

北陽電機さんから 3D LIDAR (YVT-x001) を試用する機会をいただいたため、そのお礼として、MATLAB を使って 3D LIDAR のデータ取得プログラムを公開する。

## MATLAB を使って、3D LIDAR のデータを取得するには？

3D 測域センサは、3次元データを扱うため、高速なデータ通信が不可欠である。そのため、USB 経由ではなく、Ethernet 経由でのデータのやり取りを行う。また、従来の北陽の 2D LIDAR とは異なる、VSSP プロトコル (Volumetric Scanning Sensor Protocol) を採用しており、これを解析するプログラムが必要となる。VSSP プロトコルの詳細については、ドキュメントにゆずるが、ここでは、MATLAB を使い通信、解析プログラムを作成していく。

## MATLAB で VSSP データを取り出す

VSSP プロトコルデータの先頭を判別するには、文字列 VSSP を見つける必要がある。MATLAB では、strfind 関数を使うと容易に見出せる。以下に、vssp\_data を切り出す vssp\_get 関数を示す。

vssp\_get.m

```
% Copyright (c) 2014 Gerox http://www.gerox.com
function [data,vssp_data,vssp_type,vssp_status]=vssp_get(data)
    ind=strfind(char(data),'VSSP');
    vssp_data=[];vssp_type=[];vssp_status=[];
    if isempty(ind) return;end
    data=data(ind(1):end);
% vssp=char(data(0+(1:4))); % 'VSSP' 1:4
    vssp_type=char(data(4+(1:3))); % '_ro' 5:7
    vssp_status=char(data(8+(1:3))); % '000' 9:11
    vssp_length = sum(data(14+(1:2)).*2.^(0:8:8)); % from 15:16
% vssp_recieved_time_ms = sum(data(16+(1:4)).*2.^(0:8:24)); % always zero
17:20
% vssp_send_time_ms = sum(data(20+(1:4)).*2.^(0:8:24)); % current_time
21:24
    vssp_data=data(1:vssp_length);
    data=data(ind(2):end);
```

vssp\_get 関数では、data から、vssp\_data、vssp\_type、vssp\_status を取り出す。

vssp\_type には、PNG,VER,GET,SET,DAT,\_ri,\_ro,\_ax などがある。

vssp\_status は、エラーが生じない限りは、000 となる。MATLAB での配列は 1 から始まる。

C 言語とは異なるので注意が必要である。

## vssp\_type \_ri の解析関数

この関数では、vssp\_get 関数で抽出した vssp\_data を解析し、距離ベクトル、角度ベクトルを取り出す。3D LIDAR は、4つまでのマルチエコーに対応しているため、それぞれ、r1 から r4 は、距離ベクトル、vind1 から vind4 は、vertical 角度のための配列インデックスベクトル、hrad1 から hrad4 は、horizontal 角度(rad)ベクトル、i1 から i4 は、強度ベクトル、frame は、フレーム番号を出力する。for 文を使った書き方を排除するため、配列演算子(\*), reshape 関数、sum 関数、find 関数などを使い、ベクトル化した書き方で高速化を図っている。同様の処理の流れで、ro の解析を行う vssp\_ro\_get 関数も示す。取り込み方は、強度データを含むか含まないかの違いだけでほぼ変わらない。ちなみに、どちらの関数も得られたベクトルデータ r1,vind1,hrad1 から xyz 座標に変換するには、

```
vssp_cos_tblv=vssp_cos_tblv(:);  
vssp_sin_tblv=vssp_sin_tblv(:);  
xyz=[r1(:).*vssp_cos_tblv(vind1(:)).*cos(hrad1(:)),...  
r1(:).*vssp_cos_tblv(vind1(:)).*sin(hrad1(:)),...  
r1(:).*vssp_sin_tblv(vind1(:))]
```

とすると、xyz(:,1),xyz(:,2),xyz(:,3)で1ライン分のxyz座標が得られる。

## vssp\_ri\_get.m

```
% Copyright (c) 2014 Gerox http://www.gerox.com  
function  
[r1,r2,r3,r4,vind1,vind2,vind3,vind4,hrad1,hrad2,hrad3,hrad4,i1,i2,i3,i4,frame]=vssp_ri_get(vssp_data,vssp_tblh_ratio)  
vssp_data=vssp_data(25:end);  
byte2=2.^(0:8:8);  
%byte4=2.^(0:8:24);  
range_header_length = sum(vssp_data(1+(0:1)).*byte2);% 20 1:2(25:26)  
%line_head_timestamps_ms= sum(vssp_data(1+2+(0:3)).*byte4); % current_time1 3:6 (27:30)  
%line_tail_timestamps_ms= sum(vssp_data(1+6+(0:3)).*byte4); % current_time2 current_time2 >  
current_time1 % 7:10 (31:34)  
line_head_h_angle_ratio=  
sum(vssp_data(1+10+(0:1)).*byte2)-(bitand(vssp_data(12),128)==128)*2^16;%11:12 (35:36)  
line_tail_h_angle_ratio=  
sum(vssp_data(1+12+(0:1)).*byte2)-(bitand(vssp_data(14),128)==128)*2^16;% 13:14 (37:38)  
frame= double(vssp_data(1+14)); % 85 (39)  
%field= double(vssp_data(1+15)); % 2 (40)  
%line= sum(double(vssp_data(1+16+(0:1))).*byte2); % 32..33 17:18 (41:42)  
%spot= sum(vssp_data(1+18+(0:1)).*byte2); % 0% 19:20 (43:44)  
range_index_length= sum(vssp_data(1+0+range_header_length+(0:1)).*byte2); % 21:22 (45:46)  
nspots = sum(double(vssp_data(1+2+range_header_length+(0:1))).*byte2); % 23:24 (47:48)  
necho = sum(vssp_data(1+range_header_length+4+nspots*2+(0:1)).*byte2);% 173:174 (197:198)  
%[necho,nspots,necho>=nspots]  
if(necho < nspots)  
  
r1=[];r2=[];r3=[];r4=[];vind1=[];vind2=[];vind3=[];vind4=[];hrad1=[];hrad2=[];hrad3=[];hrad4=[];  
i1=[];i2=[];i3=[];i4=[];  
return;  
end  
data3=double(vssp_data(range_header_length+1+4+(0:nspots*2+1))); %25:174 (49:198)  
data3=reshape(data3,2,length(data3)/2)';  
index = (sum(data3.* [ones(length(data3),1)*byte2],2)+1)';  
index = index(1:75); % frequently nspots == 90, then we neglect 76-90 like C sample program  
implementation  
data4=double(vssp_data(range_header_length+range_index_length+1:end));% ??????????  
data4=reshape(data4,2,length(data4)/2)';
```

```

data4 = sum(data4.* (ones(length(data4),1)*byte2),2);
r1234 = data4(1:2:end);
i1234 = data4(2:2:end);
ind1 = index(1:end-1);r1=r1234(ind1);i1=i1234(ind1);
dind1=diff(ind1);
ind2 = ind1(dind1>1)+1;r2=r1234(ind2);i2=i1234(ind2);
ind3 = ind1(dind1>2)+2;r3=r1234(ind3);i3=i1234(ind3);
ind4 = ind1(dind1>3)+3;r4=r1234(ind4);i4=i1234(ind4);
hrad1 = (line_head_h_angle_ratio + (line_tail_h_angle_ratio -
line_head_h_angle_ratio)*vssp_tblh_ratio)*2*pi/65535;
vind1 = find(ind1);
vind2=find(dind1>1);hrad2 = hrad1(vind2);
vind3=find(dind1>2);hrad3 = hrad1(vind3);
vind4=find(dind1>3);hrad4 = hrad1(vind4);

```

vssp\_ro\_get.m

```

% Copyright (c) 2014 Gerox http://www.gerox.com
function
[r1,r2,r3,r4,vind1,vind2,vind3,vind4,hrad1,hrad2,hrad3,hrad4,frame]=vssp_ro_get(vssp_data,vssp_t
blh_ratio)
vssp_data=vssp_data(25:end);
byte2=2.^(0:8:8);
%byte4=2.^(0:8:24);
range_header_length = sum(vssp_data(1+(0:1)).*byte2);% 20 1:2(25:26)
%line_head_timestamps_ms= sum(vssp_data(1+2+(0:3)).*byte4); % current_time1 3:6 (27:30)
%line_tail_timestamps_ms= sum(vssp_data(1+6+(0:3)).*byte4); % current_time2 current_time2 >
current_time1 % 7:10 (31:34)
line_head_h_angle_ratio=
sum(vssp_data(1+10+(0:1)).*byte2)-(bitand(vssp_data(12),128)==128)*2^16;%11:12 (35:36)
line_tail_h_angle_ratio=
sum(vssp_data(1+12+(0:1)).*byte2)-(bitand(vssp_data(14),128)==128)*2^16;% 13:14 (37:38)
frame= double(vssp_data(1+14)); % 85 (39)
%field= double(vssp_data(1+15)); % 2 (40)
%line= sum(double(vssp_data(1+16+(0:1))).*byte2); % 32..33 17:18 (41:42)
%spot= sum(vssp_data(1+18+(0:1)).*byte2); % 0% 19:20 (43:44)
range_index_length= sum(vssp_data(1+0+range_header_length+(0:1)).*byte2); % 21:22 (45:46)
nspots = sum(double(vssp_data(1+2+range_header_length+(0:1))).*byte2); % 23:24 (47:48)
necho = sum(vssp_data(1+range_header_length+4+nspots*2+(0:1)).*byte2);% 173:174 (197:198)
%[necho,nspots,necho>=nspots]
if(necho < nspots)

r1=[];r2=[];r3=[];r4=[];vind1=[];vind2=[];vind3=[];vind4=[];hrad1=[];hrad2=[];hrad3=[];hrad4=[];
return;
end
data3=double(vssp_data(range_header_length+1+4+(0:nspots*2+1))); %25:174 (49:198)
data3=reshape(data3,2,length(data3)/2)';
index = (sum(data3.* (ones(length(data3),1)*byte2),2)+1)';
index = index(1:75); % frequently nspots == 90, then we neglect 76-90 like C sample program
implementation
data4=double(vssp_data(range_header_length+range_index_length+1:end));% ??????????
data4=reshape(data4,2,length(data4)/2)';
r1234 = sum(data4.* (ones(length(data4),1)*byte2),2);
ind1 = index(1:end-1);r1=r1234(ind1);
dind1=diff(ind1);
ind2 = ind1(dind1>1)+1;r2=r1234(ind2);
ind3 = ind1(dind1>2)+2;r3=r1234(ind3);
ind4 = ind1(dind1>3)+3;r4=r1234(ind4);
hrad1 = (line_head_h_angle_ratio + (line_tail_h_angle_ratio -
line_head_h_angle_ratio)*vssp_tblh_ratio)*2*pi/65535;
vind1 = find(ind1);
vind2 = find(dind1>1);hrad2 = hrad1(vind2+1);
vind3 = find(dind1>2);hrad3 = hrad1(vind3+1);
vind4 = find(dind1>3);hrad4 = hrad1(vind4+1);

```

## マルチエコーデータの扱い方

VSSP プロトコルの `_ri,_ro` 形式データでは、マルチエコーのインデックスの後に、距離データ、(あれば) 強度データが続く。距離データ、強度データは、エコーの数によりインデックス番号が異なるため、若干扱いが面倒である。ここでは、簡単な例を元にその考え方を示す。

今、`index=[1,3,7,9,11,14]`

があった場合を考える。この場合、マルチエコーインデックスとの関係は、つぎのようになる。

ind1	1		3				7		9		11			14
ind2		2		4				8		10		12		
ind3					5								13	
ind4						6								

つまり、距離データ配列 `r1234` では、

第一エコーの距離データが `r1234(1),r1234(3),r1234(7),r1234(9),r1234(11),r1234(14)`

第二エコーの距離データは、`r1234(2),r1234(4),r1234(8),r1234(10),r1234(12)` であり、

第三エコーの距離データは、`r1234(5),r1234(13)` であり、

第四エコーの距離データは、`r1234(6)`

となる。

`diff` 関数を使うと、`for` 文を使わないで、前後のデータの差分を計算し、エコーの数がわかる。

```
>> diff([1,3,7,9,11,14])
ans =     2     4     2     2     3
```

1番目が2つ、2番目が4つ、3番目が2つ、4番目が2つ5番目が3つのエコーデータがあることがわかる。つまり、

```
index=[1,3,7,9,11,14];
ind1 = index
dind1=diff(ind1);
ind2 = ind1(dind1>1)+1
ind3 = ind1(dind1>2)+2
ind4 = ind1(dind1>3)+3
```

として書ける。実行結果は、

```
ind1 =     1     3     7     9    11    14
ind2 =     2     4     8    10    12
```

```
ind3 = 5 13  
ind4 = 6
```

### 角度インデックスの計算方法

今度は、角度インデックス **vind** を求める。マルチエコーのデータも同じ角度であるので、角度との対応関係から書き直すと以下のようなになる。

vind	1	2	3	4	5	6
ind1	1	3	7	9	11	14
ind2	2	4	8	10	12	
ind3		5			13	
ind4		6				

つまり、

第一エコーのインデックス **ind1** の **vind** の値は、1,2,3,4,5,6 であり、

第二エコーのインデックス **ind2** の **vind** の値は、1,2,3,4,5

第三エコーのインデックス **ind3** の **vind** の値は、2,5

第四エコーのインデックス **ind4** の **vind** の値は、2

が対応する。find 関数を使うと次のようにして求めることができる。

```
index=[1,3,7,9,11,14];  
dind1=diff(ind1);  
vind1=find(ind1)  
vind2=find(dind1>1)  
vind3=find(dind1>2)  
vind4=find(dind1>3)
```

実行結果

```
vind1 = 1 2 3 4 5 6  
vind2 = 1 2 3 4 5  
vind3 = 2 5  
vind4 = 2
```

として抽出できる。

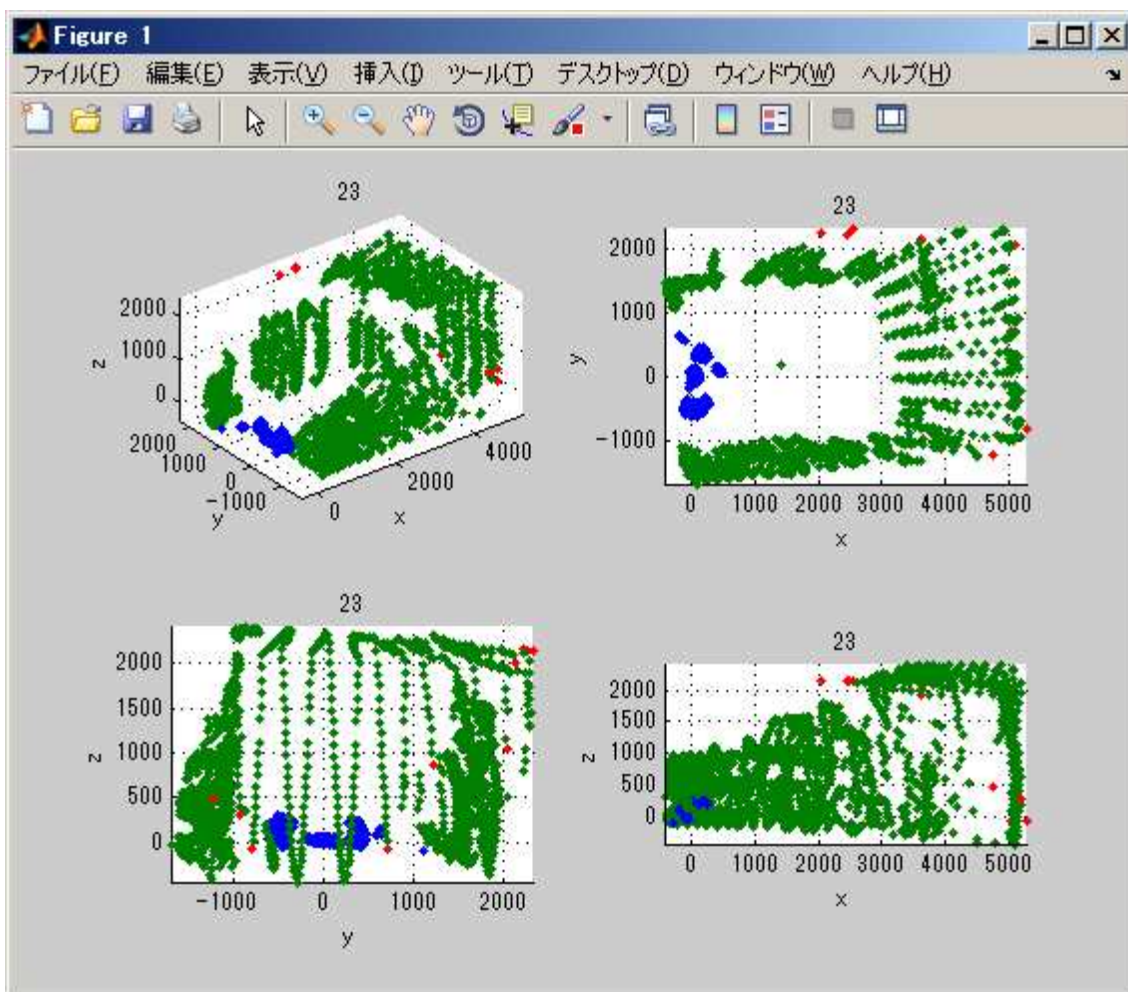
## vssp\_ri\_get.m、vssp\_ro\_get.m、vssp\_get.mを使ったlog解析プログラム

この解析プログラムは、北陽電機が提供している3Dビューアのログデータを読み込み表示するプログラムである。ファイルを指定し、読み込むことができる。ルックアップテーブルのvssp\_cos\_tblv、vssp\_tblh\_ratio、vssp\_tblh\_ratioは、本来GETコマンドにより取得するものであるが、ここでは、事前に取り込んだデータからテーブルを作成している。コメントを解除すれば、取り込みができるが、最初にGETコマンドによりtblh,tblvの取り込みを実行しないと、\_ri,\_roのデータの計算できないので注意が必要である。

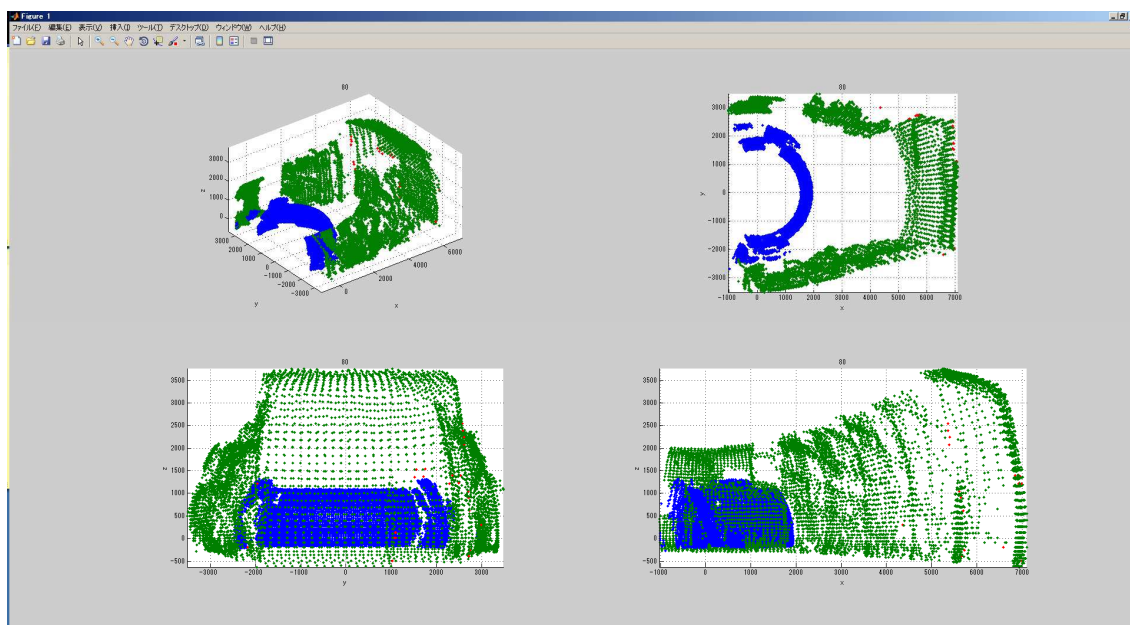
### vssp\_logview.m

```
% Copyright (c) 2014 Gerox http://www.gerox.com
close all;clear all;
%fid=fopen('2014-08-21-09-59-25-communication.log');
%fid=fopen('2014-08-26-16-34-05-communication.log');
%fid=fopen('2014-08-27-17-57-15-communication.log');
%fid=fopen('2014-08-27-18-57-27-communication.log');
[FileName,PathName] = uigetfile('*.log');fid=fopen([PathName,FileName]);
%data=fread(fid,200000);
%data=data(:)';
%fclose(fid);
rr1=[];rr2=[];rr3=[];rr4=[];vv1=[];vv2=[];vv3=[];vv4=[];hh1=[];hh2=[];hh3=[];hh4=[];
r1=1;r2=1;r3=1;r4=1;hrad1=1;hrad2=1;hrad3=1;hrad4=1;vind1=1;vind2=1;vind3=1;vind4=1;
pframe=0;frame=0;
vssp_cos_tblv=[1021/1057,1723/1796,1475/1549,1655/1752,473/505,1255/1352,1201/1306,1409/1547,186
6/2069,609/682,1290/1459,2160/2467,941/1085,654/761,1548/1817,1108/1311,224/267,1807/2180,1669/2
037,1307/1584,686/823,224/267,1108/1311,1548/1817,654/761,941/1085,2160/2467,1290/1459,609/682,1
866/2069,1409/1547,1201/1306,1255/1352,473/505,1655/1752,1475/1549,1723/1796,1021/1057,1317/1355
,2203/2254,881/897,1226/1243,587/593,1941/1955,1871/1880,1353/1357,624/625,1430/1431,5385/5386,3
48125/348126,11592/11593,2641/2642,1227/1228,743/744,524/525,1353/1357,791/794,1219/1222,743/744
,1227/1228,2641/2642,11592/11593,348125/348126,5385/5386,1430/1431,624/625,1353/1357,1871/1880,1
941/1955,587/593,1226/1243,881/897,2203/2254,1317/1355]';
vssp_sin_tblv=[635/2454,471/1669,1796/5881,5690/17341,282/805,411/1105,2649/6743,533/1291,435/10
07,343/762,804/1721,515/1066,913/1834,2102/4111,388/741,511/956,1348/2477,372/665,1877/3274,274/
485,437/791,1348/2477,511/956,388/741,2102/4111,913/1834,515/1066,804/1721,343/762,435/1007,533/
1291,2649/6743,411/1105,282/805,5690/17341,1796/5881,471/1669,635/2454,531/2258,459/2170,987/524
9,387/2348,400/2819,542/4537,306/3131,204/2659,237/4192,649/17361,466/24183,43/17940,-37/2817,-1
73/6288,-87/2156,-583/11245,-286/4635,-204/2659,-317/3650,-357/5098,-583/11245,-87/2156,-173/628
8,-37/2817,43/17940,466/24183,649/17361,237/4192,204/2659,306/3131,542/4537,400/2819,387/2348,98
7/5249,459/2170,531/2258]';
vssp_tblh_ratio=[0,95/8459,218/9699,553/16406,253/5630,159/2830,553/8203,397/5048,313/3482,550/5
439,171/1522,540/4369,547/4057,227/1554,941/5982,556/3299,313/1741,559/2764,529/2140,3829/13107,
941/2991,2363/7252,541/1605,1263/3626,1815/5048,538/1451,1781/4662,1516/3855,629/1555,2706/6509,
1327/3108,773/1764,4889/10878,1224/2657,941/1994,43/89,1099/2223,1124/2223,46/89,1053/1994,1433/
2657,6038/10967,991/1764,1781/3108,3855/6598,926/1555,2339/3855,2881/4662,913/1451,3233/5048,236
3/3626,1064/1605,4889/7252,2050/2991,1263/1813,1815/2524,1781/2331,7543/9324,2743/3299,5041/5982
,1327/1554,3510/4057,3829/4369,1351/1522,4889/5439,3169/3482,1523/1653,7650/8203,2671/2830,1764/
1847,86/89,2392/2447,2113/2137,1];
viewagl=[-37.5,30;0,90;90,0;0,0];
data=[];
while(1)
    for i=1:1000
        ind=strfind(char(data),'VSSP');
        if(length(ind) > 36) break;end
        data0=fread(fid,10940);
        data0=data0(:)';
        data=double([data,data0]);
    end
    if(length(strfind(data,'VSSP'))<3) break;end
    [data,vssp_data,vssp_type,vssp_status]=vssp_get(data);
    if(strncmp(vssp_status,'000',3))
        if(strncmp(vssp_type,'_ri',3))
            [r1,r2,r3,r4,vind1,vind2,vind3,vind4,hrad1,hrad2,hrad3,hrad4,i1,i2,i3,i4,frame]=vssp_ri_get(vssp
_data,vssp_tblh_ratio);
```

```
elseif(strncmp(vssp_type, '_ro', 3))
[r1,r2,r3,r4,vind1,vind2,vind3,vind4,hrad1,hrad2,radh3,hrad4,frame]=vssp_ro_get(vssp_data,vssp_t
blh_ratio);
% elseif(strncmp(vssp_type, 'GET', 3))
% vssp_data=vssp_data(25:end);
% if(strncmp(char(vssp_data), ['GET:tblv',10], 9))
% fprintf('GET:tblv -> [rad]¥n');
% ind=strfind(vssp_data, ',');
% data0=vssp_data(1,ind(1)-4:end);
% len=length(data0);
% data0=double(reshape(data0,5,len/5));
% vssp_tblv_rad=hex2dec(char(data0(:,1:4)))'*2*pi/65535;
% vssp_sin_tblv = sin(vssp_tblv_rad);
% vssp_cos_tblv = cos(vssp_tblv_rad);
% elseif(strncmp(char(vssp_data), ['GET:tblh',10], 9))
% fprintf('GET:tblh -> [rad]¥n');
% ind=strfind(vssp_data, ',');
% data0=vssp_data(1,ind(1)-4:end);
% len=length(data0);
% data0=double(reshape(data0,5,len/5));
% vssp_tblh_ratio=hex2dec(char(data0(:,1:4)))/65535;
% end
end
if(pframe ~= frame)
if((length(rr1)~=length(vv1))||((length(vv1)~=length(hh1))||((length(vv1)~=length(rr1))))
else
for i=1:4
subplot(2,2,i)
plot3(rr1.*vssp_cos_tblv(vv1).*cos(hh1),rr1.*vssp_cos_tblv(vv1).*sin(hh1),rr1.*vssp_sin_tblv(vv1
),'.');
title(frame);
hold all
plot3(rr2.*vssp_cos_tblv(vv2).*cos(hh2),rr2.*vssp_cos_tblv(vv2).*sin(hh2),rr2.*vssp_sin_tblv(vv2
),'.');
plot3(rr3.*vssp_cos_tblv(vv3).*cos(hh3),rr3.*vssp_cos_tblv(vv3).*sin(hh3),rr3.*vssp_sin_tblv(vv3
),'.');
plot3(rr4.*vssp_cos_tblv(vv4).*cos(hh4),rr4.*vssp_cos_tblv(vv4).*sin(hh4),rr4.*vssp_sin_tblv(vv4
),'.');
hold off;axis equal;grid on;xlabel('x');ylabel('y');zlabel('z');view(viewagl(i,:));
end
drawnow;
pframe = frame;
end
rr1=[];rr2=[];rr3=[];rr4=[];vv1=[];vv2=[];vv3=[];vv4=[];hh1=[];hh2=[];hh3=[];hh4=[];
else
rr1 = [rr1;r1(:)];vv1 = [vv1;vind1(:)];hh1 = [hh1;hrad1(:)];
rr2 = [rr2;r2(:)];vv2 = [vv2;vind2(:)];hh2 = [hh2;hrad2(:)];
rr3 = [rr3;r3(:)];vv3 = [vv3;vind3(:)];hh3 = [hh3;hrad3(:)];
rr4 = [rr4;r4(:)];vv4 = [vv4;vind4(:)];hh4 = [hh4;hrad4(:)];
end
end
end
fclose(fid);
```



ログファイルを読み込んだ実行例



インターレースキャンモードのデータの読み込み表示例



## pnet 関数を使ったリアルタイムビューア

### pnet 関数とは？

MATLAB Central で公開されている、tcp\_udp\_ip toolbox 中の MEX 関数である。現在のバージョンは、2.0.6 である。32bit,64bit とともに安定し TCP/UDP 通信プログラムが作れるためここでもリアルタイム取り込み用として使用する。ちなみに、自分でコンパイルするには、例えば、Windows の場合には、

```
mex -O pnet.c ws2_32.lib -DWIN32
```

```
mex -O pnet.c ws2_32.lib -DWIN64
```

とする。32 ビット、64 ビットなど開発環境の違いにより使い分ける必要がある。できた MEX ファイルを同じ現在フォルダーの中に入れておく。

以下に pnet 関数を使った、リアルタイム取り込み表示プログラム vssp\_realview.m を示す。

ここで示す、vssp\_realview.m では、以下のオプション変更ができる。

cmd='ri'; %cmd='ro';に変更すると、強度データの取り込みもできる。

また、この例では、インターレーススキャンモードとして

```
pnet(con,'write',['SET:_itl=0,04',10]);
```

設定しているが、通常のスキャンモードにする場合には、

```
pnet(con,'write',['SET:_itl=0,01',10]);
```

とする。この場合、更新は速くなるがデータ点数は粗くなる。

### vssp\_realview.m

```
% Copyright (c) 2014 Gerox http://www.gerox.com
close all;clear all;
con = -1;
for i=1:100
    con=pnet('tcpconnect','192.168.0.10',10940);
    pause(1);
    if(con ~= -1) break;
    else fprintf('Connection Waiting[%d]¥n',100 - i);end
end
if(i == 100) fprintf('Time out error10¥n');
else fprintf('Connected!¥n');end
pnet(con,'read','noblock',10940,'setreadtimeout',5000,'setwritetimeout',5000);
cmd='ri';%cmd='ro';
%%%%%%%%%%
while(1)
    pause(0.5);
    data0=pnet(con,'read','noblock',10940);
    pnet(con,'write',['DAT:ri=0',10]);
    pnet(con,'write',['DAT:ax=0',10]);
    timeout = 2000;
    for i=1:timeout
        pause(1);
        data0=pnet(con,'read','noblock',10940);
        char(data0)
        if isempty(data0);break;end
    end
end
%%%%%%%%%%
```

```
pnet(con, 'write', ['SET:_itl=0,04',10]);
data=[];
for i=1:timeout
    data0=pnet(con, 'read', 'noblock',10940);
    data=[data,data0];
    LF=sum(data==10);
    if LF > 1 ;break;
    elseif LF > 2;i = timeout;break;
end
pause(0.01);
end
char(data)
if(i==timeout);fprintf('time out error38');continue;end
if 1

vssp_cos_tblv=[1021/1057,1723/1796,1475/1549,1655/1752,473/505,1255/1352,1201/1306,1409/1547,186
6/2069,609/682,1290/1459,2160/2467,941/1085,654/761,1548/1817,1108/1311,224/267,1807/2180,1669/2
037,1307/1584,686/823,224/267,1108/1311,1548/1817,654/761,941/1085,2160/2467,1290/1459,609/682,1
866/2069,1409/1547,1201/1306,1255/1352,473/505,1655/1752,1475/1549,1723/1796,1021/1057,1317/1355
,2203/2254,881/897,1226/1243,587/593,1941/1955,1871/1880,1353/1357,624/625,1430/1431,5385/5386,3
48125/348126,11592/11593,2641/2642,1227/1228,743/744,524/525,1353/1357,791/794,1219/1222,743/744
,1227/1228,2641/2642,11592/11593,348125/348126,5385/5386,1430/1431,624/625,1353/1357,1871/1880,1
941/1955,587/593,1226/1243,881/897,2203/2254,1317/1355]';

vssp_sin_tblv=[635/2454,471/1669,1796/5881,5690/17341,282/805,411/1105,2649/6743,533/1291,435/10
07,343/762,804/1721,515/1066,913/1834,2102/4111,388/741,511/956,1348/2477,372/665,1877/3274,274/
485,437/791,1348/2477,511/956,388/741,2102/4111,913/1834,515/1066,804/1721,343/762,435/1007,533/
1291,2649/6743,411/1105,282/805,5690/17341,1796/5881,471/1669,635/2454,531/2258,459/2170,987/524
9,387/2348,400/2819,542/4537,306/3131,204/2659,237/4192,649/17361,466/24183,43/17940,-37/2817,-1
73/6288,-87/2156,-583/11245,-286/4635,-204/2659,-317/3650,-357/5098,-583/11245,-87/2156,-173/628
8,-37/2817,43/17940,466/24183,649/17361,237/4192,204/2659,306/3131,542/4537,400/2819,387/2348,98
7/5249,459/2170,531/2258]';

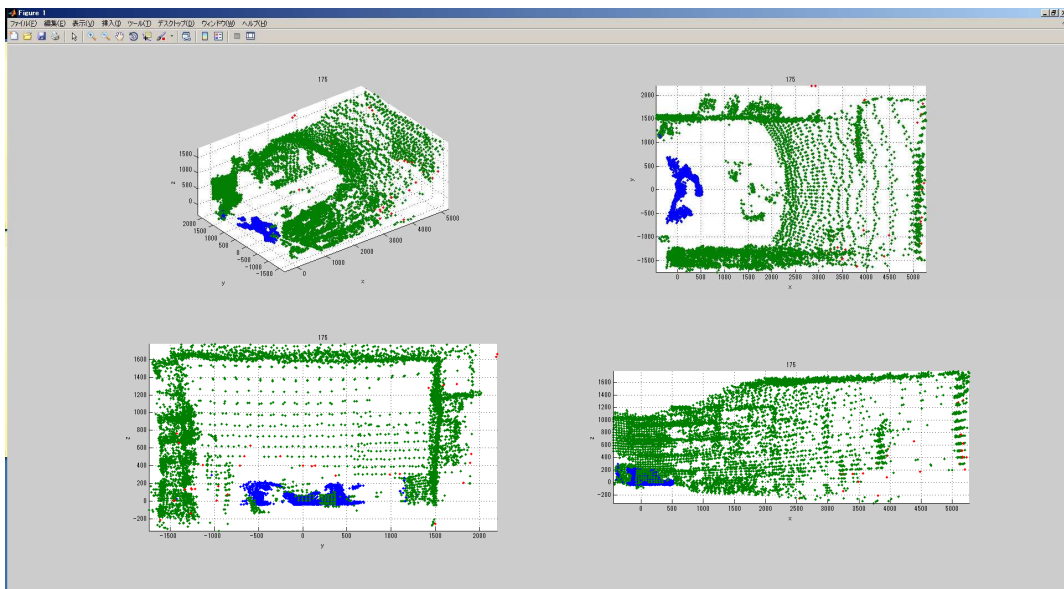
vssp_tblh_ratio=[0,95/8459,218/9699,553/16406,253/5630,159/2830,553/8203,397/5048,313/3482,550/5
439,171/1522,540/4369,547/4057,227/1554,941/5982,556/3299,313/1741,559/2764,529/2140,3829/13107,
941/2991,2363/7252,541/1605,1263/3626,1815/5048,538/1451,1781/4662,1516/3855,629/1555,2706/6509,
1327/3108,773/1764,4889/10878,1224/2657,941/1994,43/89,1099/2223,1124/2223,46/89,1053/1994,1433/
2657,6038/10967,991/1764,1781/3108,3855/6598,926/1555,2339/3855,2881/4662,913/1451,3233/5048,236
3/3626,1064/1605,4889/7252,2050/2991,1263/1813,1815/2524,1781/2331,7543/9324,2743/3299,5041/5982
,1327/1554,3510/4057,3829/4369,1351/1522,4889/5439,3169/3482,1523/1653,7650/8203,2671/2830,1764/
1847,86/89,2392/2447,2113/2137,1];
else
    pnet(con, 'write', ['GET:tblv',10]);
    data=[];
    for i=1:timeout
        data0=pnet(con, 'read', 'noblock',10940);
        data=[data,data0];
        LF=sum(data==10);
        if LF > 1;break;
        elseif LF > 2;i = timeout;break;
    end
    pause(0.01);
end
if(i==timeout);fprintf('time out error52\n');continue;
else
    ind=strfind(data, ',');
    data=data(1,ind(1)-4:end);
    len=length(data);
    data=double(reshape(data,5,len/5)');
    vssp_tblv_rad=hex2dec(data(:,1:4))*2*pi/65535;
    vssp_sin_tblv = sin(vssp_tblv_rad)';
    vssp_cos_tblv = cos(vssp_tblv_rad)';
end
%%%%%%
pnet(con, 'write', ['GET:tblh',10]);
data=[];
for i=1:timeout
    data0=pnet(con, 'read', 'noblock',10940);
    data=[data,data0];
    LF=sum(data==10);
```

```
        if LF > 1;break;
        elseif LF > 2;i == timeout;break;
        end
        pause(0.01);
    end
    if(timeout==i);fprintf('time out error76');continue;
    else
        ind=strfind(data,',');
        data=data(1,ind(1)-4:end);
        len=length(data);
        data=double(reshape(data,5,len/5)');
        vssp_tblh_ratio=hex2dec(data(:,1:4)')/65535;
    end
end

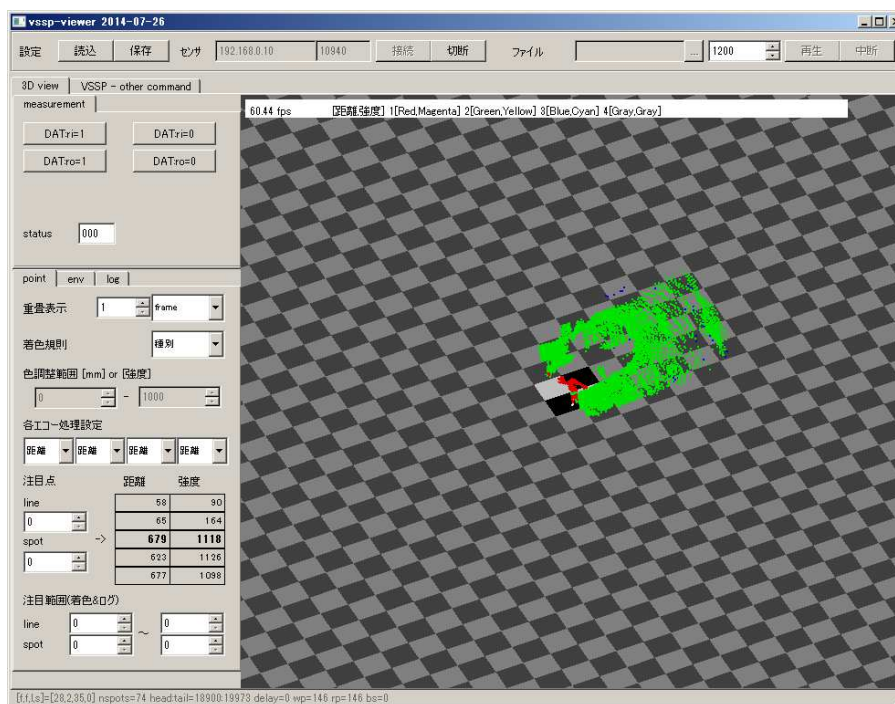
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
pnet(con,'write',['DAT:',cmd,'=1',10]);
data=[];
for i=1:timeout
    data0=pnet(con,'read','noblock',10940);
    data=[data,data0];
    LF=sum(data==10);
    if LF > 1;break;
    elseif LF > 2;i = timeout;break;
    end
    pause(0.01);
end
if(timeout==i);fprintf('time out error100');continue;
else break;end
pause(0.05);
end
rr1=[];vv1=[];hh1=[];rr2=[];vv2=[];hh2=[];rr3=[];vv3=[];hh3=[];rr4=[];vv4=[];hh4=[];
r1=1;r2=1;r3=1;r4=1;hrad1=1;hrad2=1;hrad3=1;hrad4=1;vind1=1;vind2=1;vind3=1;vind4=1;
pframe=0;frame=0;
figure
timeout=600000;
data=[];
viewagl=[-37.5,30;0,90;90,0;0,0];
while(1)
    for i=1:timeout
        ind=strfind(char(data),'VSSP');
        if(length(ind) > 36) break;end
        data0=pnet(con,'read','noblock',10940);
        data0= double(data0)-[uint16(data0)>127].*2^16;
        data0= double(data0+[data0<0]*2^8);
        data=double([data,data0]);
    end
    if(i == timeout) fprintf('time out error111');data=[];continue;end
    [data,vssp_data,vssp_type,vssp_status]=vssp_get(data);
    if(strncmp(vssp_status,'000',3))
        if(strncmp(vssp_type,'_ri',3))
            [r1,r2,r3,r4,vind1,vind2,vind3,vind4,hrad1,hrad2,hrad3,hrad4,i1,i2,i3,i4,frame]=vssp_ri_get(vssp
            _data,vssp_tblh_ratio);
            elseif(strncmp(vssp_type,'_ro',3))
                [r1,r2,r3,r4,vind1,vind2,vind3,vind4,hrad1,hrad2,hrad3,hrad4,frame]=vssp_ro_get(vssp_data,vssp_t
                blh_ratio);
            end
            if(pframe ~= frame)
                for i=1:4
                    subplot(2,2,i)
                    plot3(rr1.*vssp_cos_tblv(vv1).*cos(hh1),rr1.*vssp_cos_tblv(vv1).*sin(hh1),rr1.*vssp_sin_tblv(vv1
                    ),'.');
                    hold all
                    plot3(rr2.*vssp_cos_tblv(vv2).*cos(hh2),rr2.*vssp_cos_tblv(vv2).*sin(hh2),rr2.*vssp_sin_tblv(vv2
                    ),'.');
```

```
plot3(rr3.*vssp_cos_tblv(vv3).*cos(hh3),rr3.*vssp_cos_tblv(vv3).*sin(hh3),rr3.*vssp_sin_tblv(vv3),'.');

plot3(rr4.*vssp_cos_tblv(vv4).*cos(hh4),rr4.*vssp_cos_tblv(vv4).*sin(hh4),rr4.*vssp_sin_tblv(vv4),'.');
    hold                                off;axis                                equal;grid
on;title(frame);xlabel('x');ylabel('y');zlabel('z');view(viewagl(i,:));
    end
drawnow;
pframe = frame;
rr1=[];vv1=[];hh1=[];rr2=[];vv2=[];hh2=[];rr3=[];vv3=[];hh3=[];rr4=[];vv4=[];hh4=[];
else
    rr1 = [rr1;r1(:)];vv1 = [vv1;vind1(:)];hh1 = [hh1;hrad1(:)];
    rr2 = [rr2;r2(:)];vv2= [vv2;vind2(:)];hh2 = [hh2;hrad2(:)];
    rr3 = [rr3;r3(:)];vv3 = [vv3;vind3(:)];hh3 = [hh3;hrad3(:)];
    rr4 = [rr4;r4(:)];vv4= [vv4;vind4(:)];hh4 = [hh4;hrad4(:)];
end
end
end
```



青が第一エコー、緑が第二エコー、赤が第三エコーである。



同じ位置で、vssp-viewer を使って見た場合の例

### 演習問題

1. このプログラムでは、ax 形式のデータに対応するようにプログラムを変更せよ。
2. GUI を使って変更可能な、SET, GET, DAT, PNG, VER コマンドをサポートするプログラムを作成せよ。

実際の取り込み例

