

Implementation of Homography in MATLAB

Mahnoor Anjum
Registration Number 132937
Bachelors in Electrical Engineering (ongoing)
CS-475 Computer Vision, BEE7
Instructor: Dr. Wajahat Hussain

CODE

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%       Author: Mahnoor Anjum
%       Instructor: Dr. Wajahat Hussain
%       Date Created: 10/21/2018
%       Date Submitted: 10/22/2018
%       Comments: -/
%       Acknowledgements:
%           This assignment is my original work.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

clc
clear all
close all

% Step1: Import the input Image
input = imread('src.jpg');

% Step2: Show the input image in MATLAB
figure, imshow(input)

% Step3: Input the Original Homography Matrix
Given_H = [0.0026 -0.0030 0.4656;
            0.0027 0.0036 -1.7067;
            0.0001 0.0009 0.2560 ];
% Step4: Warp the original image based on this homography

tform_orig = projective2d(Given_H');
output_dlt_orig = imwarp(input, tform_orig);
figure, imshow(output_dlt_orig)

% Step5: A good programmer creates re-usable code, Let's define the
% number of points we need for this particular assignment.
npoints = 4

% Step6: Creating the matrix of input points
xi = [195.0000 326.0000 1;
      299.0000 248.0000 1;
      412.0000 321.0000 1;
      306.0000 422.0000 1]
```

```
% Step7: Scaling factor for a better visual output
integer=1;
```

```
% Step8: Creating the ouput points matrix
xi_ = [0 0 1 ;
       integer 0 1;
       integer integer 1;
       0 integer 1]
```

```
A = zeros(2*npoints, 9)
```

```
k = 1
for i = 1:2:8
    xi_s = xi_(k,:);
    x = xi_s(1);
    y = xi_s(2);
    w = xi_s(3);
    A(i,4:6) = -w*xi(k,:);
    A(i,7:9) = y*xi(k,:);
    A(i+1,1:3) = w*xi(k,:);
    A(i+1,7:9) = -x*xi(k,:);
    k = k+1;
end;
```

```
if npoints==4
    H = null(A);
else
    [U,S,V] = svd(A);
    H = V(:,9);
end;
```

```
%Visualizing the warp
H_shaped = reshape(H,3,3);
tform = projective2d(H_shaped);
output_dlt = imwarp(input, tform);
figure, imshow(output_dlt)
```

```
%Verify
x = H_shaped * xi';
x = x';
for i=1:4
    x(i,:) = x(i,+)/x(i,3);
end
x
Given_H
H_shaped
```

OUTPUT

x =

```
-0.0030 -0.0013  1.0000
-0.0051 -0.0000  1.0000
-0.0054  0.0002  1.0000
-0.0033 -0.0011  1.0000
```

Given_H =

```
0.0026 -0.0030  0.4656
0.0027  0.0036 -1.7067
0.0001  0.0009  0.2560
```

H_shaped =

```
0.0014  0.0015  0.0001
-0.0017  0.0020  0.0005
0.2605 -0.9548  0.1432
```

SIMULATION _ GUI

(code can be found at github.com/mahnooranjum)





