NAME
OptiPNG – Advanced optimization program for **Portable Network Graphics (PNG)**

SYNOPSIS
optipng [**−?** | **−h** | **−help**]
optipng [*options...*] *files...*

DESCRIPTION
The **OptiPNG** program shall attempt to *optimize* PNG files, i.e. reduce their size to a minimum, without losing any information. In addition, this program shall perform a suite of auxiliary functions like integrity checks, metadata recovery and pixmap-to-PNG conversion.

The optimization attempts are not guaranteed to succeed. Valid PNG files that cannot be optimized by this program are normally left intact; their size will not grow. The user may request to override this default behavior.

FILES
The input files are raster image files encoded either in PNG format (the native format), or in an external format. The currently supported external formats are GIF, BMP, PNM and TIFF.

OptiPNG processes each image file given in the command line as follows:

− If the image is in PNG format:

Attempts to optimize the given file in-place. If optimization is successful, or if the option **−force** is in effect, replaces the original file with its optimized version. The original file is backed up if the option **−keep** is in effect.

− If the image is in an external format:

Creates an optimized PNG version of the given file. The output file name is composed from the original file name and the `.png` extension.

OPTIONS
General options
-?, **−h**, **−help**
> Show a complete summary of options.

−o *level*
> Select the optimization level.
> The optimization level 0 enables a set of optimization operations that require minimal effort. There will be no changes to image attributes like bit depth or color type, and no recompression of existing IDAT datastreams.
> The optimization level 1 enables a single IDAT compression trial. The trial chosen is what **OptiPNG** *thinks* it's probably the most effective.
> The optimization levels 2 and higher enable multiple IDAT compression trials; the higher the level, the more trials.
> The behavior and the default value of this option may change across different program versions. Use the option **−h** to see the details pertaining to your specific version.

−dir *directory*
> Write output file(s) to *directory*.

−fix
> Enable error recovery. This option has no effect on valid input files.
> The program will spend a reasonable amount of effort to recover as much data as possible, without increasing the output file size, but the success cannot be generally guaranteed. The program may even increase the file size, e.g., by reconstructing missing critical data. Under this option, integrity shall take precedence over file size.

When this option is not used, the invalid input files are left unprocessed.

–force   Enforce writing of a new output file.
           This option overrides the program's decision not to write such file, e.g. when the PNG input is digitally signed (using dSIG), or when the PNG output becomes larger than the PNG input.

–keep   Keep a backup of the modified file(s).
           The files that use the backup names prior to the program execution are *not* overwritten.

–log *file*
           Log messages to *file*. For safety reasons, *file* must have the extension `.log`.

–out *file*
           Write output file to *file*. The command line must contain exactly one input file.

–preserve
           Preserve file attributes (time stamps, file access rights, etc.) where applicable.

–quiet   Run in quiet mode.
           These messages are still written to the log file if the option **–log** is in effect.

–simulate
           Run in simulation mode: perform the trials, but do not create output files.

–snip   Cut one image out of multi-image, animation or video file(s).
           Depending on the input format, this may be either the first or the most relevant (e.g. the largest) image.

–v   Enable the options **–verbose** and **–version**.

–verbose
           Run in verbose mode.

–version
           Show copyright, version and build info.

––   Stop option switch parsing.

PNG encoding and optimization options

–f *filters*
           Select the PNG delta filters.
           The *filters* argument is specified as a rangeset (e.g. **–f0–5**), and the default *filters* value depends on the optimization level set by the option **–o**.
           The filter values 0, 1, 2, 3 and 4 indicate static filtering, and correspond to the standard PNG filter codes (*None*, *Left*, *Up*, *Average* and *Paeth*, respectively). The filter value 5 indicates adaptive filtering, whose effect is defined by the **libpng**(3) library used by **OptiPNG**.

–full   Produce a full report on IDAT. This option might slow down the trials.

–i *type*   Select the interlace type (0–1).
           If the interlace type 0 is selected, the output image shall be non-interlaced (i.e. progressive-scanned). If the interlace type 1 is selected, the output image shall be interlaced using the *Adam7* method.
           By default, the output shall have the same interlace type as the input.

–nb   Do not apply bit depth reduction.

–nc   Do not apply color type reduction.

–np   Do not apply palette reduction.

–nx   Do not apply any lossless image reduction: enable the options **–nb**, **–nc** and **–np**.

–nz   Do not recode IDAT datastreams.
           The IDAT optimization operations that do not require recoding (e.g. IDAT chunk concatenation) are still performed.

This option has effect on PNG input files only.

−zc *levels*

Select the zlib compression levels used in IDAT compression.
The *levels* argument is specified as a rangeset (e.g. **−zc6−9**), and the default *levels* value depends on the optimization level set by the option **−o**.
The effect of this option is defined by the **zlib**(3) library used by **OptiPNG**.

−zm *levels*

Select the zlib memory levels used in IDAT compression.
The *levels* argument is specified as a rangeset (e.g. **-zm8−9**), and the default *levels* value depends on the optimization level set by the option **−o**.
The effect of this option is defined by the **zlib**(3) library used by **OptiPNG**.

−zs *strategies*

Select the zlib compression strategies used in IDAT compression.
The *strategies* argument is specified as a rangeset (e.g. **-zs0−3**), and the default *strategies* value depends on the optimization level set by the option **−o**.
The effect of this option is defined by the **zlib**(3) library used by **OptiPNG**.

−zw *size*

Select the zlib window size (32k,16k,8k,4k,2k,1k,512,256) used in IDAT compression.
The *size* argument can be specified either in bytes (e.g. 16384) or kilobytes (e.g. 16k). The default *size* value is set to the lowest window size that yields an IDAT output as big as if yielded by the value 32768.
The effect of this option is defined by the **zlib**(3) library used by **OptiPNG**.

Notes

Options may come in any order (except for −−), before, after, or alternating with file names. Option names are case-insensitive and may be abbreviated to their shortest unique prefix.

Some options may have arguments that follow the option name, separated by whitespace or the equal sign ('='). If the option argument is a number or a rangeset, the separator may be omitted. For example:

    −out `newfile.png`   <=>   **−out=**`newfile.png`
    −o3   <=>   **−o 3**   <=>   **−o=3**
    −f0,3−5   <=>   **−f 0,3−5**   <=>   **−f=0,3−5**

Rangeset arguments are cumulative; e.g.

    −f0 −f3−5   <=>   **−f0,3−5**
    −zs0 −zs1 −zs2−3   <=>   **−zs0,1,2,3**   <=>   **−zs0−3**

EXTENDED DESCRIPTION

The PNG optimization algorithm consists of the following steps:

1. Reduce the bit depth, the color type and the color palette of the image. This step may reduce the size of the uncompressed image, which, indirectly, may reduce the size of the compressed image (i.e. the size of the output PNG file).

2. Run a suite of compression methods and strategies and select the compression parameters that yield the smallest output file.

3. Store all IDAT contents into a single chunk, eliminating the overhead incurred by repeated IDAT headers and CRCs.

4. Set the zlib window size inside IDAT to a mininum that does not affect the compression ratio, reducing the memory requirements of PNG decoders.

Not all of the above steps need to be executed. The behavior depends on the actual input files and user options.

Step 1 may be customized via the no-reduce options **−nb**, **−nc**, **−np** and **−nx**. Step 2 may be customized via

the **−o** option, and may be fine-tuned via the options **−zc**, **−zm**, **−zs** and **−zw**. Step 3 is always executed. Step 4 is executed only if a new IDAT is being created, and may be fine-tuned via the option **−zw**.

Extremely exhaustive searches are not generally expected to yield significant improvements in compression ratio, and are recommended to advanced users only.

EXAMPLES

```
optipng file1.png file2.gif file3.tif

optipng -o5 file1.png file2.gif file3.tif

optipng -i1 -o7 -v -full -sim experiment.png
```

BUGS

Lossless image reductions are not completely implemented. (This does NOT affect the integrity of the output files.) Here are the missing pieces:

- The color palette reductions are implemented only partially.
- The bit depth reductions below 8, for grayscale images, are not implemented yet.

TIFF support is limited to uncompressed, PNG-compatible (grayscale, RGB and RGBA) images.

Metadata is not imported from the external image formats.

There is no support for pipes or streams.

SEE ALSO

png(5), **libpng**(3), **zlib**(3), **pngcrush**(1), **pngrewrite**(1).

STANDARDS

The files produced by **OptiPNG** are compliant with **PNG−2003**:
Glenn Randers-Pehrson et al. *Portable Network Graphics (PNG) Specification, Second Edition.*
W3C Recommendation 10 November 2003; ISO/IEC IS 15948:2003 (E).
`http://www.w3.org/TR/PNG/`

AUTHOR

OptiPNG is written and maintained by Cosmin Truta.

This manual page was originally written by Nelson A. de Oliveira for the Debian Project. It was later updated by Cosmin Truta, and is now part of the **OptiPNG** distribution.