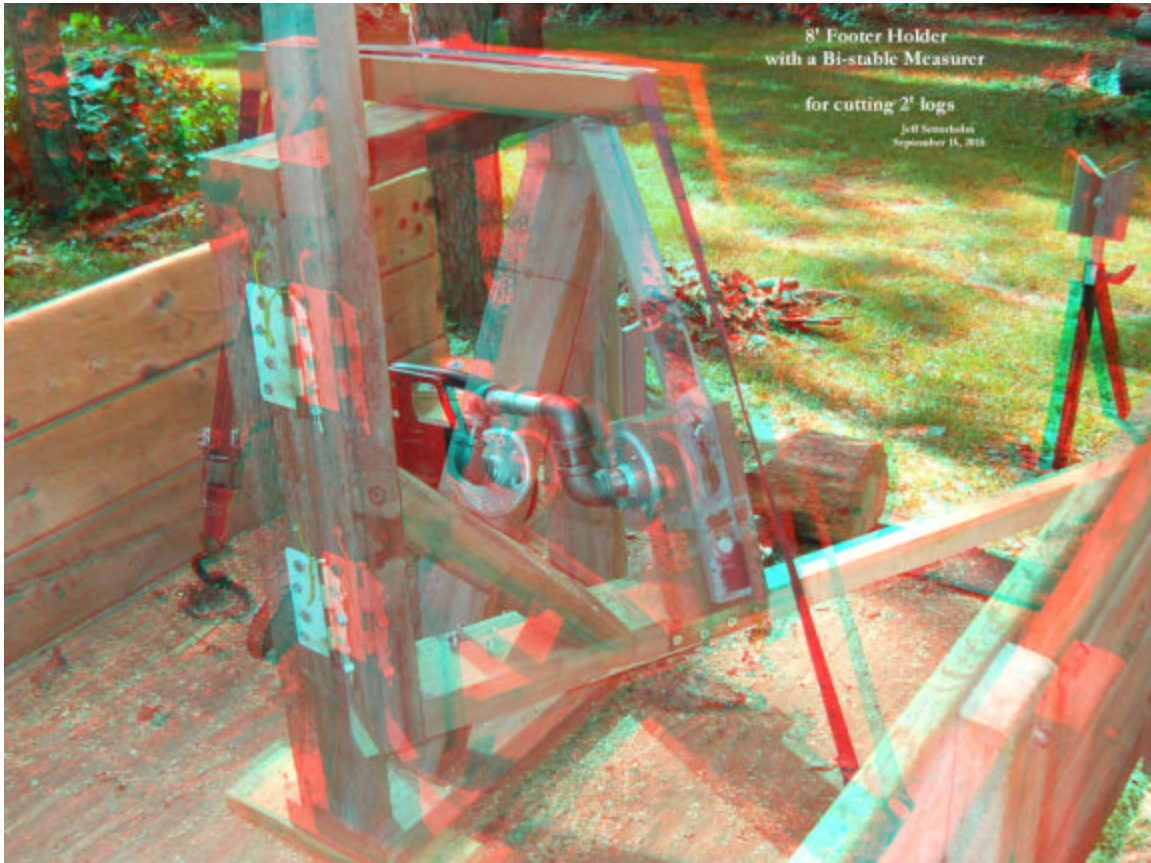# (Red) : (Green+Blue) 3D

Jeff Setterholm

Lakeville, Minnesota, U.S.A
jeff.setterholm@gmail.com

Sat 2015.09.19

The image above is a 3D stereo anaglyph created by merging **two** full-color images into **one** full-color image using:

A. The red pixels of the <u>left</u> image –and-
B. The green and blue pixels of the <u>right</u> image.

(I recommend viewing such images with Red/Cyan 3D glasses
such as American Paper Optics LLC's item: AGRCHHV.)

I came up with *and implemented* this idea on my own a few days ago.
To the extent of my rights therein:
I hereby place (Red) : (Green+Blue) 3D images in the public domain.

**While this notice remains posted on my website: www.setterholm.com/Images/Red-GreenBlue3D.pdf
… no one has proven to me that the idea doesn't rightfully belong in the public domain.**

Sun 2015.09.27 …a 57 megabyte image down-converted to a 28 megabyte RedGB3D in 5 seconds.

For experimental conversion software see: www.setterholm.com/Fortran/RedGB3D/
My runtime screen when converting a large, full-color qVP stereo pair:

```
                                                        2015.09.27.1203.11 L

        RedGB3D.exe    version 0.5    2015.09.27    Jeff Setterholm
                                                    jeff.setterholm@gmail.com

  On the web: http://ftp.setterholm.com/Fortran/RedGB3D/RedGB3D.exf
                                        change to:    ^.exe to run
                Initialization file:  /Fortran/RedGB3D/RedGB3D.ini
                     Source code    : /Fortran/RedGB3D/RedGB3D.f95

  Your text file named `RedGB3D.ini` initializes this program by providing
  three pieces of information:

  1. Your input .bmp image file name - a side-by-side stereo pair
     which can be either 24-bit color or 8-bit grayscale.

  2. a flag = -1 for a Right/Left ( =   crossed-eye) input image
            = +1 for a Left/Right (e.g. stereoscope) input image

  3. The name for the resulting `-REDGB3d.bmp` image'

  4. Convergence shift = screen depth adjustment, in pixels (if any)
        >0 moves screen depth to a point presently -farther- than the screen
        <0                                          -nearer -
        The R & GB halves can even be completely separated in the output.

  For example:
  "qVP.bmp"C               <- Your input .bmp image filename
  -1                       <- -1 for Right/Left ; +1 for Left/Right
  "qVP-RedGB3D.bmp"C       <- Your name for the output -RedGB3d.bmp image
  0                        <- Convergence adjustment, if any.

                  This software is experimental & has NO warranties.
                   In particular, input error checking is minimal.
                  You're welcome to improve the source code yourself.

                  Use the program ONLY at your own risk.

            Press 0 to exit now  -or-  Press 1 to proceed:1

                                                        2015.09.27.1203.14 L
    Searching for StPaul-LookingDown-Qvp.bmp
    ...found.  Opening...
     Bitmap Header (BH) values:
     BM=BM            nSizeTot=  57749814   nReserv1=       0   nReserv2=       0
     nOffBit =       54   nSizeH  =       40   nWidth  =    4320   nHeight =    4456
     nPlanes =        1   nBitsPP =       24   nCompres=       0   nSizeC  =       0
     XPpM    =     0.00   YPpM    =     0.00   nClrUsed=       0   nClrImpo=       0

  I1.PixelsPerRow       =     4320
  I1.PixelsConvAdj      =        0
  I1.BytesPerPix        =        3
  I1.BytesPerRow        =    12960
  I1.PadBytes           =        0
  I1.BytesPerRowPadded=     12960

  Output image:
     Bitmap Header (BH) values:
     BM=BM            nSizeTot=  28874934   nReserv1=       0   nReserv2=       0
     nOffBit =       54   nSizeH  =       40   nWidth  =    2160   nHeight =    4456
     nPlanes =        1   nBitsPP =       24   nCompres=       0   nSizeC  =       0
     XPpM    =     0.00   YPpM    =     0.00   nClrUsed=       0   nClrImpo=       0

  I2.PixelsPerRow       =     2160
  I2.PixelsConvAdj      =        0
  I2.BytesPerPix        =        3
  I2.BytesPerRow        =     6480
  I2.PadBytes           =        0
  I2.BytesPerRowPadded=      6480
                                                        2015.09.27.1203.18 L
    Program RedGB3d.exe:
      is about to write output file:StPaul-LookingDown-Qvp-RedGB3D

      which is  2160 pixels by  4456 pixels.

            Press 0 to exit now  -or-  Press 1 to proceed:1

                                                        2015.09.27.1203.18 L
                                                        2015.09.27.1203.19 L

  RedGB3D.exe image processing completed. Press enter.
```