

Date: Tue, 05 Feb 2008 19:02:33 +0900  
From: Tsukasa NAKANO  
To: Akira TSUCHIYAMA  
Cc: 飯田 洋祐, Kentaro Uesugi, Masayuki Uesugi  
Subject: programs-for-stereo-animation-via-STL

---

つちやまさま、

なかのです。今朝差し上げた E-mail に一カ所間違いを見つけたので、修正版を再送します。それから、可能なら、明日からの実験に SHARP の 3 次元ディスプレイを持って来てもらえませんか？ よろしくお願ひします。

つちやまさま、

なかのです。先日の E-mail で紹介したステレオ画像のアニメーション作成用プログラム群が一応完成したので紹介します。これは以下の 2 種類のプログラム群から構成されます：

[1] STL データファイル中の物体像を様々な方向から鳥瞰した図に対応するステレオペアの TIFF 画像の複数個を作成するプログラム群。これには STL シリーズのプログラム群 stl\_bev\* で描画できた 3 × 2 種類の鳥瞰図に対応する 2 枚一組のステレオペアの画像を同時に作成できる以下の 6 個のプログラムがある：

#### stl\_sp と stl\_sp\_SS

従来の stl\_bev と stl\_bev\_SS に対応した物体像に陰影を付けただけのグレースケールの鳥瞰図のステレオペアの画像を作成するプログラム。"\_SS" 付きのプログラムは描画時に物体像の表面のスムージングも行う。

#### stl\_sp\_C と stl\_sp\_C\_SS

stl\_bev\_C と stl\_bev\_C\_SS に対応した物体像を陰影を考慮したカラーで表現した鳥瞰図のステレオペアの画像を作成するプログラム。"\_SS" 付きのプログラムは物体像の表面のスムージングも行う。

#### stl\_sp\_wf と stl\_sp\_wf\_nf

stl\_bev\_wf と stl\_bev\_wf\_nf に対応した物体像表面をワイアフレームで表した鳥瞰図のステレオペアの 2 値画像を作成するプログラム。"\_nf" なしのプログラムは物体像がちょうど納まる直方体領域を示す枠線 (frame) を描く。

[2] それぞれのステレオペアの 2 枚の画像を合成し、それらをアニメーション GIF のファイルにまとめるプログラム群。個々のステレオペアの合成法が異なる以下の 3 個のプログラムがある：

**sp\_align**

ステレオペアの画像を単純に左右に並べて表示する。

**sp\_BR**

左目が青、右目が赤（「Right が Red」です）のフィルタ付きの眼鏡で立体視可能な画像を合成する。

**sp\_PBD**

視差バリア（parallax barrier）方式の液晶ディスプレイにそのまま表示すると立体視できる画像を合成する。

## (1) プログラムのインストール

上杉くんの FTP サイトにアップした以下のいずれかの書庫ファイルを適切なディレクトリにコピーして下さい。

<http://www-bl20.spring8.or.jp/~sp8ct/tmp/stereo.taz>

<http://www-bl20.spring8.or.jp/~sp8ct/tmp/stereo.zip>

これらを解凍・展開すると stereo/ というディレクトリが得られます。Cygwin や Linux の場合、以下を入力すれば展開・解凍できます：

```
tar xzf stereo.taz
```

こうして得た stereo/ の中には以下のディレクトリがあるはずです：

**src/**

プログラムのソースファイルが入っている。

**bin/**

はじめは空。UNIX 用の実行ファイルを入れる。

**exe/**

MinGW (Minimalist GNU for Windows) の GNU-C でコンパイルした Windows 用の実行ファイルが入っている。

**test/**

地球儀の STL データファイルとそれを用いた 3 種類の立体視アニメーション作成の実例を記した C-shell scripts および Windows 用のバッチファイルが入っている。

Windows でディレクトリ test/ の中のバッチファイルを試すだけなら特に何もする必要はありません。Windows のコマンドプロンプト (DOS 窓) を開いて任意のディレクトリからプログラムを実行したい場合は、ディレクトリ exe/ の中の実行ファイルすべてを実行パスに登録済みの適切なディレクトリにコピーするか、ディレクトリ exe/ (の絶対パス) を実行パスに登録して下さい。

Cygwinの上でも exe/ の中の実行ファイルを走らせることができます。Cygwinの上で exe/ の中の実行ファイルを使う場合はWindowsの場合と同様な実行パスに関する作業を行って下さい。ただし、test/ の中の C-shell scripts はディレクトリ bin/ 中にある実行ファイルを使って処理を行うので、これらを試したい場合には exe/ の中の実行ファイルすべてを bin/ にコピーして下さい。

UNIX (Linux や Cygwin) で今回のプログラム群すべてをインストールする場合は以下のようにキー入力すれば OK です :

```
# ディレクトリの移動
cd 何とかかんとか/stereo/src
# GNU-C を用いたプログラム群のコンパイル
make
# コンパイルしたプログラム (実行ファイル) 群のコピー
make install          # UNIX (Linux) の場合
make install.bin     # Cygwin の場合
# 不要なファイルの消去 (実施しなくてもかまいませんが)
make clean
```

この結果、今回のプログラム群の実行ファイルがディレクトリ bin/ にコピーされるので、これらすべてを実行パスに登録済みの適当なディレクトリにコピーするか、ディレクトリ bin/ (の絶対パス) を実行パスに登録して下さい。

## (2) ステレオペアを作成するプログラム群

これまでにぼくが作成した STL データから鳥瞰図を描くプログラム群では立体視用のステレオペアを作ることが困難でした。つまり、通常の左右に並べて立体視するステレオペアとしては指定した視線方向から左右 (投影した鳥瞰図の画像の左右) に適当な視差角 (parallax angle) だけ異なった方向から眺めた 2 枚の鳥瞰図が必要ですが、限られた場合を除いてそのような 2 個の視線方向の算出は簡単ではなく、かつ、それらを指定したとしても従来のプログラムでは回転しなければ通常の立体視ができない鳥瞰図の画像 (ペア) しか得られないことが普通でした。ここで紹介するプログラム stl\_sp\* はいずれも、指定した視線の方向に対して、その左右に指定した視差角だけ異なった方向から眺めた 2 枚の鳥瞰図の画像 (ステレオペアの画像) を同時に作成します。これらの画像ペアはそのまま左右に並べて表示すれば通常の立体視に使えます。

なお、STL シリーズのプログラム stl\_bev\* をもとにコーディングしたので、今回のプログラム群 (stl\_sp\* ; "sp" は stereo pair のつもり) の基本的な機能や使用法は従来の stl\_bev\* のものと概ね同じです :

今回のプログラム	対応する従来のプログラム
stl_sp	stl_bev
stl_sp_SS	stl_bev_SS
stl_sp_C	stl_bev_C
stl_sp_C_SS	stl_bev_C_SS
stl_sp_wf	stl_bev_wf
stl_sp_wf_nf	stl_bev_wf_nf

今回のプログラムの個々の起動法は以下の通りです。

グレースケールの鳥瞰図のステレオペアを作成するプログラム

物体像の表面をスムージングしない：

```
stl_sp STLs scale gamma bias bg fg
```

スムージングする（3次元画像から si\_stl\_[A,B,C] で作成した STL データを描画する場合に使うとよいです）：

```
stl_sp_SS STLs scale gamma bias bg fg
```

カラーの鳥瞰図のステレオペアを作成するプログラム

物体像の表面をスムージングしない：

```
stl_sp_C STLs scale gamma bias (改行しない)
```

```
bgR bgG bgB fgR fgG fgB scR scG scB
```

スムージングする（si\_stl\_[A,B,C]の STL データ用）：

```
stl_sp_C_SS STLs scale gamma bias (改行しない)
```

```
bgR bgG b gB fgR fgG fgB scR scG scB
```

黒白（ワイアフレーム）の鳥瞰図のステレオペアを作成するプログラム

物体像がちょうどおさまる直方体領域を示す枠線を描く：

```
stl_sp_wf STLs scale
```

枠線を描かない：

```
stl_sp_wf_nf STLs scale
```

ただし、起動パラメータの意味は以下の通りです。

#### STLs

STL データファイルの名前。複数を指定してもよく、その場合にはそれぞれが表現している物体像を指定された順に描く。

#### scale

正の値（浮動小数点数も可）を指定するとそれが STL データの座標値のスケールとなる。物体像はその全体が完全に入る画素の横縦幅を 1 とする正方形の画像の

上に投影される。また、負の値（整数値でなければならない）を指定すると、その絶対値が指す横縦画素数を持つ正方形の画像の上に適切にスケーリングされた物体像全体が投影される。

#### **gamma**

物体像の表面の陰影の強調の度合いを決める **gamma factor**。正の浮動小数点数で、通常は 1 を指定すればよい。大きな値を指定すると陰影の差違が大きくなる。

#### **bias**

完全な影の部分の明るさを指す **bias** の値。0～255 の範囲の値（浮動小数点数も可）で、通常は 32～64 の値を指定すればよい。値 0 を指定すると完全な影の部分は真っ黒になる。

#### **bg** もしくは **bgR**、**bgG**、**bgB**

物体像がない背景の部分に塗る画素値もしくは **RGB** 成分の値。0～255 の整数値を指定する。

#### **fg** もしくは **fgR**、**fgG**、**fgB**

いずれに対しても 0～255 の整数の値を指定すると、それらは物体像がちょうどおさまる直方体領域を示す枠線に塗る画素値もしくは **RGB** 成分値と見なされる。また、これらのいずれかが負の値なら、物体像の領域を示す枠線を描かない。

#### **scR**、**scG**、**scB**

**STL** データに含まれるかもしれない色情報を持たない三角形を塗るのに用いる色の **RGB** 成分の値（0～255 の整数値）。

このようにして起動すると、いずれのプログラムも描画すべきステレオペアの鳥瞰図の視線方向などの情報を標準入力から読み取ろうとします。その情報は以下の 2 種類の形式の行のいずれかで指定します。ただし、以下の括弧 “[ ]” 内の値は行中のパラメータの個数で、それらを空白もしくはタブコード区切りで並べた行にする必要があります。

#### **[5] lon lat hpa pathL pathR**

視線方向の経度と緯度が **lon** と **lat** で、その左右（描画する鳥瞰図の左右）の視差角の半分の角度 (**half parallax angle**) が **hpa** のステレオペアの 2 枚の鳥瞰図の画像を同時に作成し、左目用をパス名 **pathL** の、右目用を **pathR** の **TIFF** ファイルに格納する。ただし、これらの角度の単位はいずれも度であり浮動小数点数の任意の値を指定できるが、**hpa** に負の値を指定すると **pathL** の画像が右目用、**pathR** のものが左目用になる。

#### **[9] lon0 lat0 hpa0 dLon dLat dHpa count formatL formatR**

経度 **lon**、緯度 **lat** および視差角の半分の角度 **hpa** を

```
for index = 0 ~ count-1
```

$$\text{lon} = \text{lon0} + \text{dLon} \times \text{index}$$

$$\text{lat} = \text{lat0} + \text{dLat} \times \text{index}$$

$$\text{hpa} = \text{hpa0} + \text{dHpa} \times \text{index}$$

とした、count（自然数）個の鳥瞰図のステレオペアの画像を次々と描き、左目用の画像を文字列 `formatL` で、また、右目用を `formatR` で指定したパス名の TIFF ファイルにそれぞれ格納する。ただし、`formatL` と `formatR` はいずれも C 言語の標準ライブラリ関数 `sprintf()` などが取り扱うフォーマット指定用の文字列で、上記の `index` が指す整数値を変換できるようにする必要がある。例えば、個数 `count` が 100 以下なら `index` は 10 進数表示で 2 桁以下の整数値となるので、`pathL` と `pathR` としてそれぞれ "`L%02d`" と "`R%02d`" のような文字列を指定してやればよい。この場合、`index` の値が 1 の左目用の画像ファイルのパス名は "`L01`" に、右目用は "`R01`" になる。なお、Windows のバッチファイル上でコマンド `echo` を使ってこのようなフォーマット指定の文字列をプログラムに渡す場合は、特殊文字として解釈されないように記号 "%" を "%%" と記す必要があるかもしれない。

これらの標準入力からの指定では視差角の半分の値を与えなければならないことに御注意下さい。また、一連のステレオペアを作る後者の場合に `dHpa` に 0 でない値を与えると、それぞれが異なる視差角で描かれたステレオペアの画像になることにも注意が必要です。

異なる視差角のステレオペアを立体視すると物体像の「飛び出し方」が違うはずですが、ぼくはそれをテストしていません。これを試してみてください。

従来の `stl_bev*` とは異なり、今回のプログラム群は描画したそれぞれの鳥瞰図のステレオペアごとに以下の 13 個の値をタブコード区切りで 1 行にまとめて標準出力に書き出します。ただし、以下の括弧 “[ ]” 内に示したものは行上の値の並びの順番です。

[1~3] `lon`, `lat`, `hpa`

そのステレオペアの視線方向の経度、緯度と視差角の半分の値。ただし、これらの角度の値の単位は度です。

[4] `pathL`

左目用の鳥瞰図の画像を格納した TIFF ファイルのパス名。

[5~8] `x1L`, `y1L`, `x2L`, `y2L`

左目用の画像上で鳥瞰図や枠線などがちょうど入る長方形の対角点の座標値 (`x1L` と `y1L` は左上隅、`x2L` と `y2L` は右下隅)。

[9] `pathR`

右目用の鳥瞰図の画像を格納した TIFF ファイルのパス名。

[10~13] `x1L`, `y1L`, `x2L`, `y2L`

右目用の画像上で鳥瞰図や枠線などがちょうど入る長方形の対角点の座標値 (`x1L`

と y1L は左上隅、x2L と y2L は右下隅)。

これらの標準出力に書き出されるテキストデータは次に説明する立体視が可能なアニメーションを作成するプログラム群でそのまま使えるので、ファイルにリダイレクトするなどしておけばよいです。

### (3) 立体視可能なアニメーションを作成するプログラム群

プログラム群 `stl_sp*` で作った鳥瞰図の TIFF 画像のステレオペアはそのまま左右に並べて画面表示すれば通常の立体視が可能はずです。さらに、プログラム `sp_align` を使えば、画像上で物体像が描画されている領域だけをトリミングした後に、それらの画像ペアを左右に並べたものを次々と表示するアニメーション GIF のファイルを作成できます。また、`sp_BR` はトリミングしたグレースケールの鳥瞰図のステレオペアを左目が青で右目が赤のフィルタ付き眼鏡で立体視可能な画像に合成し、アニメーション GIF にします。そして、`sp_PBD` を使えば、SHARP 製の LL-151D などの視差バリア方式の液晶ディスプレイで立体視することが可能なアニメーション GIF も合成できます。なお、視差バリア方式のディスプレイによる立体視の原理は以下のホームページをご覧ください：

<http://www.wakayama-u.ac.jp/~tokoi/opengl/3dlcd.html>

これらのプログラムの起動法は以下の通りです。

複数のステレオペアのそれぞれの描画領域をトリミングした後、単純に左右に並べたアニメーション GIF を作成するプログラム：

```
sp_align centerSpace xMargin yMargin bgR bgG bgB GIF
```

青赤のフィルタ付き眼鏡で立体視可能なアニメーションを作成：

```
sp_BR xMargin yMargin R_and_B G GIF
```

視差バリア方式の液晶ディスプレイで立体視可能なものを作る：

```
sp_PBD xMargin yMargin bgR bgG bgB GIF
```

ただし、起動パラメータの意味は以下の通りです。

#### `centerSpace`

トリミング後のステレオペアの 2 枚の画像の間隔。画素幅を単位とする 0 以上の整数値を指定する。

#### `xMargin` と `yMargin`

合成したステレオペアの左右と上下に付ける余白の幅。画素幅を単位とする 0 以上の整数値を指定する。

#### `bgR`、`bgG`、`bgB`

前記の余白や画像のトリミングで生じた「空き」の部分に塗る色の成分値。0～255の整数値を指定する。

#### R\_and\_B

sp\_BR で青もしくは赤色に変換した左右の画像の余白や「空き」の部分に塗る画素値。0～255の整数値を指定する。

#### G

sp\_BR で左右の物体像の表示に使わない緑色の成分値。0～255の整数値を指定。

#### GIF

作成するアニメーション GIF のファイルのパス名。

これらのプログラムはいずれも、アニメーション化するステレオペアの TIFFF ファイルのパス名などの情報を標準入力から読み取ろうとします。この情報は以下の4通りの形式の行のいずれかで指定します。ただし、以下の括弧 “[ ]” 内の値は行中のパラメータの個数で、それらを空白もしくはタブコード区切りで並べた行にする必要があります。

#### [2] pathL pathR

ファイル名が pathL の TIFFF ファイルに格納されている左目用の画像と pathR の右目用の画像をそれぞれ読み込み、左目用と右目用のものごとに、それぞれの画像を最大の画像と同じ広がり領域の中央に配置する。

#### [6] pathL pathR x1 y1 x2 y2

pathL と pathR の画像上の鳥瞰図の描画領域のどちらもが対角点の4個の座標値が x1, y1, x2, y2 の長方形だとして、その画像領域のトリミングを行う。その後、左目用と右目用の画像ごとにそれぞれを最大のものの領域の中央に配置する。

#### [10] pathL x1L y1L x2L y2L pathR x1R y1R x2R y2L

pathL と pathR の画像の上の描画領域がそれぞれ x1L, y1L, x2L, y2L と x1R, y1R, x2R, y2R で指定された長方形だとして、それらの画像領域のトリミングを行う。その後、左目用と右目用のそれぞれの画像を最大のものの中央に配置する。

#### [13] lon lat hpa pathL x1L y1L x2L y2L pathR x1R y1R x2R y2L

最初の3個のパラメータを読み飛ばした後に入力形式 [10] と同じ処理を行う。

上記の入力形式 [13] はプログラム stl\_sp\* との連携のために設けました。つまり、後述するデモプログラムで行っているように、stl\_sp\* と sp\_\* をパイプラインで連結すればステレオ画像のアニメーションの作成が可能です。また、それ以外の入力形式を使えば、stl\_sp\* で作成したステレオペア以外の様々な画像のペアを合成したアニメーションも作成可能なはずです（ただし、sp\_BR や sp\_PBD の出力は特殊ですが）。



#### (4) デモプログラム

先のディレクトリ test/ の下には上杉くんの FTP サイトにアップした以下のステレオ画像のアニメーションを作る C-shell scripts と Windows 用バッチファイルが入っています。

[http://www-bl20.spring8.or.jp/~sp8ct/tmp/e\\_g\\_align.gif](http://www-bl20.spring8.or.jp/~sp8ct/tmp/e_g_align.gif)

C-shell script "e\_g\_align.csh" もしくはバッチファイル "e\_g\_align.bat" で作成した単純な立体視が可能な回転するカラーの地球儀のアニメーション。

[http://www-bl20.spring8.or.jp/~sp8ct/tmp/e\\_g\\_BR.gif](http://www-bl20.spring8.or.jp/~sp8ct/tmp/e_g_BR.gif)

"e\_g\_BR.csh" や "e\_g\_BR.bat" で作った左目が青で右目が赤のフィルタ付きの眼鏡で見れば立体視できるグレースケールの回転する地球儀のアニメーション。

[http://www-bl20.spring8.or.jp/~sp8ct/tmp/e\\_g\\_PBD.gif](http://www-bl20.spring8.or.jp/~sp8ct/tmp/e_g_PBD.gif)

"e\_g\_PBD.csh" もしくは "e\_g\_PBD.bat" で作成。視差バリア方式の液晶ディスプレイに表示すれば立体視が可能な回転するカラーの地球儀のアニメーション。

これらはいずれも、test/ の中に入っている color STL データファイル "e\_g.stl" を用いてアニメーションを作りますが、このファイルに関する説明は省略します（以前に紹介した "globe" シリーズのプログラム群を用いて作った STL データファイルです）。

Windows 用のバッチファイルはファイルブラウザでクリックすれば実行します。また、C-shell scripts は UNIX の端末環境 (Cygwin を含む) で以下のように入力して起動します。

```
# ディレクトリの移動
cd 何とかかんとか/stereo/test
# C-shell scripts の起動
e_g_align.csh
e_g_BR.csh
e_g_PBD.csh
```

長い E-mail になってすみません。とりあえず以上です。