Get 2D Homography of a projective geometry transform

- Author: WANG Lei,USTB
- Link: https://github.com/shidafu/ViewConeCalibration.git
- Date:2016/3/3
- Algorithom:

Get *s* and [*H*] From:

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

By solving:

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -u_1x_1 & -u_1y_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -v_1x_1 & -v_1y_1 \\ & & & & \vdots & & & \\ x_n & y_n & 1 & 0 & 0 & 0 & -u_nx_n & -u_ny_n \\ 0 & 0 & 0 & x_n & y_n & 1 & -v_nx_n & -v_ny_n \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix} = \begin{bmatrix} u_1 \\ v_1 \\ \vdots \\ u_n \\ v_n \end{bmatrix}$$

$$[H_{list}] = \left([XYUV_{list}]^{\mathrm{T}} \cdot [XYUV_{list}]\right)^{-1} \cdot [XYUV_{list}]^{\mathrm{T}} \cdot [UV_{list}]$$

$$s = [H] \cdot [XY1] \cdot [UV1]^{\mathrm{T}} \cdot \left([UV1] \cdot [UV1]^{\mathrm{T}}\right)$$

- Inputs:

```
XY----cordNum by pointNum matrix,
             cordNum==2,pointNum>=4,
             [x1,x2,...;
              y1,y2,...]
UV----cordNum by pointNum matrix,
             cordNum==2,pointNum>=4,
             [u1,u2,...;
              v1,v2,...]
```

- Outputs:

```
           H----3 by 3 matrix: [h11 h12 h13;
                                h21 h22 h23;
                                h31 h32  1]
           s----projective para
```

```
function [H,s] = GetHomography2D(XY,UV)
% Initial
[cordNum, pointNum]=size(XY);
if ~(cordNum==2 || cordNum==3)
    error('Input matrix size error!');
end
XY1=ones(3,pointNum,'double');
XY1(1:2,:)=XY(1:2,:);
UV1=ones(3,pointNum,'double');
UV1(1:2,:)=UV(1:2,:);
HList=zeros(8,1,'double');
XYUVList=zeros(2*pointNum,8,'double');
UVList=zeros(2*pointNum,1,'double');
for i=1:pointNum
    XYUVList(i*2-1,:)=[XY(1,i), XY(2,i), 1, 0, 0, 0,
 -1*UV(1,i)*XY(1,i), -1*UV(1,i)*XY(2,i)];
    XYUVList(i*2,:)=[0, 0, 0, XY(1,i), XY(2,i), 1, -1*UV(2,i)*XY(1,i),
 -1*UV(2,i)*XY(2,i)];
    UVList(i*2-1,:)=UV(1,i);
    UVList(i*2,:)=UV(2,i);
end
% Algorithm
HList=inv(XYUVList'*XYUVList)*XYUVList'*UVList;
% Set outputs
H(1,:)=HList(1:3,1);
H(2,:)=HList(4:6,1);
H(3,1:2)=HList(7:8,1);
H(3,3)=1;
s=H*XY1*UV1'*inv(UV1*UV1');
```

```
Error using GetHomography2D (line 62)
Not enough input arguments.
```

Published with MATLAB® R2015a