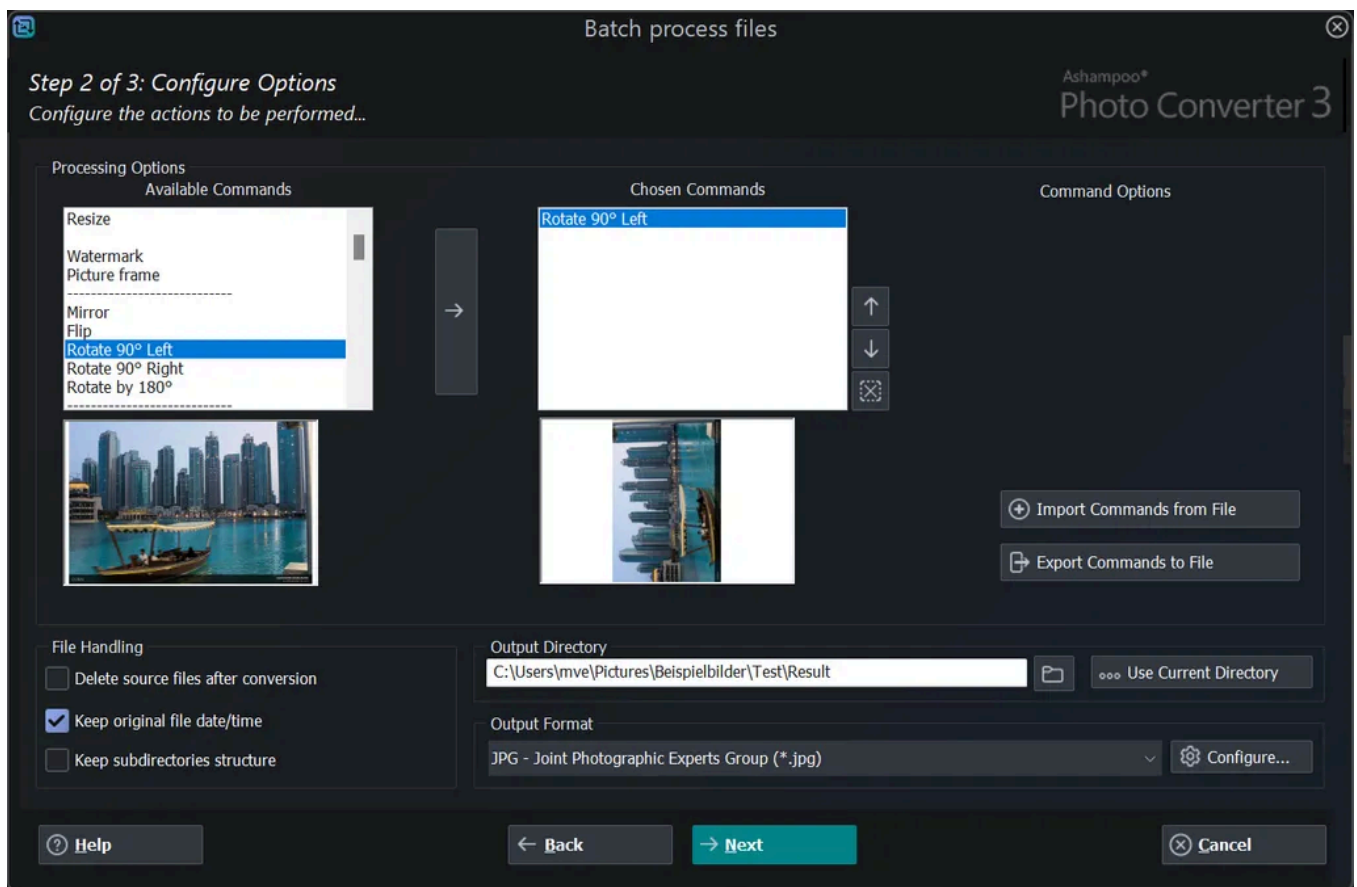


Click "Add File(s)" and select the image files to convert. You can also drag and drop your files in the program window. Hit "Add Folder(s)" to add all files from a folder and contained subfolders.

Click existing entries to change their order or to remove them.

The top-most entry is used for the preview in step 2.

## 1.2 Step 2: Configure Options



## 1.2.1 Processing Options

Supported operations appear under "Available Commands". Click an operation followed by the arrow button to add it to the processing queue. Use the arrow buttons under "Chosen Commands" to modify the order of operations. Hit "X" to remove an operation.

The images allow you to preview the expected outcome. The original image is always on the left, the expected outcome on the right.

### 1.2.1.1 Command Options

You can export the current operations and their order to a text file to reuse (import) them at a later time.

You can also use the file in [CLI mode](#) along with the "-commands" parameter e.g., to apply image operations as part of a batch script.

## 1.2.2 File Handling

Here you can decide if the original files should be deleted after conversion, and/or if their original date should be applied to the output files. If you've added entire directories and subdirectories, you can also choose whether to maintain the original directory structure for the new files. Otherwise, all output files will be placed in the same directory.

### 1.2.3 Output Directory

Set the desired output location and format here. Use "Use Current Directory" to save it to the location of the most recently added image file.

### 1.2.4 Output Format

Using "Configure," you can adjust additional parameters like "Quality" for the selected output format. Different parameters are available depending on the image format.

## 2. Modus: Command Line Interface (CLI)

This mode runs in a command line and is optimized for keyboard operation.

### 2.1 Collecting Files

You can select files using the "-source" parameter. It supports both complete file/directory paths including the "\*" wildcard character, as well as text files with a list of file paths.

Example	Description
-source "C:\temp\Image.jpg"	The file "Image.jpg" is being used at the specified path.

-source "C:\temp"	All image files under "C:\temp" directory will be converted.
-source "C:\temp\Image*.jpg"	All JPG files in the "C:\temp" directory that start with "Image" and have any additional characters, like "Image1.jpg", will be converted.
-source "C:\temp\Files.txt"	All files for the file paths from "Files.txt" will be converted.
"Files.txt" contents:	
C:\temp\Converter\Image1.jpg	
C:\temp\Converter\Image2.jpg	

## 2.2 Output Format

For the output format, use the file extension of the respective format without a ".", like "jpg" or "png". Depending on the format, you might be able to set the quality numerically (1-100) using the "-outputquality" parameter.

Example: -outputformat jpg -outputquality 85

You can find the list of available output formats under [Output Format](#) in easy program mode.

## 2.3 Output Directory (Optional)

The output directory is set using the "-destination" parameter, for example, -destination "C:\temp\results". If not specified, the source directory will be used.

## 2.4 Command Options (Optional)

Besides converting to another image format, you can use all image operations from the simple program mode. To do this, you first need to [export](#) the desired operations into a



.apcbatch file. After that, you can use them with the "-commands" parameter.

Example: -commands "C:\temp\commands.apcbatch"

## 2.5 Flags (Optional)

Name	Value	Description
-deleteoriginalfiles	yes, no	Deletes the original file after conversion.
-keepdirectorystructure	yes, no	Keeps the original directory structure. Requires -sourcewithsubdirectories.
-keeporiginalfiledate	yes, no	Applies the original file date to the new files.
-logging	yes, no, Ausgabepfad	Creates an activity log and either displays it in the command line window (yes, no) or saves it to an output path, for example, "C:\temp\Converter.log".
-overwriteexistingfiles	yes, no	It overwrites existing files with the same name.
-sourcewithsubdirectories	yes, no	Recursively convert all files in the subdirectories of the selected source directory.