
PUBLIC LICENSE

Set Partitioning in Hierarchical Trees (SPIHT) Algorithm in MATLAB Program Language

Non-recursive version 1.0 2/12/2008

The authors of this MATLAB source code are Dr. Mustafa Sakalli (e-mail: msakalli@cipr.rpi.edu) and Dr. William A. Pearlman (e-mail: wpearlman@spiht.com).

The Set Partitioning In Hierarchical Trees (SPIHT) algorithm is protected by US Patent #5,764,807 (issued June 9, 1998), US Patent #6,674,911 (issued January 6, 2004), and other international patents and patents pending. The implementation of the SPIHT algorithm, included herein is furnished by PrimaComp, Inc., exclusive holder of patent rights.

PrimaComp, Inc., hereby grants the following license governing the terms and conditions for use, copying, distribution, and modification of the SPIHT algorithm implementation contained herein (hereafter referred to as "the SPIHT source code").

0. Use of the SPIHT source code, including any executable-program or linkable-library form resulting from its compilation, is restricted to solely academic or non-commercial research activities.
1. Any other use, including, but not limited to, use in the development of a commercial product, use in a commercial application, or commercial distribution, is prohibited by this license. Such acts require a separate license directly from PrimaComp, Inc..
2. For academic and non-commercial purposes, this license does not restrict use; copying, distribution, and modification are permitted under the terms of the GNU General Public License as published by the Free Software Foundation, with the further restriction that the terms of the present license shall also apply to all subsequent copies, distributions, or modifications of the SPIHT source code.

NO WARRANTY

3. PrimaComp, Inc., disclaims all warranties, expressed or implied, including without limitation any warranty whatsoever as to the fitness

for a particular use or the merchantability of the SPIHT source code.

4. In no event shall PrimaComp, Inc., be liable for any loss of profits, loss of business, loss of use or loss of data, nor for indirect, special, incidental or consequential damages of any kind related to use of the SPIHT source code.

END OF TERMS AND CONDITIONS

There are other versions of MATLAB SPIHT authored by M. Sakalli not covered in this license. The code in this release is the conventional (breadth-first search) SPIHT code. Other versions in MATLAB include:

- recursive conventional
- vectorized SPIHT - fast and slow versions
- blockwise –breadth-first search and depth-first search

There are also versions of SPIHT in C/C++ source code, which are likewise not covered by this public license.

Persons seeking to license the SPIHT algorithm for commercial purposes or for uses otherwise prohibited by this license may wish to contact PrimaComp, Inc., regarding the possibility of negotiating such licenses:

PrimaComp, Inc.
851 Maxwell Drive
Schenectady, NY 12309
U.S.A.
+1 (857) 231-6135
email: bmazor@spiht.com
<http://www.cipr.rpi.edu/research/SPIHT>

PROGRAM INFORMATION

The package of programs included in this distribution comprises a demonstration version of SPIHT written in MATLAB source language. As written, a single runtime command compresses the input image, writes it to a file (*.sp), decompresses the image and writes it to a file, and displays a figure with four images, the input image, the decompressed image (*Rec*.raw), the compressed wavelet transform, and the error image. The PSNR and MSE (mean squared error) are also calculated and displayed in the figure. The compressed file and reconstructed image are written to the directory of the input image. The input image must be 1-byte per pixel grayscale (monochrome). The wavelet transform is produced by recursive filtering with the 9/7 biorthogonal filters. The

raw bits put out by the SPIHT algorithm acting on the wavelet coefficients are sent directly to the code stream without entropy coding.

Usage: Compress "image_name.raw" , 1-byte pixels, r rows, c columns to CR bits/pel.

```
> mSphit_n( 'image_name', [r, c], CR <, 'aspectratio-colormap'>)
```

Quantities enclosed with braces < > are optional

r : number of image rows (height)

c : number of image columns (width) (default = r)

CR : compression rate in bits per pixel

aspectratio-colormap (optional): default aspect ratio is fixed to the image size, and the default colormap is the original gray. Different view colors are provided.

To change the color in a view mode just enter a number from 1 (gray) to 5 (orange-red). Default is 1 (gray).

For stretched view enter 'n', and (optional) with colormode 'nN', N is a number.

Examples:

Compress gray 8-bit image named 'goldy.raw', dimensions 512x 720 (height x width) to 0.50 bpp.

```
> mSpiht_n('goldy', [512,720],0.5) % normal view with gray colors
```

```
> mSpiht_n('bike', [512,720],0.5, 'n3') % stretched view with warmer colors
```