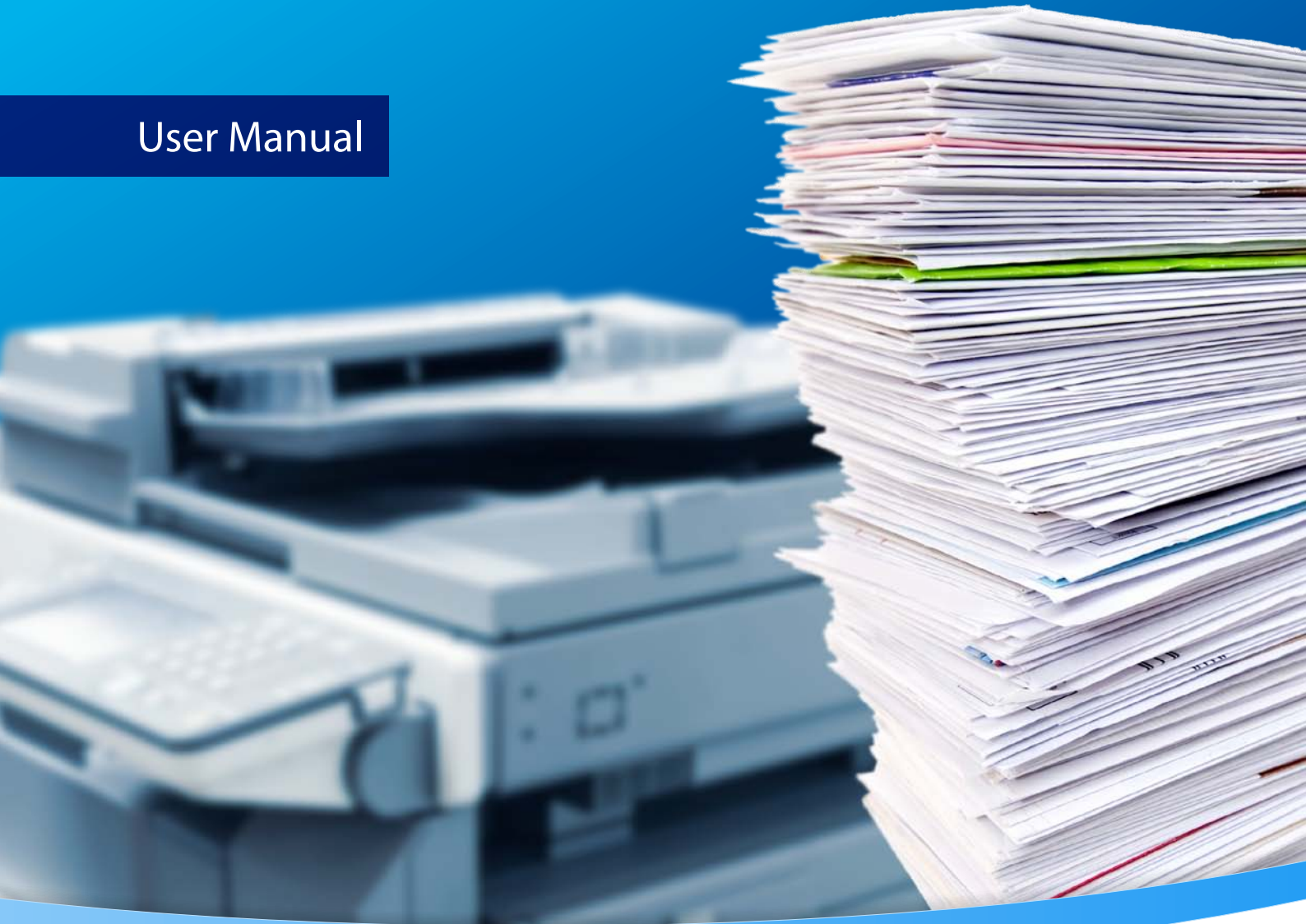


User Manual



3-Heights® Image to PDF Converter Shell

Version 6.27.6



Contents

1	Introduction	4
1.1	Description	4
1.2	Functions	4
1.2.1	Features	4
1.2.2	Formats	5
1.2.3	Conformance	6
1.3	Operating systems	6
2	Installation	7
2.1	Windows	7
2.1.1	How to set the environment variable "Path"	7
2.2	Linux and macOS	8
2.2.1	Linux	8
2.3	Uninstall	9
2.4	Color profiles	9
2.4.1	Default color profiles	9
2.4.2	Set other color profiles	9
2.4.3	Get other color profiles	10
2.5	Fonts	10
2.5.1	Font cache	10
2.6	Note about the evaluation license	10
2.7	Special directories	10
2.7.1	Directory for temporary files	10
2.7.2	Cache directory	11
2.7.3	Font directories	11
3	License management	13
4	Getting started	14
4.1	Specify the folder of the output file	14
4.2	Using wildcards (*)	14
4.3	Converting images to images	15
4.4	JPM to PDF/A	15
4.5	PdfOcr	16
5	Interface reference	17
5.1	Options for image to PDF	17
5.1.1	@<filename> Use a control file	17
5.1.2	-a Adjust the page size to the size of the image	17
5.1.3	-aa Set alternate text	17
5.1.4	-al Set language for alternate text	18
5.1.5	-ao Adjust page orientation	18
5.1.6	-ax Add XMP metadata	18
5.1.7	-b Set bits per pixel	18
5.1.8	-bc Set crop box	19
5.1.9	-c Center images	19
5.1.10	-cl Set conformance level	19
5.1.11	-cs Color space profile	20
5.1.12	-d Set resolution in DPI	20

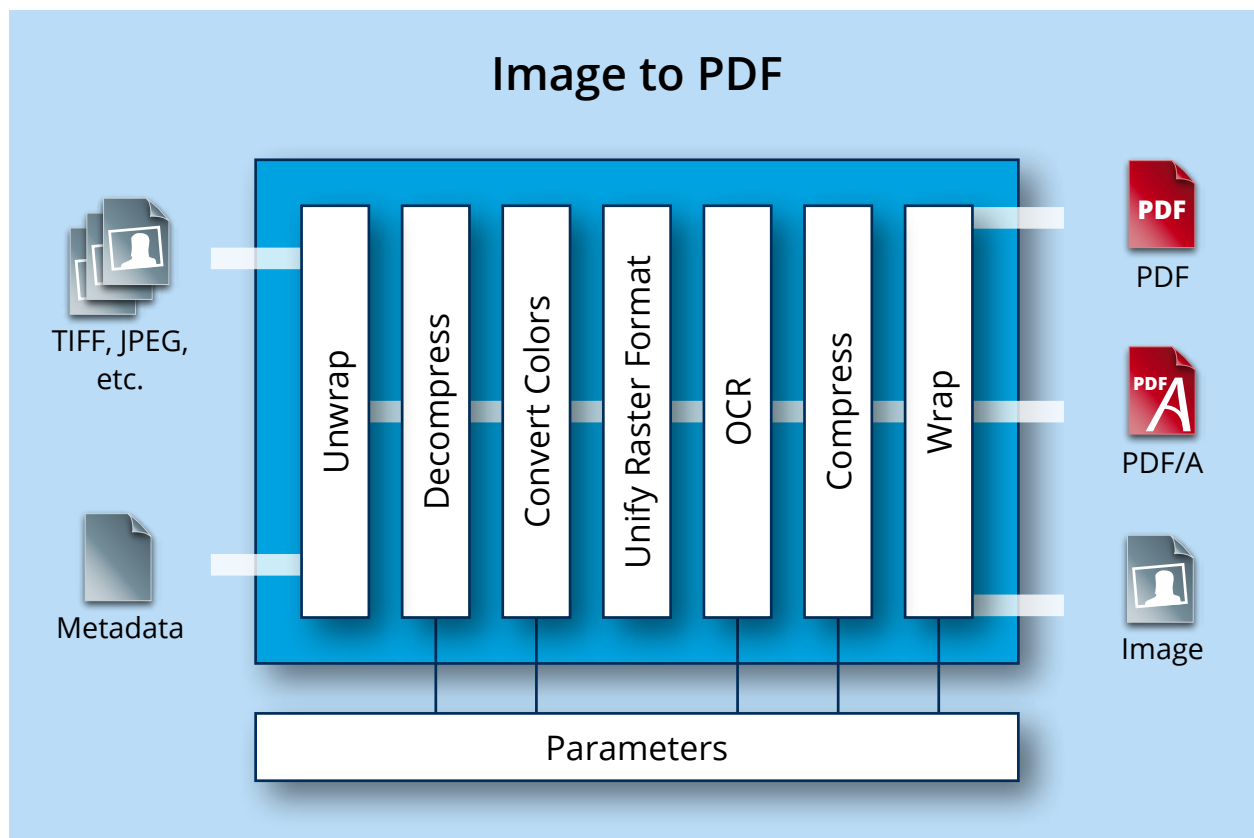
5.1.13	-f Fit the image size to the page size	20
5.1.14	-fb Bitonal image compression	21
5.1.15	-fc Color / Grayscale image compression	21
5.1.16	-h Dithering mode	21
5.1.17	-fi Indexed image compression	22
5.1.18	-fr Recompress JPEG streams	22
5.1.19	-fu Unpack indexed images	22
5.1.20	-i Set document information	22
5.1.21	-le List OCR engines	22
5.1.22	-lk Set license key	23
5.1.23	-o Owner password	23
5.1.24	-ocb Convert images to bitonal before OCR recognition	24
5.1.25	-ocbc Embed barcodes	24
5.1.26	-ocd Resolution for OCR recognition	24
5.1.27	-oci Do not deskew image	24
5.1.28	-ocl Set OCR language	24
5.1.29	-ocr Load OCR engine	25
5.1.30	-ocri Reembed pre-processed image	25
5.1.31	-ocp Set OCR parameters	25
5.1.32	-occs Correct skew angle	25
5.1.33	-ocs Do not reembed deskewed image	26
5.1.34	-oct Threshold resolution for OCR	26
5.1.35	-ocx Export recognized OCR text to file	27
5.1.36	-oi Set output intent	27
5.1.37	-or Set image orientation	28
5.1.38	-ow Optimize for the Web	28
5.1.39	-p Permission flags	29
5.1.40	-pg Page range	30
5.1.41	-q Set image quality	30
5.1.42	-rw Report decode errors	30
5.1.43	-sb Set border size	30
5.1.44	-sp Set page dimensions	31
5.1.45	-u User password	31
5.1.46	-v Verbose mode	31
5.2	Options for image to image	31
5.2.1	-cb Bitonal image compression	31
5.2.2	-cc Color / Grayscale image compression	32
5.2.3	-ci Indexed image compression	32
5.2.4	-d Set resolution	32
5.2.5	-dd Set default resolution	32
5.2.6	-fu Unpack indexed images	33
5.2.7	-hp Set height	33
5.2.8	-ht Set true height	33
5.2.9	-lk Set license key	33
5.2.10	-m Merge input to multi-page output	34
5.2.11	-pg Page range	34
5.2.12	-q Set image quality	34
5.2.13	-r Allow re-sampling	34
5.2.14	-s Split multi-page input to single-page output	35
5.2.15	-t Transform colors from CMYK to RGB	35
5.2.16	-wp Set width	35
5.2.17	-wt Set true width	35

5.3	Options for PdfOcr	36
5.3.1	-le List OCR engines	36
5.3.2	-lk Set license key	36
5.3.3	-o Owner password	36
5.3.4	-oca Rotate the image to the detected angle	37
5.3.5	-ocb Convert images to bitonal before OCR recognition	37
5.3.6	-ocbc Embed barcodes	37
5.3.7	-occs Correct skew angle	37
5.3.8	-ocd Resolution for OCR recognition	37
5.3.9	-oci Do not deskew image	38
5.3.10	-ocl Set OCR language	38
5.3.11	-ocm OCR mode	38
5.3.12	-ocp Set OCR parameters	38
5.3.13	-ocr Load OCR engine	39
5.3.14	-ocri Reembed pre-processed image	39
5.3.15	-ocs Do not reembed deskewed image	39
5.3.16	-oct Threshold resolution for OCR	40
5.3.17	-p Permission flags	40
5.3.18	-pw Read an encrypted PDF file	41
5.3.19	-u User password	42
5.3.20	-v Verbose mode	42
5.4	Supported image extensions	42
5.5	Compression types	43
5.6	Compression types: TIFF only	43
5.7	Specification of resolution and image dimensions	43
6	Return codes	45
6.1	Image to PDF Converter	45
6.2	Image to Image Converter	45
6.3	PdfOcr	46
7	Version history	47
7.1	Changes in versions 6.19–6.27	47
7.2	Changes in versions 6.13–6.18	47
7.3	Changes in versions 6.1–6.12	47
7.4	Changes in version 5	47
7.5	Changes in version 4.12	47
7.6	Changes in version 4.11	47
7.7	Changes in version 4.10	47
7.8	Changes in version 4.9	48
7.9	Changes in version 4.8	48
8	Licensing, copyright, and contact	49

1 Introduction

1.1 Description

The 3-Heights® Image to PDF Converter Shell converts raster image formats to PDF and PDF/A. PDF/A has been acknowledged world-wide as the ISO standard for long-term archiving since 2005. The Image to PDF Converter is used to convert images into a standardized format, for instance for electronic archiving or electronic data exchange. It is also possible to include metadata from external sources. The Converter is characterized by a robust design, high throughput and accurate image reproduction. The optional OCR add-in makes output files searchable in full text mode.



1.2 Functions

The 3-Heights® Image to PDF Converter Shell converts raster image formats such as JPEG, TIFF or PNG to PDF or PDF/A. It can merge pages from various image files to form a single PDF and can also split multi-page image files into single-page PDF files. Further options include defining page size and resolution, image scaling and the inclusion of (external) metadata. Optical character recognition (OCR) is also available as an option.

1.2.1 Features

Image to PDF

- Conversion of single-page or multi-page raster images to PDF
- Convert JPM to PDF/A-2 with MRC

- Set PDF conformance
- Automatic or selectable image compression, depending on the image type
- Automatic or selectable PDF page size
- Selectable page area
- Selectable image quality for lossy compression
- Set image position
- Set scaling
- Set standard resolution (DPI / X and Y coordinates)
- Set encryption and user access permissions
- Selectable and embeddable ICC color profile
- Define alternative text (tagging) and image language
- Set document attributes
- Optional JPEG image recompression
- Set image orientation (portrait or landscape)
- Optical character recognition (OCR)
- Embedding XMP metadata
- Support for image masks
- Support for mixed raster content (MRC)

Image to Image

- Split single-page or multi-page raster images into individual, single page images
- Merge multiple images to form one multi-page image
- Convert to an image format of the same color depth
- Modify TIFF image compression
- Set quality index for lossy image compression
- Create lossless JBIG2 images and lossy/lossless JPEG2000
- Set resolution and image dimensions

PdfOcr

- Recognition of machine generated text
- Recognition of typewriter scripts and barcodes (1D)
- Image manipulation
- Image pre-processing

1.2.2 Formats

Input formats

- BMP (1, 2, 4, 8, 24 bit)
- GIF (2 to 8 bit)
- HEIC/HEIF
- JBIG2 (lossless compression)
- JPEG, JPEG2000 and JPEG-LS (grayscale, RGB)
- JPM
- PBM and PNG (1 to 8, 24 bit)
- TIFF
 - Bitonal : uncompressed, CCITT G3, CCITT G3-2D, CCITT G4, LZW, ZIP, Packbits

- Grayscale, RGB and CMYK: uncompressed, LZW, JPEG, JPEG (old), ZIP, Packbits

Output formats - Image to PDF Converter

- PDF 1.x (PDF 1.0, ..., PDF 1.7)
- PDF 2.0
- PDF/A-1a, PDF/A-1b
- PDF/A-2a, PDF/A-2b, PDF/A-2u
- PDF/A-3a, PDF/A-3b, PDF/A-3u

Output formats - Image to Image Converter

- All input formats plus EPS

1.2.3 Conformance

- Standards:
 - ISO 32000-1 (PDF 1.7)
 - ISO 32000-2 (PDF 2.0)
 - ISO 19005-1 (PDF/A-1)
 - ISO 19005-2 (PDF/A-2)
 - ISO 19005-3 (PDF/A-3)
 - TIFF V6
- Quality assurance: Isartor test suite

1.3 Operating systems

The 3-Heights® Image to PDF Converter Shell is available for the following operating systems:

- Windows Client 7+ | x86 and x64
- Windows Server 2008, 2008 R2, 2012, 2012 R2, 2016, 2019, 2022 | x86 and x64
- Linux:
 - Red Hat, CentOS, Oracle Linux 7+ | x64
 - Fedora 29+ | x64
 - Debian 8+ | x64
 - Other: Linux kernel 2.6+, GCC toolset 4.8+ | x64
- macOS 10.10+ | x64

'+' indicates the minimum supported version.

2 Installation

2.1 Windows

The 3-Heights® Image to PDF Converter Shell comes as a ZIP archive or as an MSI installer.

To install the software, proceed as follows:

1. You need administrator rights to install this software.
2. Log in to your download account at <https://www.pdf-tools.com>. Select the product “Image to PDF Converter Shell”. If you have no active downloads available or cannot log in, please contact pdfsales@pdf-tools.com for assistance.

You can find different versions of the product available. Download the version that is selected by default. You can select a different version.

There is an MSI (*.msi) package and a ZIP (*.zip) archive available. The MSI (Microsoft Installer) package provides an installation routine that installs and uninstalls the product for you. The ZIP archive allows you to select and install everything manually.

There is a 32 and a 64-bit version of the product available. While the 32-bit version runs on both 32 and 64-bit platforms, the 64-bit version runs on 64-bit platforms only. The MSI installs the 64-bit version, whereas the ZIP archive contains both the 32-bit and the 64-bit version of the product. Therefore, on 32-bit systems, the ZIP archive must be used.

3. If you select an MSI package, start it and follow the steps in the installation routine.
4. If you are using the ZIP archive, unzip the archive to a local folder, e.g. C:\Program Files\PDF Tools AG\.

This creates the following subdirectories:

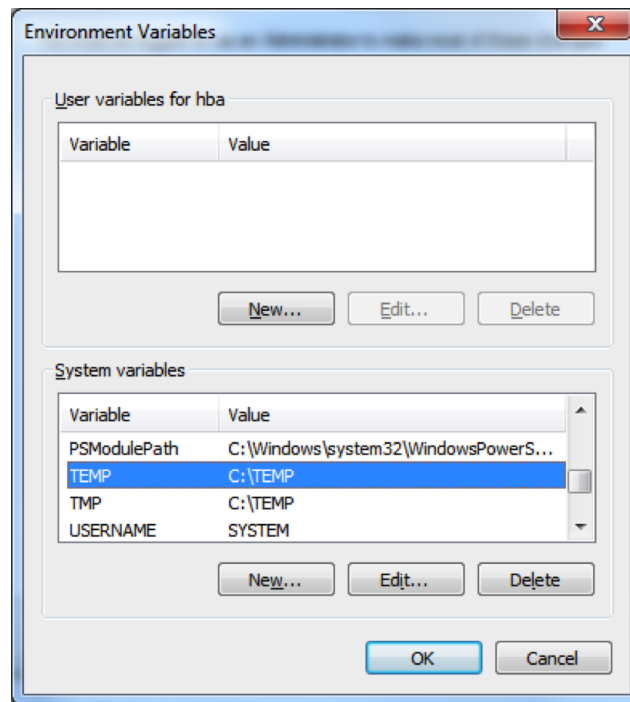
Subdirectory	Description
bin	Runtime executable binaries
doc	Documentation

5. (Optional) To easily use the 3-Heights® Image to PDF Converter Shell from a shell, the directory needs to be included in the “Path” environment variable.
6. (Optional) Register your license key using the [License management](#).
7. Ensure the cache directory exists as described in [Special directories](#).
8. Make sure your platform meets the requirements regarding fonts described in [Fonts](#).
9. (Optional) Download and install the 3-Heights® OCR Enterprise Add-on, and the OCR Engine as described in the respective manuals:
 - 3-Heights® OCR Add-on for ABBYY FineReader Engine v10: [OcrAbbyy10.pdf](#)
 - 3-Heights® OCR Add-on for ABBYY FineReader Engine v11: [OcrAbbyy11.pdf](#)
 - 3-Heights® OCR Add-on for ABBYY FineReader Engine v12: [OcrAbbyy12.pdf](#)
 - 3-Heights® OCR Service: [OcrService.pdf](#) from the separate product kit.

2.1.1 How to set the environment variable “Path”

To set the environment variable “Path” in Windows, go to Start → Control Panel (classic view) → System → Advanced → Environment Variables.

Select “Path” and “Edit”, then add the directory where `img2pdf.exe` is located to the “Path” variable. If the environment variable “Path” does not exist, create it.



2.2 Linux and macOS

This section describes installation steps required on Linux or macOS.

Here is an overview of the files that come with the 3-Heights® Image to PDF Converter Shell:

File description

Name	Description
bin/x64/img2pdf	Main executable
bin/x64/*.ocr	OCR plugin modules
doc/*.*	Documentation

2.2.1 Linux

1. Unpack the archive in an installation directory, e.g. `/opt/pdf-tools.com/`
2. Verify that the GNU shared libraries required by the product are available on your system:

```
ldd img2pdf
```

If the previous step reports any missing libraries, you have two options:

- a. Download an archive that is linked to a different version of the GNU shared libraries and verify whether they are available on your system. Use any version whose requirements are met. Note that this option is not available for all platforms.
 - b. Use your system's package manager to install the missing libraries. It usually suffices to install the package `libstdc++6`.
3. Create a link to the executable from one of the standard executable directories, e.g.

```
ln -s /opt/pdf-tools.com/bin/x64/img2pdf /usr/bin
```

4. Optionally, register your license key using the [license manager](#).
5. Ensure the cache directory exists as described in [Special directories](#).
6. Make sure your platform meets the requirements regarding fonts described in [Fonts](#).
7. (Optional) Download and install the 3-Heights® OCR Enterprise Add-on, and the OCR Engine as described in the respective manuals:
 - 3-Heights® OCR Add-on for ABBYY FineReader Engine v10: [OcrAbbyy10.pdf](#)
 - 3-Heights® OCR Add-on for ABBYY FineReader Engine v11: [OcrAbbyy11.pdf](#)
 - 3-Heights® OCR Add-on for ABBYY FineReader Engine v12: [OcrAbbyy12.pdf](#)
 - 3-Heights® OCR Service: [OcrService.pdf](#) from the separate product kit.

2.3 Uninstall

If you have used the MSI for the installation, go to Start → 3-Heights® Image to PDF Converter Shell... → Uninstall ...

If you have used the ZIP file for the installation, undo all the steps done during installation.

2.4 Color profiles

In PDF/A, using uncalibrated color spaces (DeviceGray, DeviceRGB, and DeviceCMYK) is prohibited because colors that are specified in this way cannot be reproduced reliably on multiple output devices. Therefore, when converting to PDF/A, a color profile has to be embedded.

If no color profiles are available, default profiles for both RGB and CMYK are generated on the fly by the 3-Heights® Image to PDF Converter Shell.

2.4.1 Default color profiles

If no particular color profiles are set, default profiles are used. For device RGB colors, a color profile named "sRGB Color Space Profile.icm" and for device CMYK, a profile named "USWebCoatedSWOP.icc" are searched for in the following directories:

Windows

1. %SystemRoot%\System32\spool\drivers\color
2. directory `Icc`, which must be a direct subdirectory of where the `img2pdf.exe` resides.

Linux and macOS

1. `$PDF_ICC_PATH` if the environment variable is defined
2. the current working directory

2.4.2 Set other color profiles

Another color profile may be set using the [-oi](#) or [-cs](#) options.

2.4.3 Get other color profiles

Most systems have pre-installed color profiles available. For example, on Windows at %SystemRoot%\system32\spool\drivers\color\. Color profiles can also be downloaded from the links provided in the directory bin\Icc\ or from the following websites:

- <https://www.pdf-tools.com/public/downloads/resources/colorprofiles.zip>
- <https://www.color.org/srgbprofiles.html>

2.5 Fonts

Fonts are required, if OCR is preformed and OCR text is added to a PDF document. Therefore, it is crucial, that the fonts available in the [Font directories](#) contain all characters required for the OCR text. For example, when recognizing Japanese OCR text, it is recommended to add the fonts “MS Mincho” or “MS Gothic” to the [Font directories](#).

Note that on Windows, when a font is installed, it is by default installed only for a particular user. It is important to either install fonts for all users, or make sure the 3-Heights® Image to PDF Converter Shell is run under that user and the user profile is loaded.

2.5.1 Font cache

A cache of all fonts in all [Font directories](#) is created. If fonts are added or removed from the font directories, the cache is updated automatically.

In order to achieve optimal performance, make sure that the cache directory is writable for the 3-Heights® Image to PDF Converter Shell. Otherwise, the font cache cannot be updated and the font directories have to be scanned on each program startup.

The font cache is created in the subdirectory <CacheDirectory>/Installed Fonts of the [Cache directory](#).

2.6 Note about the evaluation license

With the evaluation license, the 3-Heights® Image to PDF Converter Shell automatically adds a watermark to the output files.

2.7 Special directories

2.7.1 Directory for temporary files

This directory for temporary files is used for data specific to one instance of a program. The data is not shared between different invocations and is deleted after termination of the program.

The directory is determined as follows. The product checks for the existence of environment variables in the following order and uses the first path found:

Windows

1. The path specified by the %TMP% environment variable
2. The path specified by the %TEMP% environment variable
3. The path specified by the %USERPROFILE% environment variable
4. The Windows directory

Linux and macOS

1. The path specified by the \$PDFTMPDIR environment variable
2. The path specified by the \$TMP environment variable
3. The /tmp directory

2.7.2 Cache directory

The cache directory is used for data that is persistent and shared between different invocations of a program. The actual caches are created in subdirectories. The content of this directory can safely be deleted to clean all caches.

This directory should be writable by the application; otherwise, caches cannot be created or updated and performance degrades significantly.

Windows

- If the user has a profile:
%LOCAL_APPDATA%\PDF Tools AG\Caches
- If the user has no profile:
<TempDirectory>\PDF Tools AG\Caches

Linux and macOS

- If the user has a home directory:
~/.pdf-tools/Caches
- If the user has no home directory:
<TempDirectory>/pdf-tools/Caches

where <TempDirectory> refers to the [Directory for temporary files](#).

2.7.3 Font directories

The location of the font directories depends on the operating system. Font directories are traversed recursively in the order as specified below.

If two fonts with the same name are found, the latter one takes precedence, i.e. user fonts always take precedence over system fonts.

Windows

1. %SystemRoot%\Fonts
2. User fonts listed in the registry key \HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Fonts. This includes user specific fonts from C:\Users\<user>\AppData\Local\Microsoft\Windows\Fonts and app specific fonts from C:\Program Files\WindowsApps
3. Fonts directory, which must be a direct subdirectory of where img2pdf.exe resides.

macOS

1. /System/Library/Fonts
2. /Library/Fonts

Linux

1. /usr/share/fonts
2. /usr/local/share/fonts

3. ~/.fonts
4. \$PDFFONTPATH or /usr/lib/X11/fonts/Type1

3 License management

The 3-Heights® Image to PDF Converter Shell requires a valid license in order to run correctly. If no license key is set or the license is not valid, then the executable will fail and the return code is set to 10.

More information about license management is available in the [license key technote](#).

4 Getting started

3-Heights® Image to PDF Converter Shell provides three separate programs:

img2pdf this executable is for converting most raster images files, such as TIFF, JPEG, PNG, BMP, GIF, etc. to a PDF or PDF/A document. The majority of this documentation is dedicated to this executable.

img2img this executable is for converting a raster image from one format to another raster image format, e.g. from BMP to PNG.

jpm2pdf this executable is for converting a JPM image (JPEG 2000, Part 6) to a PDF/A-2 document with Mixed Raster Content (MRC).

pdfocr this executable is used in combination with an optical character recognition (OCR) engine to make PDF documents searchable by performing OCR on embedded images.

All features of the tools are listed in their usage. The usage can be retrieved by typing the name of the executable, such as `img2pdf`, without parameters.

The simplest command requires an input image file parameter and an output PDF file as parameters:

```
img2pdf input.tif output.pdf
```

It converts an image type file to a PDF document. If the image is multi-page TIFF image, then each page in the image is converted to a page in the PDF output document.

To convert and concatenate several image files into one PDF document, add the input files as additional parameters before the output file:

```
img2pdf input1.tif input2.jpg input3.gif output.pdf
```

4.1 Specify the folder of the output file

The output folder can simply be added in front of the output file name.

```
img2pdf input.pdf myfolder\output.pdf
```

or absolute (Windows):

```
img2pdf input.pdf C:\myfolder\output.pdf
```

4.2 Using wildcards (*.*)

The 3-Heights® Image to PDF Converter Shell supports wildcards. If a directory, for example, contains the following input JPEG files:

A01.jpg
A02.jpg
A03.jpg
B01.jpg
B02.jpg

Then the following command processes all JPEG files starting with the letter "A".

```
img2pdf A*.jpg output.pdf
```

Note: The file extension of the input files must always be a supported format. When using wildcards, it is helpful to set the verbose mode option, [-v](#). The command then looks like this:

```
img2pdf -v *.jpg output.pdf
```

And the generated output message looks like this:

Wildcards return a list of existing files. If you would like to convert all files in a directory to individual output files, use a variable to name the output files.

Example: Use the for command of the Windows CMD shell, to convert all JPEG files to individual PDF files with the same name and the extension `.pdf`, in the same directory:

```
for %f in (*.jpg) do img2pdf -v %f %~nf.pdf
```

Example: You can adjust the paths or use a different output name:

```
for %f in (C:\InputDir\*.jpg) do img2pdf -v %f C:\OutputDir\%~nf.pdf
```

Note: Variables used in a batch file (`.bat`) require two leading % instead of one.

4.3 Converting images to images

To convert an image file to an image file with another format, you just need to enter the image to image executable, the file name, and the desired output image file name, with the correct extension:

```
img2img input.tif output.jpg
```

Note: that the image-to-image converter cannot change the color depth.

4.4 JPM to PDF/A

Use the `jpm2pdf` executable to convert a JPEG 2000 Part 6 to a PDF/A-2 document.

```
jpm2pdf input.jpm output.pdf
```


The tool can create optional content groups that can be used by the reader of the PDF document to switch off and on the various MRC layers.

```
jpm2pdf -ocg input.jpm output.pdf
```

4.5 PdfOcr

After the PDF OCR Import Shell and the 3-Heights® OCR Enterprise Add-on are installed, you can list the available OCR add-ons to retrieve the name of the OCR engine using the `pdfocr -le` command as shown below:

Example:

```
pdfocr -le
List of available OCR engines:
- abbyy10
- abbyy11
- service
End of list.
```

The list includes only the entries of the available OCR add-ons. The entries in the list indicate which of the add-ons `pdfocrpluginService.ocr`, `pdfocrpluginAbbyy10.ocr`, and `pdfocrpluginAbbyy11.ocr` are found. The add-ons are required to communicate with the actual OCR engine or service. Being able to list the add-ons does not necessarily mean the OCR engine is installed and ready. Installation of the OCR engine is described in the documentation `PdfOcrEnterprise.pdf`.

Once the name (e.g. “abbyy11”) is known, it is provided as argument to the `-ocr` switch. The command following example is the basic command to apply OCR to a document, i.e. the input document `input.pdf` is read, OCR is applied, and the resulting, OCR-processed document is saved as `output.pdf`.

Example: Set the OCR engine to the “Abbyy FineReader 11 OCR Engine”

```
pdfocr -ocr abbyy11 input.pdf output.pdf
```

Additional OCR engine dependant settings or settings related to encryption are described in [Interface reference](#).

5 Interface reference

5.1 Options for image to PDF

5.1.1 @<filename> Use a control file

Use a control file @<filename>

Read the parameters from a control file. This is particularly useful for long commands, as usually shells have a limited length of a command. Each option and file name is to be written on a new line in the control file. Use double quotes around strings containing blanks.

Example: Control file control.txt

```
-q 80
-i "Title=My Title"
"C:\Some Path\input.tif"
output.pdf
```

Example: A command using a control file:

```
img2pdf @control.txt
```

5.1.2 -a Adjust the page size to the size of the image

Adjust the page size to the size of the image -a

Adjust the pages of the PDF document to the size of the image. This disables [-c](#) and [-f](#).

5.1.3 -aa Set alternate text

Set alternate text -aa <alt>

To create a document that conforms to PDF/A-2a, an image must have an alternate text. The **-aa** option sets this alternate text. This option is only relevant in combination with PDF/A-2a. The default text is "Imported image".

Example: Set the conformance to PDF/A-2a and set the alternative text for the image to "some text"

```
img2pdf -cl pdfa-2a -aa "some text" input.tif output.pdf
```

5.1.4 -al Set language for alternate text

Set language for alternate text -al <lang>

Set the language for the alternate text that is set using the [-aa](#) option.

The default language is US - EN. Other languages can be set using the corresponding abbreviations, e.g. DE (German), FR (French), etc.

Example: Set the conformance to PDF/A-2a, set the alternative text to "Beschreibung" and the language to German

```
img2pdf -c1 pdfa-2a -aa "Beschreibung" -al DE input.tif output.pdf
```

5.1.5 -ao Adjust page orientation

Adjust page orientation -ao

When set, every page of the PDF is oriented in such a way that the longer side length of the input image conforms with the longer side length of the corresponding page.

Example: Fit input TIFF (multipage) to PDF with A5 pages. Adjust orientation of every PDF page to corresponding input TIFF page

```
img2pdf -ao -f -sp 420 595 input.tif output.pdf
```

5.1.6 -ax Add XMP metadata

Add XMP metadata -ax <file>

Specify a file with XMP metadata, which are added to the output document. The XMP metadata are copied only and not checked for PDF conformance.

5.1.7 -b Set bits per pixel

Set bits per pixel -b <n>

Set the color depth. Available: Bitonal: **1**. When using 1 bit per pixel, it is suggested to set a suitable dithering algorithm ([-h](#) option).

Example: Convert to bitonal and use LZW compression

```
img2pdf -b 1 -fb 3 input.jpg output.pdf
```

5.1.8 -bc Set crop box

Set crop box -bc <x> <y> <w> <h>

Set the crop box. It takes four parameters: x-position, y-position, width and height. All values are in PDF points (A4 = 595 x 842 points).

The crop box is a rectangle, defining the visible region of the page. When the page is displayed or printed, its contents are to be clipped (cropped) to this rectangle and then imposed on the output medium in some implementation-defined manner.

Example: The following command creates an image with a crop box that is positioned 50 points from the left border, 100 points from the bottom, is 150 points wide and 200 points high

```
img2pdf -bc 50 100 150 200 input.tif output.pdf
```

If no crop box is set, the crop box is equal to the media box.

5.1.9 -c Center images

Center images -c

Center the images on the pages horizontally and vertically. This disables [-a](#).

5.1.10 -cl Set conformance level

Set conformance level -cl <level>

Set the PDF conformance level. Supported conformance levels are:

pdf1.x Regular PDF versions 1.0, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7

pdf2.0 Regular PDF version 2.0

pdfa-1b PDF/A-1b format

pdfa-1a PDF/A 1a format (accessibility)

pdfa-2b PDF/A 2b format

pdfa-2u PDF/A 2u format (Unicode)

pdfa-2a PDF/A 2a format (accessibility)

pdfa-3b PDF/A 3b format

pdfa-3u PDF/A 3u format (Unicode)

pdfa-3a PDF/A 3a format (accessibility)

The default is pdf1.7.

Example: To create a document that conforms to PDF/A-1b, use a setting like this:

```
img2pdf -cl pdfa-2b input.tif output.pdf
```

Note: To create PDF/A compatible documents, it may be required to provide a color profile. The color profile is then embedded in the PDF/A document. (See [-cs](#) and [-oi](#) switches)

Selecting a PDF/A conformance level automatically generates the XML metadata and other requirements to meet the PDF/A specification. If JPEG2000 images are to be converted to PDF/A and the JPEG2000 compression is to be retained, a PDF/A-2 or PDF/A-3 conformance level must be selected.

5.1.11 -cs Color space profile

Color space profile -cs

Set a color profile for embedding in the output PDF. The color profile provided here is used directly for the image's color space in the output PDF. See also [-oi](#) for setting the PDF's output intent.

At maximum three profiles (one RGB profile, one CMYK profile, and one Gray profile) can be set by using at most one [-oi](#) switch and/or at most three [-cs](#) switches.

Example: Set a color profile for all RGB images in the PDF.

```
img2pdf -cs "C:\Windows\system32\spool\drivers\color\sRGB Color ^  
Space Profile.icm" input.jpg output.pdf
```

5.1.12 -d Set resolution in DPI

Set resolution in DPI -d <dpi>

Set the default resolution in dots per inch (DPI) if not provided from the image. The default is 96. If the resolution is given by the image, then this option does not have any effect. Basically, the [-d](#) switch changes the amount of dots per inch by changing the size of the image in the PDF document. The size of the raster image in pixel is not changed.

Example: Set the resolution to 150 DPI.

```
img2pdf -d 150 input.tif output.pdf
```

5.1.13 -f Fit the image size to the page size

Fit the image size to the page size -f

Scale the image to fit on the page dimensions. This disables [-a](#).

5.1.14 -fb Bitonal image compression

Bitonal image compression -fb <compr>

Set the bitonal image compression. Default is 6 (CCITT Fax Group 4). See [Compression types](#) for possible codecs.

Example: Set the compression for bitonal images to CCITT Fax Group 3

```
img2pdf -fb 4 input.tif output.pdf
```

5.1.15 -fc Color / Grayscale image compression

Color / Grayscale image compression -fc <compr>

Set the color / gray image compression. Default is 1 (JPEG). See [Compression types](#) for possible codecs.

Example: Set the compression for color images to JPEG2000

```
img2pdf -fc 8 input.jpg output.pdf
```

5.1.16 -h Dithering mode

Dithering mode -h <mode>

Set the dithering mode. Allowed values are:

Dithering mode

Value	Description
0	No dithering
1	(Default) Floyd-Steinberg
2	Halftone block
3	Halftone continuous
4	G3 Optimized
5	G4 Optimized
6	Atkinson dithering is very fast and produces images that can be compressed really well with reasonably good image quality.

Dithering provides a better image quality, especially for 1 bit images, at the cost of a larger file size.

Example: Convert image to bitonal, use dithering mode halftone continuous and use jbig2 compression

```
img2pdf -h 3 -b 1 -fb 7 input.jpg output.tif
```

5.1.17 -fi Indexed image compression

Indexed image compression -fi <compr>

Set the indexed image compression. Default is 2 (Flate). See [Compression types](#) for possible codecs.

Example: Set the compression for indexed images to LZW

```
img2pdf -fi 3 input.tif output.pdf
```

5.1.18 -fr Recompress JPEG streams

Recompress JPEG streams -fr

Re-compress JPEG streams. This is useful for JPEG streams that cannot be read by certain (older) PDF viewing applications.

5.1.19 -fu Unpack indexed images

Unpack indexed images -fu

This switch instructs the converter to unpack images with bits per sample of less than 8 to exactly 8 bits.

5.1.20 -i Set document information

Set document information -i <key=value>

Set document properties such as "Title", "Author", "Subject", and "Keywords".

Example: Set the title and the author entries of the document properties

```
img2pdf -i Title="My Title" -i Author="My Name" input.tif output.pdf
```

5.1.21 -le List OCR engines

List OCR engines -le

OCR engines are accessed through the corresponding OCR interface DLLs. The following engines are supported:

Abbyy FineReader 11 OCR engine This engine is accessed by the OCR interface DLL `pdfocrpluginAbbyy11.ocr`.

Abbyy FineReader 10 OCR engine This engine is accessed by the OCR interface DLL `pdfocrpluginAbbyy10.ocr`.

3-Heights® OCR service This service is accessed by the OCR interface DLL `pdfocrpluginService.ocr`. The service accesses the Abbyy FineReader 10 or 11 OCR Engine.

The OCR interface DLLs are provided by the 3-Heights® Image to PDF Converter Shell. The OCR engine is provided as a separate product, such as 3-Heights® OCR Enterprise Add-on.

Here is an example of listing available OCR engines:

```
img2pdf -le
List of available OCR engines:
- abbyy11
- abbyy10
- service
End of list.
```

In order to make use of the OCR engine, the OCR interface DLL and the OCR engine must be installed. The `-le` switch lists all available OCR interface DLLs. It does not verify the corresponding OCR engine is installed and can be initialized. The OCR engine is actually accessed when using the switch `-ocr`.

5.1.22 -lk Set license key

Set license key `-lk <key>`

Pass a license key to the application at runtime, instead of using one that is installed on the system.

```
img2pdf -lk X-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX ...
```

This is required in an OEM scenario only.

5.1.23 -o Owner password

Owner password `-o <owner>`

The owner password is required to change the security settings of the document. To apply permission flags, an owner password must be set. Permission flags are set with the `-p` switch.

Example: Encrypt a document and set the owner password to `<owner>`.

```
img2pdf -o owner input.pdf output.pdf
```


5.1.24 -ocb Convert images to bitonal before OCR recognition

Convert images to bitonal before OCR recognition -ocb

Specify whether the images should be converted to bitonal (black and white) before OCR recognition.

Enabling this feature can improve the memory consumption of the OCR process.

5.1.25 -ocbc Embed barcodes

Embed barcodes -ocbc

Embed the recognized barcodes in the XMP metadata.

5.1.26 -ocd Resolution for OCR recognition

Resolution for OCR recognition -ocd <n>

Re-sample images to target resolution before they are sent to the OCR engine. If no value is set, images are re-sampled to 300 DPI for OCR, which is the preferred resolution for most OCR engines.

5.1.27 -oci Do not deskew image

Do not deskew image -oci

Do not deskew original image (with [-ocs](#) only). This option specifies whether the image and text are deskewed according to the recognized skew angle.

- With option [-oci](#):
Do not change skew of images (i.e. do not change appearance of the page). This setting is recommended for born-digital documents.
- Without option [-oci](#):
Rotate image, such that lines of text are made horizontal. This might change the appearance of the page. This setting is recommended for scanned documents.

5.1.28 -ocl Set OCR language

Set OCR language -ocl <languages>

To optimize the performance of the OCR engine, it can be given hints what languages are used. The default language of the Abbyy FineReader 11 OCR Engine is English. This switch can only be used if the switch [-ocr](#) is set. This setting depends on the OCR engine.

The following switch set the languages to English and German:

```
img2pdf -ocr abbyy11 -ocl "English, German" input.tif output.pdf
```

See also documentation for the 3-Heights® OCR Add-on.

5.1.29 -ocr Load OCR engine

Load OCR engine -ocr <name>

If a PDF document has to be made fully text searchable, even if the text is part of a raster image, then the images that are contained in the PDF document must be run through an OCR engine. With this switch, the user can select an OCR engine, e.g. Abbyy11, and instruct the tool to embed the recognized text as a hidden layer on top of the image. If the add-in is not found or the engine cannot be initialized (because it is not installed or the license key is not valid), then an error message is issued.

The name of the OCR engine can be retrieved using the switch -le. If the switch -ocr is not used, no OCR is applied.

Example: The following switch sets the OCR engine to the OCR Service

```
img2pdf -ocr service input.tif output.pdf
```

See also documentation for the 3-Heights® OCR Add-on.

5.1.30 -ocri Reembed pre-processed image

Reembed pre-processed image -ocri

This option currently requires the [-occs](#) to be set.

The OCR engine deskews and denoises the input image before recognizing the characters. This option controls whether the 3-Heights® Image to PDF Converter Shell should use the pre-processed image or keep the original image.

This option has only an effect, if the pre-processed image is provided by the OCR engine, which depends on the type and settings of the engine.

If this option is set, the resulting image may have a different color space, compression and size.

Since this option currently requires [-occs](#), it is recommended only for simple scanned documents.

5.1.31 -ocp Set OCR parameters

Set OCR parameters -ocp <params>

Using this switch, OCR engine specific parameters (key/value pairs) can be set to optimize the performance.

The following switch sets a predefined profile (i.e. a configuration setting), which is optimized for creating electronic archives with high accuracy:

```
img2pdf -ocr abbyy11 -ocp "PredefinedProfile = DocumentArchiving_Accuracy"
input.tif output.pdf
```

See also documentation for the 3-Heights® OCR Add-on.

5.1.32 -occs Correct skew angle

Correct skew angle -occs

Correct the skew angle of images.

This option has only an effect if the required information is provided by the OCR engine, which depends on the type and settings of the engine.

This option may change the appearance of the page and is only recommended for simple scanned documents that consist of a single image.

Using the option for digital-born documents may destroy the page layout.

5.1.33 -ocs Do not reembed deskewed image

Do not reembed deskewed image -ocs

The OCR engine deskews and denoises the input image before recognizing the characters. This option controls whether the 3-Heights® Image to PDF Converter Shell should use the deskewed image or keep the original image.

- With option -ocs:
Embed the original image (also see option [-oci](#)). This setting is recommended for born-digital documents.
- Without option -ocs:
Embed the deskewed and denoised image from the OCR engine. This might change the appearance of the page. This setting is recommended for scanned documents.

5.1.34 -oct Threshold resolution for OCR

Threshold resolution for OCR -oct <n>

Only images with a higher resolution than the threshold are re-sampled before OCR. The default is 400 DPI. If set to -1, no re-sampling is applied.

Example: Re-sample all images with a resolution of more than 300 DPI to 300 DPI:

```
img2pdf -ocd 300 -oct 1 input.tif output.pdf
```

Example: Re-sample all images with a resolution of 400 DPI or more to 300 DPI (default):

```
img2pdf -ocd 300 -oct 400 input.tif output.pdf
```

Example: Do not re-sample:

```
img2pdf -oct -1 input.tif output.pdf
```

Compatibility note: Initially, this switch was called -ocD and then renamed to -oct to avoid confusions with the switch -ocd.

5.1.35 -ocx Export recognized OCR text to file

```
Export recognized OCR text to file -ocx <file>
```

Export the retrieved OCR text to a file. This function can only be used in combination with an OCR engine (see [-ocr](#)). When an OCR engine is set, the OCR text is always embedded in the resulting PDF document. If this method is used, it is also extracted to a file.

The output format is a table, where rows are separated by a new line and columns are separated by a tabulator. The table contains the following columns:

Output column	Description
Page	Page number
Image	PDF object number which contains the image
FontSize	Font size in points
FontName	Font name, for any barcode font the name is Barcode. This value is only set if the font name is returned by the OCR engine.
FontFamily	<div>1 Serif</div> <div>2 SansSerif</div> <div>3 Monospaced</div> <div>This value is only set if provided by the OCR engine</div>
FontStyles	<div>2 Bold</div> <div>4 Italic</div> <div>8 Underline</div> <div>16 Strikeout</div> <div>This value is only set if provided by the OCR engine</div> <div>Example: 6 = 2 + 4 = Bold + Italic</div>
Baseline	Baseline of the text
Left, Top, Right, Bottom	Bounding box of the text in PDF coordinates
String	Recognized text

Example: Write extracted text to the file `text.txt`.

```
img2pdf -ocr abbyy11 -ocx text.txt input.tif output.pdf
```

5.1.36 -oi Set output intent

```
Set output intent -oi <profile>
```

The output intent holds the output color profile. Setting the output intent is generally only recommended for images intended for a particular CMYK output device. In other cases, the option `-cs` should be used.

Color profiles are usually provided with the OS. On Windows for example they can be found at `C:\Windows\System32\spool\drivers\color`.

Alternatively profiles can be found here:

- <https://www.pdf-tools.com/public/downloads/resources/colorprofiles.zip>
- <https://www.color.org/srgbprofiles.html>
- https://www.adobe.com/support/downloads/iccprofiles/iccprofiles_win.html

Note: Most color profiles are copyrighted, therefore you should read the license agreements on the above links before using the color profiles.

Example: Set the output intent to a specific profile that exists on the system.

```
img2pdf -oi "C:\Windows\system32\spool\drivers\color\USWebCoatedSWOP.icc" input.tif output.pdf
```

5.1.37 -or Set image orientation

Set image orientation `-or <n>`

Set the orientation of the image. Available orientations are:

Value	Description	Transformation
0 (Default)	inherent	Undefined
1	top-left	Untransformed
2	top-right	Horizontal flip
3	bottom-right	Rotation by 180°
4	bottom-left	Vertical flip
5	left-top	Rotation by 90° clockwise followed by horizontal flip
6	right-top	Rotation by 90° clockwise
7	right-bottom	Rotation by 90° clockwise followed by vertical flip
8	left-bottom	Rotation by 90° counter-clockwise

5.1.38 -ow Optimize for the Web

Optimize for the Web `-ow`

Add linearization tags to the document. A linearized document has a slightly larger file size than a non-linearized file, and provides the following features (among others):

- When a document is opened through a PDF viewing application plugin for an Internet browser, the first page can be viewed without downloading the entire PDF file.
- When another page is requested by the user, the page is displayed as quickly as possible and incrementally as data arrives, without downloading the entire PDF file.

5.1.39 -p Permission flags

Permission flags -p <flags>

This option sets the permission flags. It is only usable when producing encrypted documents. In other words, at least an owner password must be set with [-o](#), and additionally a user password can be set with [-u](#). When omitting the [-p](#) option, then all permissions are granted. The permissions that can be granted are listed below.

Permission flags

Value	Description
0	Allow nothing (no permissions are granted)
4	Allow printing (low resolution)
8	Allow change of the document
16	Allow content copying or extraction
32	allow annotations
256	Allow filling of form fields
512	Support disabilities
1024	Allow document assembly
2048	Allow high quality printing

Any combination of permissions can be granted by adding up their values.

Example: The following command sets the owner password to “owner” and the permission flags to allow “printing in low resolution” 4 and “allow form filling” 256.

```
img2pdf -o owner -p 260 input.tif output.pdf
```

Example: “High quality printing” 2048 requires the standard printing flag 4 to be set too.

```
img2pdf -o owner -p 2052 input.tif output.pdf
```

For further information about the permission flags, see [PDF Reference 1.7](#) Section 3.5.2.

5.1.40 -pg Page range

Page range -pg <first> <last>

If not all pages should be printed, the page range can be defined using the **-pg** switch. Use 1 for the first page. Use -1 for the last page.

Example: Convert the first page only

```
img2pdf -pg 1 1 input.tif output.pdf
```

Example: Convert all but the first page

```
img2pdf -pg 2 -1 input.tif output.pdf
```

5.1.41 -q Set image quality

Set image quality -q <quality>

Some image compression algorithms such as JPEG or JPEG2000 support lossy compression. The quality index can be controlled using the **-q** option. The lowest quality index is 1; the highest is 100. The default value is 80. If the quality is set to 100, JPEG2000 images are compressed lossless.

Example: Set the image quality to 100, use JPEG2000 with lossless compression for color images:

```
img2pdf -fc 8 -q 100 input.jpg output.pdf
```

5.1.42 -rw Report decode errors

Report decode errors -rw

Return error code if decode errors occur.

5.1.43 -sb Set border size

Set border size -sb <n>

Define the width of a white border around the image in pages of the PDF document. The units are points (1 point = 1/72 inch). The default is 0 points. The border does not increase the dimensions of the page set by the **-sp** option.

Example: Set the width of the border to 20 points

```
img2pdf -sb 20 input.tif output.pdf
```

5.1.44 -sp Set page dimensions

Set page dimensions -sp <w> <h>

Set the dimensions of the pages of the PDF document in points (1 point = 1/72 inch). The default is A4 (595 x 842 points). It disables the -a option.

Example: Create "Letter" sized PDF pages

```
img2pdf -sp 612 792 input.tif output.pdf
```

5.1.45 -u User password

User password -u <user>

Set the user password of the document. If a document that has a user password is opened for any purpose (such as viewing, printing, editing), either the user or the owner password must be provided.

A user who knows the user password is able to open and read the document. A user who knows the owner password is able to open, read, and modify (e.g. change passwords) the document. A PDF document can have none, either, or both passwords.

Example: Encrypt a document with a user and an owner password

```
img2pdf -u userpassword -o ownerpassword input.tif output.pdf
```

5.1.46 -v Verbose mode

Verbose mode -v

This option turns on the verbose mode.

In the verbose mode, additional information during the processing is written to the shell.

5.2 Options for image to image

5.2.1 -cb Bitonal image compression

Bitonal image compression -cb <compr>

Set the bitonal image compression. Default is 6 (CCITT Fax Group 4). See [Compression types: TIFF only](#) for possible codecs.

Example: Set the compression for bitonal images to CCITT Fax Group 3.

```
img2img -cb 4 input.tif output.tif
```

5.2.2 -cc Color / Grayscale image compression

Color / Grayscale image compression -cc <compr>

Set the color / gray image compression. Default is 9 (TIFFJPEG). See [Compression types: TIFF only](#) for possible codecs.

Example: Set the compression for color images to JPEG

```
img2img -cc 1 input.jpg output.tif
```

5.2.3 -ci Indexed image compression

Indexed image compression -ci <compr>

Set the indexed image compression. Default is 3 (LZW). See [Compression types: TIFF only](#) for possible codecs.

Example: Set the compression for indexed images to Flate

```
img2img -ci 2 input.tif output.tif
```

5.2.4 -d Set resolution

Set resolution -d <dpi>

Set the resolution of the output image in dots per inch (DPI). The width and the height remain constant. Furthermore, if the [-r](#) option is set, true_width and true_height are constant, whereas the width and the height are changed. This latter case results in a re-sampling of the image. For more information about the behavior, see [Specification of resolution and image dimensions](#).

Example: Set the resolution of output file to 300 DPI and let true_width and true_height be constant

```
img2img -d 300 -r input.tif output.tif
```

5.2.5 -dd Set default resolution

Set default resolution -dd <dpi>

Set the default resolution of the input image in dots per inch (DPI). It is only effective where the input image has no resolution stated. Default 96 DPI.

Example: Set the default resolution of input file to 72 DPI

```
img2img -dd 72 input.tif output.bmp
```

5.2.6 -fu Unpack indexed images

Unpack indexed images -fu

This switch instructs the converter to unpack images with bits per sample of less than 8 to exactly 8 bits.

5.2.7 -hp Set height

Set height -hp <height>

Set the height in pixels of the output image. The width is calculated respecting proportions. If the width is set too (see [-wp](#)), the height is omitted. For more information about the behavior, see [Specification of resolution and image dimensions](#).

Example: Set height of output file to 1800 pixels

```
img2img -hp 1800 input.tif output.bmp
```

5.2.8 -ht Set true height

Set true height -ht <true_height>

Set the true_height in mm of output image. The true_width is calculated respecting proportions. If the true_width is set too (see [-wt](#)), the true_height is omitted. For more information about the behavior, see [Specification of resolution and image dimensions](#).

Example: Set true_height of output file to 500 mm and let the resolution be constant

```
img2img -ht 500 -r input.bmp output.jpg
```

5.2.9 -lk Set license key

Set license key -lk <key>

Pass a license key to the application at runtime, instead of using one that is installed on the system.

```
img2pdf -lk X-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX ...
```

This is required in an OEM scenario only.

5.2.10 -m Merge input to multi-page output

Merge input to multi-page output -m

Merge several input files into one image output file.

Example: Merge all TIFF files in a folder into one TIFF file

```
img2img -m *.tif output.tif
```

5.2.11 -pg Page range

Page range -pg <first> <last>

If not all pages should be converted, the page range can be defined using the **-pg** switch. Use 1 for the first page, use -1 for the last page. This switch is not available when merging (see [-m](#)).

Example: Convert the first page only

```
img2img -pg 1 1 input.tif output.tif
```

Example: Convert all but the first page to jpg

```
img2img -pg 2 -1 -s input.tif output%d.jpg
```

5.2.12 -q Set image quality

Set image quality -q <quality>

Set the quality index for lossy compression such as JPEG. Allowed values are 1 (lowest) to 100 (highest). Default is 80.

Example: Set the image quality to 50

```
img2img -q 50 input.jpg output.jpg
```

A quality index of 100 means lossless compression if the format supports it (JPEG2000).

5.2.13 -r Allow re-sampling

Allow re-sampling -r

Forces re-sampling where only the resolution (see [-d](#)) or only the true_width/true_height (see [-wt](#), [-ht](#)) is set. For more information about the behavior, see [Specification of resolution and image dimensions](#).

Example: Set resolution of output to 72 DPI, let true_width and true_height be constant

```
img2img -r -d 72 input.tif output.tif
```

5.2.14 -s Split multi-page input to single-page output

Split multi-page input to single-page output -s

Split a multiple-page input file (e.g. a 5-page TIFF file) into single-page output files (e.g. 5 separate TIFF files).

5.2.15 -t Transform colors from CMYK to RGB

Transform colors from CMYK to RGB -t

Apply a color conversion and convert CMYK to RGB.

5.2.16 -wp Set width

Set width -wp <width>

Set the width in pixels of the output image. The height is calculated respecting proportions. If the height is set too (see [-hp](#)), the height is omitted. For more information about the behavior, see [Specification of resolution and image dimensions](#).

Example: Set width of output file to 1500 pixels and resolution to 300 DPI

```
img2img -wp 1500 -d 300 input.tif output.bmp
```

5.2.17 -wt Set true width

Set true width -wt <true_width>

Set the true_width in mm of output image. The true_height is calculated respecting proportions. If the true_height is set too (see [-ht](#)), the true_height is omitted. For more information about the behavior, see [Specification of resolution and image dimensions](#).

Example: Set true_width of output file to 500 mm and width to 1000 pixels

```
img2img -wt 500 -wp 1000 input.bmp output.tif
```

5.3 Options for PdfOcr

5.3.1 -le List OCR engines

List OCR engines -le

OCR engines are accessed through the corresponding OCR interface DLLs. The following engines are supported:

Abbyy FineReader 11 OCR engine This engine is accessed by the OCR interface DLL `pdfocrpluginAbbyy11.ocr`.

Abbyy FineReader 10 OCR engine This engine is accessed by the OCR interface DLL `pdfocrpluginAbbyy10.ocr`.

3-Heights® OCR service This service is accessed by the OCR interface DLL `pdfocrpluginService.ocr`. The service accesses the Abbyy FineReader 10 or 11 OCR Engine.

The OCR interface DLLs are provided by the 3-Heights® Image to PDF Converter Shell. The OCR engine is provided as a separate product, such as 3-Heights® OCR Enterprise Add-on.

Here is an example of listing available OCR engines:

```
img2pdf -le
List of available OCR engines:
- abbyy11
- abbyy10
- service
End of list.
```

In order to make use of the OCR engine, the OCR interface DLL and the OCR engine must be installed. The `-le` switch lists all available OCR interface DLLs. It does not verify the corresponding OCR engine is installed and can be initialized. The OCR engine is actually accessed when using the switch `-ocr`.

5.3.2 -lk Set license key

Set license key -lk <key>

Pass a license key to the application at runtime, instead of using one that is installed on the system.

```
img2pdf -lk X-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX ...
```

This is required in an OEM scenario only.

5.3.3 -o Owner password

Owner password -o <owner>

The owner password is required to change the security settings of the document. To apply permission flags, an owner password must be set. Permission flags are set with the `-p` switch.

Example: Encrypt a document and set the owner password to `<owner>`.

```
img2pdf -o owner input.pdf output.pdf
```

5.3.4 -oca Rotate the image to the detected angle

Rotate the image to the detected angle -oca

The OCR engine may detect that an image needs to be rotated to have the text in an up-right position. If this is the case and this switch is used, then the original image is replaced by the rotated image.

5.3.5 -ocb Convert images to bitonal before OCR recognition

Convert images to bitonal before OCR recognition -ocb

Specify whether the images should be converted to bitonal (black and white) before OCR recognition. Enabling this feature can improve the memory consumption of the OCR process.

5.3.6 -ocbc Embed barcodes

Embed barcodes -ocbc

Embed the recognized barcodes in the XMP metadata.

5.3.7 -occs Correct skew angle

Correct skew angle -occs

Correct the skew angle of images.

This option has only an effect if the required information is provided by the OCR engine, which depends on the type and settings of the engine.

This option may change the appearance of the page and is only recommended for simple scanned documents that consist of a single image.

Using the option for digital-born documents may destroy the page layout.

5.3.8 -ocd Resolution for OCR recognition

Resolution for OCR recognition -ocd `<n>`

Re-sample images to target resolution before they are sent to the OCR engine. If no value is set, images are re-sampled to 300 DPI for OCR, which is the preferred resolution for most OCR engines.

5.3.9 -oci Do not deskew image

Do not deskew image -oci

Do not deskew original image (with [-ocs](#) only). This option specifies whether the image and text are deskewed according to the recognized skew angle.

- With option [-oci](#):
Do not change skew of images (i.e. do not change appearance of the page). This setting is recommended for born-digital documents.
- Without option [-oci](#):
Rotate image, such that lines of text are made horizontal. This might change the appearance of the page. This setting is recommended for scanned documents.

5.3.10 -ocl Set OCR language

Set OCR language -ocl <languages>

To optimize the performance of the OCR engine, it can be given hints what languages are used. The default language of the Abbyy FineReader 11 OCR Engine is English. This switch can only be used if the switch [-ocr](#) is set. This setting depends on the OCR engine.

The following switch set the languages to English and German:

```
img2pdf -ocr abbyy11 -ocl "English, German" input.tif output.pdf
```

See also documentation for the 3-Heights® OCR Add-on.

5.3.11 -ocm OCR mode

OCR mode -ocm <n>

Specify behavior of the 3-Heights® Image to PDF Converter Shell for files with existing OCR text. Available OCR modes are the following:

- 1 Only perform OCR for images without existing OCR text (default).
- 2 If OCR engine is active, remove old OCR text and perform OCR for all images. Hence, existing OCR text is not removed if OCR engine is not active.
- 4 Only perform OCR if input file contains no text.

Example: Set OCR mode 2

```
img2pdf -ocr abbyy11 -ocm 2 input.pdf output.pdf
```

5.3.12 -ocp Set OCR parameters

Set OCR parameters -ocp <params>

Using this switch, OCR engine specific parameters (key/value pairs) can be set to optimize the performance.

The following switch sets a predefined profile (i.e. a configuration setting), which is optimized for creating electronic archives with high accuracy:

```
img2pdf -ocr abbyy11 -ocp "PredefinedProfile = DocumentArchiving_Accuracy"
input.tif output.pdf
```

See also documentation for the 3-Heights® OCR Add-on.

5.3.13 -ocr Load OCR engine

Load OCR engine -ocr <name>

If a PDF document has to be made fully text searchable, even if the text is part of a raster image, then the images that are contained in the PDF document must be run through an OCR engine. With this switch, the user can select an OCR engine, e.g. Abbyy11, and instruct the tool to embed the recognized text as a hidden layer on top of the image. If the add-in is not found or the engine cannot be initialized (because it is not installed or the license key is not valid), then an error message is issued.

The name of the OCR engine can be retrieved using the switch -le. If the switch -ocr is not used, no OCR is applied.

Example: The following switch sets the OCR engine to the OCR Service

```
img2pdf -ocr service input.tif output.pdf
```

See also documentation for the 3-Heights® OCR Add-on.

5.3.14 -ocri Reembed pre-processed image

Reembed pre-processed image -ocri

This option currently requires the [-occs](#) to be set.

The OCR engine deskews and denoises the input image before recognizing the characters. This option controls whether the 3-Heights® Image to PDF Converter Shell should use the pre-processed image or keep the original image.

This option has only an effect, if the pre-processed image is provided by the OCR engine, which depends on the type and settings of the engine.

If this option is set, the resulting image may have a different color space, compression and size.

Since this option currently requires [-occs](#), it is recommended only for simple scanned documents.

5.3.15 -ocs Do not reembed deskewed image

Do not reembed deskewed image -ocs

The OCR engine deskews and denoises the input image before recognizing the characters. This option controls whether the 3-Heights® Image to PDF Converter Shell should use the deskewed image or keep the original image.

■ With option -ocs:

Embed the original image (also see option [-oci](#)). This setting is recommended for born-digital documents.

- Without option [-ocs](#):

Embed the deskewed and denoised image from the OCR engine. This might change the appearance of the page. This setting is recommended for scanned documents.

5.3.16 [-oct](#) Threshold resolution for OCR

Threshold resolution for OCR [-oct <n>](#)

Only images with a higher resolution than the threshold are re-sampled before OCR. The default is 400 DPI. If set to -1, no re-sampling is applied.

Example: Re-sample all images with a resolution of more than 300 DPI to 300 DPI:

```
img2pdf -ocd 300 -oct 1 input.tif output.pdf
```

Example: Re-sample all images with a resolution of 400 DPI or more to 300 DPI (default):

```
img2pdf -ocd 300 -oct 400 input.tif output.pdf
```

Example: Do not re-sample:

```
img2pdf -oct -1 input.tif output.pdf
```

Compatibility note: Initially, this switch was called [-ocD](#) and then renamed to [-oct](#) to avoid confusions with the switch [-ocd](#).

5.3.17 [-p](#) Permission flags

Permission flags [-p <flags>](#)

This option sets the permission flags. It is only usable when producing encrypted documents. In other words, at least an owner password must be set with [-o](#), and additionally a user password can be set with [-u](#). When omitting the [-p](#) option, then all permissions are granted. The permissions that can be granted are listed below.

Permission flags

Value	Description
0	Allow nothing (no permissions are granted)
4	Allow printing (low resolution)
8	Allow change of the document

Permission flags

16	Allow content copying or extraction
32	allow annotations
256	Allow filling of form fields
512	Support disabilities
1024	Allow document assembly
2048	Allow high quality printing

Any combination of permissions can be granted by adding up their values.

Example: The following command sets the owner password to “owner” and the permission flags to allow “printing in low resolution” 4 and “allow form filling” 256.

```
img2pdf -o owner -p 260 input.tif output.pdf
```

Example: “High quality printing” 2048 requires the standard printing flag 4 to be set too.

```
img2pdf -o owner -p 2052 input.tif output.pdf
```

For further information about the permission flags, see [PDF Reference 1.7](#) Section 3.5.2.

5.3.18 -pw Read an encrypted PDF file

Read an encrypted PDF file -pw <password>

A PDF document that has a user password (the password to open the document) can only be processed when either the user or the owner password is provided. The password can be provided using the option **-pw** followed by the password.

Example: The input PDF document is encrypted with a user password. Either the user or the owner password of the input PDF is “mypassword”. The command to process such an encrypted file is:

```
img2pdf -pw mypassword input.pdf output.pdf
```

When a PDF is encrypted with a user password and the password is not provided or is incorrect, the 3-Heights® Image to PDF Converter Shell cannot read and process the file. Instead it generates the following error message:

```
Password wasn't correct.
```

5.3.19 -u User password

User password -u <user>

Set the user password of the document. If a document that has a user password is opened for any purpose (such as viewing, printing, editing), either the user or the owner password must be provided.

A user who knows the user password is able to open and read the document. A user who knows the owner password is able to open, read, and modify (e.g. change passwords) the document. A PDF document can have none, either, or both passwords.

Example: Encrypt a document with a user and an owner password

```
img2pdf -u userpassword -o ownerpassword input.tif output.pdf
```

5.3.20 -v Verbose mode

Verbose mode -v

This option turns on the verbose mode.

In the verbose mode, additional information during the processing is written to the shell.

5.4 Supported image extensions

The following extensions are supported:

Supported image extensions table

Supported Image Extensions	
.tif, .tiff	Tagged Image File Format
.jpg, .jpe, .jpeg	Joint Photographic Expert Group
.png	Portable Network Graphics
.gif	Graphics Interchange Format
.bmp	Window Bitmap
.jb2	Joint Bi-level Image Experts Group
.jp2	JPEG2000
.jpx	Extended JPEG2000
.pbm, .pgm, .pnm, .ppm	Portable Bitmap File Format
.eps	Encapsulated PostScript (Output only)

5.5 Compression types

The following compression types can be set:

Value	Compression
0	raw
1	JPEG
2	Flate (ZIP)
3	LZW
4	CCITT Fax Group 3
5	CCITT Fax Group 3 2D
6	CCITT Fax Group 4
7	JBIG2 (Supported in PDF 1.4 or later)
8	JPEG2000 (Supported in PDF 1.5 or later, not supported in PDF/A-1)

5.6 Compression types: TIFF only

The following compression types can be set for converting TIFF files.

Value	Compression
0	raw
1	JPEG
2	Flate (ZIP)
3	LZW
4	CCITT Fax Group3
5	CCITT Fax Group3 2D
6	CCITT Fax Group 4
9	CCITT TIFFJPEG

Compression types 7 (Jbig) and 8 (JPEG2000) are not applicable for TIFF.

5.7 Specification of resolution and image dimensions

The three image dimensions (resolution, true_width and width) depend on each other. They have to satisfy the following relation (the same is true for the height and the true_height):

$$\text{resolution} = \frac{\text{width}}{\text{true_width}}$$

If the width (see [-wp](#)) and the height (see [-hp](#)) are set at the same time, the height is omitted due to priority of width. Equivalently, true_width has priority to true_height. All transformations are done respecting image proportions. The option [-r](#) (allow_resampling) can be used to force in certain situations to perform a re-sampling. The table below enlists the possible parameter combinations and shows the behavior of the Image to Image Converter Shell.

Options				Properties of Output Image
-d	-wp/-hp	-wt/-ht	-r	
not set	not set	not set	set/not set	no changes
not set	set	not set	set/not set	true_width/true_height constant, resolution modified
not set	set	set	set/not set	resolution modified
not set	not set	set	set	true_width/true_height constant, width/height modified
not set	not set	set	not set	width/height constant, true_width/true_height modified
set	not set	not set	set	true_width/true_height constant, width/height modified
set	not set	not set	not set	width/height constant, true_width/true_height modified
set	set	not set	set/not set	true_width/true_height modified
set	not set	set	set/not set	width/height modified
set	set	set	set/not set	width/height and true_width/true_height have priority over resolution

6 Return codes

All return codes other than 0 indicate an error in the processing. Messages with return code 0 are written to stdout. Messages with return codes other than 0 are written to stderr. Status and processing messages with no return code can be written to either stdout or stderr. See the specific list of messages below. Providing no options returns the usage and return code 0.

6.1 Image to PDF Converter

Table: Return codes for Image to PDF Converter

Error code	Meaning	Possible messages
0	Success	<ul style="list-style-type: none">No return stringReturn usage
1	Input file could not be opened or invalid parameters	<ul style="list-style-type: none">"Couldn't open control file""Too many parameters""Couldn't import image file %s."
2	The PDF output file could not be written.	<ul style="list-style-type: none">"Couldn't create output file %s"
3	Option error	<ul style="list-style-type: none">"Key/value pair %s doesn't contain a '='""Invalid option %s."
4	OCR error	
5	Decode errors	
10	License error	
none	Information (stdout)	<ul style="list-style-type: none">"Done.""Converting file %s(%d)"

In the above table, %s is the corresponding parameter name (file name or option) and %d a page counter.

6.2 Image to Image Converter

Table: Return codes for Image to PDF Converter

Error code	Meaning	Possible messages
0	Success	<ul style="list-style-type: none">No return stringReturn usage

Table: Return codes for Image to PDF Converter

1	Input file could not be opened or invalid parameters	<ul style="list-style-type: none">■ "Specify at least one input and one output file name."■ "Format of output file %s isn't supported."■ "Too many parameters"■ "Couldn't create output file %s."
2	The PDF output file could not be written.	<ul style="list-style-type: none">■ "Couldn't create output file %s"
3	Option error	<ul style="list-style-type: none">■ "Too many parameters."■ "Invalid option %s."■ "Cannot split and merge at the same time."
10	Invalid license	
none	Information (stdout)	<ul style="list-style-type: none">■ "Done."■ "Converting image %s(%d)"
none	Skipped processing errors (stderr)	<ul style="list-style-type: none">■ "Format of input file %s isn't supported."■ "Couldn't read page %d of input file %s."■ "Couldn't open input file %s."■ "Inappropriate Output file Name."■ "Conversion from input to output image format is not supported."■ "Output file %s doesn't support multiple pages."

In the above table, %s is the corresponding parameter name (file name or option) and %d a page counter.

6.3 PdfOcr

All return codes other than 0 indicate an error in the processing.

Return codes

Value	Description
0	Success.
1	Couldn't open input file.
2	PDF output file could not be created.
4	OCR engine error.
5	OCR parameter error.
6	OCR error.
10	License error, e.g. invalid license key.

7 Version history

7.1 Changes in versions 6.19–6.27

- **Update** license agreement to version 2.9

7.2 Changes in versions 6.13–6.18

- **New** supported image format HEIC/HEIF.

7.3 Changes in versions 6.1–6.12

- **Improved** search algorithm for installed fonts: User fonts under Windows are now also taken into account.

7.4 Changes in version 5

- **New** additional supported operating system: Windows Server 2019.

7.5 Changes in version 4.12

- **New** OCR plugin "abbyy12" for the ABBYY FineReader 12 engine.
- **Improved** reading and recovery of corrupt TIFF images.
- **New** HTTP proxy setting in the GUI license manager.

Shell `img2pdf`

- **Changed** behavior, option set page dimension `-sp` disables option adjust page `-a`.

7.6 Changes in version 4.11

- **Merged** manual `PdfOcrShell.pdf` into `Image2PdfShell.pdf`.
- **New** support for reading and writing PDF 2.0 documents.
- **Improved** font subsetting of CFF and OpenType fonts.
- **Improved** repair of corrupt image streams.

7.7 Changes in version 4.10

Shell `img2pdf`

- **New** option `-ao`: Adjust page orientation.

Shell `img2img`

- **New** option `-pg first last`: If not all pages should be converted, the page range can be defined explicitly.

7.8 Changes in version 4.9

- **Improved** metadata generation for standard PDF properties.
- **New** option `-h`: Set the dithering algorithm.
- **New** option `-b`: Set the color depth. Available: Bi-tonal: **1**. When using 1 bit per pixel, it is suggested to set a suitable dithering algorithm.

7.9 Changes in version 4.8

Shell `img2img`

- **New** option `-hp` to set the height of the image in pixel.
- **New** option `-ht` to set the true height of the image in mm.
- **New** option `-wp` to set the width of the image in pixel.
- **New** option `-wt` to set the true width of the image in mm.
- **New** option `-r` to force re-sampling.
- **New** option `-dd` to set the resolution of input image, if input image has none.
- **Changed** option `-d`, no re-sampling performed when set only.
- **Deprecated** option `-h`, use now `-hp`.
- **Deprecated** option `-w`, use now `-wp`.

8 Licensing, copyright, and contact

Pdftools (PDF Tools AG) is a world leader in PDF software, delivering reliable PDF products to international customers in all market segments.

Pdftools provides server-based software products designed specifically for developers, integrators, consultants, customizing specialists, and IT departments. Thousands of companies worldwide use our products directly and hundreds of thousands of users benefit from the technology indirectly via a global network of OEM partners. The tools can be easily embedded into application programs and are available for a multitude of operating system platforms.

Licensing and copyright The 3-Heights® Image to PDF Converter Shell is copyrighted. This user manual is also copyright protected; It may be copied and distributed provided that it remains unchanged including the copyright notice.

Contact

PDF Tools AG
Brown-Boveri-Strasse 5
8050 Zürich
Switzerland
<https://www.pdf-tools.com>
pdfsales@pdf-tools.com