Homography estimation via DLT

Estimate an homography via DLT References: Hartley Zisserman, Multiview Geometry. Chapter 4

.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.

Image Analysis and Computer Vision Politecnico di Milano

Luca Magri for comments and suggestions please send an email to luca.magri@polimi.it

.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.\\°//.

Contents

Let's take two picture of a planar scene

```
im1 = imread('E3_data/ella1.jpg');
im2 = imread('E3_data/ella2.jpg');
im1 = imresize(im1,0.6);
im2 = imresize(im2,0.6);
figure;
subplot(1,2,1);
imshow(im1);
subplot(1,2,2);
imshow(im2);
```

Homography transformation

the image coordinates that lie on the same plane of these two images are related by an **homography**: Without loss of generality, we can assume that the point on the plane have coordinates M = [x,y, 0,1], and are mapped via P1 and P2 to m1 = P1*M and m2 =P2*M, where Pi = Ki[Ri,ti] and Ri = [Ri1,Ri2,Ri3]. Having z=0 implies that the coordinates mi = Ki[Ri1,Ri2,ti] [x,y,1] are determined by a 3x3 non singular transformation Hi. The realtion between {m1<->m2} is hence given by H = H1^-1*H2. Our aim is to determine such H.

Identify four pairs of corresponding points

```
m1 = nan(3,4);
m2 = nan(3,4);

figure;
imshow(im1);
hold on;
[x, y] = getpts();
m1(:,1) = [x(1); y(1); 1];
m1(:,2) = [x(2); y(2); 1];
m1(:,3) = [x(3); y(3); 1];
m1(:,4) = [x(4); y(4); 1];

FNT_SZ = 20;
plot(m1(1,1), m1(2,1),'o', 'Color','r');
plot(m1(1,2), m1(2,2),'o','Color', 'r');
plot(m1(1,3), m1(2,3),'o','Color', 'r');
plot(m1(1,4), m1(2,4),'o','Color', 'r');
text(m1(1,1), m1(2,1), 'a', 'FontSize', FNT_SZ, 'Color', 'r');
```

```
text(m1(1,2), m1(2,2), 'b', 'FontSize', FNT_SZ, 'Color', 'r');
text(m1(1,3), m1(2,3), 'c', 'FontSize', FNT_SZ, 'Color', 'r');
text(m1(1,4), m1(2,4), 'd', 'FontSize', FNT_SZ, 'Color', 'r');
```


```
figure;
imshow(im2);
hold on;
[x, y] = getpts();
m2(:,1) = [x(1); y(1); 1];
m2(:,2) = [x(2); y(2); 1];
m2(:,3) = [x(3); y(3); 1];
m2(:,4) = [x(4); y(4); 1];

FNT_SZ = 20;
plot(m2(1,1), m2(2,1),'o', 'Color','r');
plot(m2(1,2), m2(2,2),'o','Color', 'r');
plot(m2(1,3), m2(2,3),'o','Color', 'r');
plot(m2(1,4), m2(2,4),'o','Color', 'r');
text(m2(1,1), m2(2,1), 'a''', 'FontSize', FNT_SZ, 'Color', 'r');
text(m2(1,2), m2(2,2), 'b''', 'FontSize', FNT_SZ, 'Color', 'r');
text(m2(1,3), m2(2,3), 'c''', 'FontSize', FNT_SZ, 'Color', 'r');
text(m2(1,4), m2(2,4), 'd''', 'FontSize', FNT_SZ, 'Color', 'r');
```


ELLA AT THE
HOLLYWOOD BOWL
THE IRVING BERLIN SONGBOOK
ELLA AT THE HOLLYWOOD BOWL: The Irving Berlin Songbook
a
b
d


d'
c'
a'
b'
ELLA AT THE HOLLYWOOD BOWL
THE IRVING BERLIN SONGBOOK

Direct linear transform

Hartley & Zisserman Alg 4.2 page 109 in 2nd edition

We want to estimate the homography H = [h1^t; h2^t;h3^t] that maps each mi=[xi;yi;wi] in mi'=[xi';yi';wi']. Since mi and mi' are in homogenous coordinates, mi and mi' represents the same points if exists a lambda different from zero such taht Hmi = lambda*mi'. This condition can be expressed using the cross product as mi'xHmi = 0. This equation be written as yi'*h3'+xi - wi* h2'*xi =0 wi'*h1'*xi - xi'*h3'*xi =0 xi'*h2'*xi - yi'*h1'*xi =0 So for each pairs of corresponding points we have three equations. But only two of the trhee are linear independent. So each correspondence provide 2 constraints. Four points are enough to constraints the 8 d.o.f of H (= 3x3 - 1 to account for the scale).

```
% Build the  matrix of the linear homogeneous system
% A vec(H) = 0
num_points = 4;
A = zeros(2*num_points,9);
% the rows are the number of independent constraints
% the columns are the number of unknown (9 for a 3x3 matrix)
for i =1:num_points
    p1= m1(:,i);
    p2 = m2(:,i);
    A(2*i-1,:) = [zeros(1,3), p1'*p2(3), - p1'*p2(2)];
    A(2*i,:) = [p2(3)*p1', zeros(1,3), -p1'*p2(1)];
end

% now we have to solve an homogeneous system: svd!
[u,s,v] = svd(A);
% let's have a look at how well the system is conditioned
s = diag(s);
% rough estimate of the nullspace dimension
nullspace_dimension = sum(s < eps * s(1) * 1e3);
if (nullspace_dimension > 1)
  fprintf('Nullspace is too large to accomodate a single solution...\n');
end
h = v(:,end);
H = reshape(h,3,3)';
H = H./norm(H);
% let's factorize this code into a function dlt_homography

%save('ella_homo.mat','H','m1','m2');
```

Showing the mapping on the points

```
Hm1 = H*m1;
% dehomogenize points
Hm1 = Hm1./repmat(Hm1(3,:),3,1);
figure;
imshow(im2);
hold all;
MRK_SZ = 300;
scatter(m2(1,:),m2(2,:),MRK_SZ,'ro','filled');
scatter(Hm1(1,:),Hm1(2,:),MRK_SZ,'y+','LineWidth',4);
```

Rendering the results

```
t = maketform( 'projective', H');
J = imtransform(im1,t);


figure;
subplot(1,3,1);
imshow(im1);
title('im1')
subplot(1,3,2);
imshow(J);
title('transformed im1')
subplot(1,3,3);
imshow(im2);
title('im2')
```

Degenerate configurations - collinearity

now let's repeat the same procedure but this time we select 4 collinears point

```
m1 = nan(3,4);
m2 = nan(3,4);

% select points on the first image
figure;
imshow(im1);
hold on;
[x, y] = getpts();
m1(:,1) = [x(1); y(1); 1];
m1(:,2) = [x(2); y(2); 1];
m1(:,3) = [x(3); y(3); 1];
m1(:,4) = [x(4); y(4); 1];

FNT_SZ = 20;
plot(m1(1,1), m1(2,1),'o', 'Color','r');
plot(m1(1,2), m1(2,2),'o','Color', 'r');
plot(m1(1,3), m1(2,3),'o','Color', 'r');
plot(m1(1,4), m1(2,4),'o','Color', 'r');
text(m1(1,1), m1(2,1), 'a', 'FontSize', FNT_SZ, 'Color', 'r');
text(m1(1,2), m1(2,2), 'b', 'FontSize', FNT_SZ, 'Color', 'r');
text(m1(1,3), m1(2,3), 'c', 'FontSize', FNT_SZ, 'Color', 'r');
text(m1(1,4), m1(2,4), 'd', 'FontSize', FNT_SZ, 'Color', 'r');
```

select points on the second image

```
figure;
imshow(im2);
hold on;
[x, y] = getpts();
m2(:,1) = [x(1); y(1); 1];
m2(:,2) = [x(2); y(2); 1];
m2(:,3) = [x(3); y(3); 1];
m2(:,4) = [x(4); y(4); 1];

FNT_SZ = 20;
plot(m2(1,1), m2(2,1),'o', 'Color','r');
plot(m2(1,2), m2(2,2),'o','Color', 'r');
plot(m2(1,3), m2(2,3),'o','Color', 'r');
plot(m2(1,4), m2(2,4),'o','Color', 'r');
text(m2(1,1), m2(2,1), 'a''', 'FontSize', FNT_SZ, 'Color', 'r');
text(m2(1,2), m2(2,2), 'b''', 'FontSize', FNT_SZ, 'Color', 'r');
text(m2(1,3), m2(2,3), 'c''', 'FontSize', FNT_SZ, 'Color', 'r');
text(m2(1,4), m2(2,4), 'd''', 'FontSize', FNT_SZ, 'Color', 'r');
```


ELLA
AT THE
HOLLYWOOD BOWL
THE IRVING
BERLIN SONGBOOK
ELLA AT THE HOLLYWOOD BOWL: The Irving Berlin Songbook
a
b
c
d


a'
b'
c'
d'

compute the homography

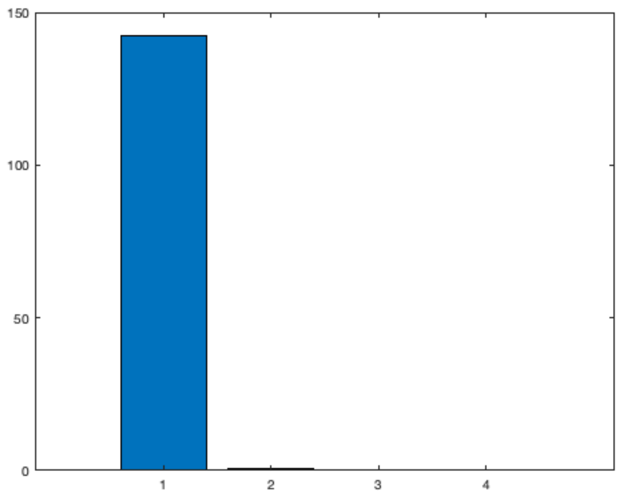
```
[H,A_collinear] = dlt_homography(m1,m2);

Hm1 = H*m1;
% dehomogenize points
Hm1 = Hm1./repmat(Hm1(3,:),3,1);
figure;
imshow(im2);
hold all;
MRK_SZ = 300;
scatter(m2(1,:),m2(2,:),MRK_SZ,'ro','filled');
scatter(Hm1(1,:),Hm1(2,:),MRK_SZ,'y+');

% the condition number of A is much larger than 1, the matrix is sensitive to the inverse calculation.
% let's see what happe to the nullspace of A when points are collinear
cond(A)
cond(A_collinear);
[~,s_collinear,~]=svd(A_collinear);
s_collinear = diag(s_collinear);
figure;
bar([s(end-1),s_collinear(end-1),s(end),s_collinear(end)])
```

```
ans =

   1.3067e+07
```



let's show the result

```
t = maketform( 'projective', H');
J = imtransform(im1,t);


figure;
subplot(1,2,1);
imshow(im2);
subplot(1,2,2);
imshow(J);
```

```
Warning: The condition number of A is 1604493683560.505.
```