SVD based Relative Orientation from Planar Scenes

Bill Triggs

INRIA Rhône-Alpes, 655 avenue de l'Europe, 38330 Montbonnot St. Martin, France.
Bill.Triggs@inrialpes.fr ◇ *http://www.inrialpes.fr/movi/people/Triggs*

This note describes a simple method for the relative orientation of two calibrated cameras from 4 or more 3D points which are coplanar, but otherwise unknown. The procedure is very similar to Wunderlich's 1982 method, but is based on Singular Value Decomposition (SVD) of the inter-image homography $\boldsymbol{H}$ rather than eigen-decomposition of $\boldsymbol{H}^\top\boldsymbol{H}$. This should make it a little stabler for difficult cases such as very near or very distant planes (relative to the camera translation). As with any plane based method, even if 'twisted pair' solutions are eliminated by a sign test, there are almost always two possible solutions for the orientation and 3D structure. Both are internally self-consistent and have small epipolar residual. Visibility constraints are sometimes enough to eliminate the false one, but more often additional, external information is required.

We use homogeneous, calibrated, lens-distortion-corrected image coordinates, *i.e.* image points are represented by *bona fide* 3D direction vectors $\boldsymbol{x}$, $\boldsymbol{x}'$ with respect the unknown 3D frames of the two cameras. Any two projective images of a plane are in projective correspondence: there is a nonsingular 3×3 matrix $\boldsymbol{H}$ such that $\boldsymbol{x}' \sim \boldsymbol{H}\boldsymbol{x}$ for all corresponding point pairs $\boldsymbol{x}, \boldsymbol{x}'$, where $\sim$ indicates parallelism (equality up to scale) of 3D vectors. The inter-image homography $\boldsymbol{H}$ can be estimated linearly or in statistical least squares from 4 or more corresponding point pairs. We suppose that this has been done, and use $\boldsymbol{H}$ rather than the points themselves. The sign of $\boldsymbol{H}$ should be chosen so that $\boldsymbol{x}'^\top\boldsymbol{H}\boldsymbol{x} > 0$ for all corresponding points $\boldsymbol{x}, \boldsymbol{x}'$ on the plane. The SVD of $\boldsymbol{H}$ is $\boldsymbol{H} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$ where $\boldsymbol{U}$ and $\boldsymbol{V}$ are 3×3 rotation matrices and $\boldsymbol{S} = \mathrm{diag}(s_1, s_2, s_3)$ is positive decreasing diagonal $s_1 \geq s_2 \geq s_3 \geq 0$. We denote the associated columns of $\boldsymbol{U}$ and $\boldsymbol{V}$ by $\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3$ and $\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3$.

In the coordinate frame of the first camera, let the 3D plane be $\boldsymbol{n} \cdot \boldsymbol{x} = z = 1/\zeta$, where $\boldsymbol{n}$ is the *outward* (away from camera) normal. $\zeta = 1/z > 0$ is the inverse distance to the plane. In this frame the first camera has 3×4 projection matrix $\boldsymbol{P}_1 = (\boldsymbol{I}_{3\times3}|\boldsymbol{0})$. Let the matrix of the second camera be $\boldsymbol{P}_2 = \boldsymbol{R}(\boldsymbol{I}_{3\times3}| - \boldsymbol{t})$ where $\boldsymbol{t}$ is the inter-camera translation (*i.e.* the second camera's optical centre) and $\boldsymbol{R}$ the inter-camera rotation. Then the homography from image 1 to image 2 is $\boldsymbol{H} = \boldsymbol{R}\boldsymbol{H}_1$ where $\boldsymbol{H}_1 = \boldsymbol{I}_{3\times3} - \zeta\boldsymbol{t}\boldsymbol{n}^\top$. (For a 3D point $\boldsymbol{x}$ on the plane $\boldsymbol{H}\boldsymbol{x} = \boldsymbol{R}(\boldsymbol{x} - \zeta\boldsymbol{t}\boldsymbol{n}^\top\boldsymbol{x}) = \boldsymbol{R}(\boldsymbol{x} - \boldsymbol{t}) \sim \boldsymbol{P}_2\boldsymbol{x}$, since $\zeta\boldsymbol{n}^\top\boldsymbol{x} = 1$ there. Treating $\boldsymbol{x}$ as a point in image 1 changes only the overall scale factor). Only the product $\zeta\boldsymbol{t}\boldsymbol{n}^\top$ is recoverable, so we normalize to $\|\boldsymbol{t}\| = \|\boldsymbol{n}\| = 1$ (*i.e.* the plane distance $1/\zeta$ is measured in units of the baseline $\|\boldsymbol{t}\|$) and use visibility tests to work out the allowable signs.

The SVD's of $\boldsymbol{H} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}$ and $\boldsymbol{H}_1 = \boldsymbol{U}_1\boldsymbol{S}\boldsymbol{V}$ are identical up to a factor of $\boldsymbol{R}$: $\boldsymbol{U} = \boldsymbol{R}\boldsymbol{U}_1$. $\boldsymbol{H}_1$ leaves the cross-product vector $\boldsymbol{t} \wedge \boldsymbol{n}$ invariant. If the singular values are distinct, $\boldsymbol{t} \wedge \boldsymbol{n}$ must correspond to a singular vector. It turns out that this is always the *middle* one $\boldsymbol{v}_2$, so the 'correct' normalization of $\boldsymbol{H}$ is $\boldsymbol{H} \to \boldsymbol{H}/s_2$, *i.e.* $(s_1, s_2, s_3) \to (s_1/s_2, 1, s_3/s_2)$. We suppose henceforth that this has been done.

Given that $\boldsymbol{t} \wedge \boldsymbol{n}$ corresponds to $\boldsymbol{v}_2$ in the image 1 frame, the $\{\boldsymbol{t}, \boldsymbol{n}\}$ subspace must be spanned by $\{\boldsymbol{v}_1, \boldsymbol{v}_3\}$, say $\boldsymbol{n} = \beta\,\boldsymbol{v}_1 - \alpha\,\boldsymbol{v}_3$, $\boldsymbol{n} \wedge (\boldsymbol{t} \wedge \boldsymbol{n}) \sim \alpha\,\boldsymbol{v}_1 + \beta\,\boldsymbol{v}_3$ for some $\alpha^2 + \beta^2 = 1$. Any direction orthogonal to $\boldsymbol{n}$ — in particular $\boldsymbol{n} \wedge (\boldsymbol{t} \wedge \boldsymbol{n})$ — has norm unchanged by $\boldsymbol{H}$ or $\boldsymbol{H}_1$, whence $(\alpha\, s_1)^2 + (\beta\, s_3)^2 = \alpha^2 + \beta^2$ or $(\alpha, \beta) \sim (\pm\sqrt{1 - s_3^2}, \pm\sqrt{s_1^2 - 1})$. Had we taken $\boldsymbol{t} \wedge \boldsymbol{n}$ to correspond to $\boldsymbol{v}_1$ or $\boldsymbol{v}_3$ above, there would have been no real solution here, so $\boldsymbol{v}_2$ is the only possibility.

Exactly the same argument on the left shows that $\boldsymbol{R}\,\boldsymbol{t} = -(\beta\,\boldsymbol{u}_1 + \alpha\,\boldsymbol{u}_3)$. As $\boldsymbol{t}$ is an eigenvector of

$\boldsymbol{H}_1$ with eigenvalue $1-\zeta\,\boldsymbol{n}\cdot\boldsymbol{t}$, we have $\boldsymbol{H}\,\boldsymbol{t} = (1-\zeta\,\boldsymbol{n}\cdot\boldsymbol{t})\boldsymbol{R}\,\boldsymbol{t}$, whence $\boldsymbol{t} \sim \boldsymbol{H}^{-1}(\boldsymbol{R}\,\boldsymbol{t}) \sim \beta/s_1\,\boldsymbol{v}_1 + \alpha/s_3\,\boldsymbol{v}_3$ and (after simplification) $\zeta = s_1 - s_3$. The left singular vectors of $\boldsymbol{H}_1$ (columns $\boldsymbol{u}'_1, \boldsymbol{u}'_2, \boldsymbol{u}'_3$ of $\boldsymbol{U}_1$) can be recovered by noting that $\boldsymbol{u}'_2 = \boldsymbol{v}_2$ and requiring that $\boldsymbol{t}$ be an eigenvector of $\boldsymbol{H}_1$, whence $\boldsymbol{u}'_1 = \gamma\,\boldsymbol{v}_1 + \delta\,\boldsymbol{v}_3, \boldsymbol{u}'_3 = \delta\,\boldsymbol{v}_1 - \gamma\,\boldsymbol{v}_3$ with (simplifying) $(\gamma,\delta) \sim (1 + s_1\,s_3, \pm\alpha\,\beta)$ and hence

$$\boldsymbol{R} = \boldsymbol{U}\,\boldsymbol{U}_1^\top = \boldsymbol{U} \begin{pmatrix} \gamma & 0 & \delta \\ 0 & 1 & 0 \\ -\delta & 0 & \gamma \end{pmatrix} \boldsymbol{V}^\top$$

In summary, the following OCTAVE/MATLAB routine generates the two possible orientation solutions given homography $\boldsymbol{H}$ (`unitize()` scales two variables so that their sum of squares is 1):

```
function [R1,t1,n1, R2,t2,n2, zeta] = homog_to_Rt(H)
   [U,S,V] = svd(H);
   s1 = S(1,1)/S(2,2);
   s3 = S(3,3)/S(2,2);
   zeta = s1-s3;
   a1 = sqrt(1-s3^2);
   b1 = sqrt(s1^2-1);
   [a,b] = unitize(a1,b1);
   [c,d] = unitize( 1+s1*s3, a1*b1 );
   [e,f] = unitize( -b/s1, -a/s3 );
   v1 = V(:,1); v3 = V(:,3);
   n1 = b*v1-a*v3;
   n2 = b*v1+a*v3;
   R1 = U*[c,0,d; 0,1,0; -d,0,c]*V';
   R2 = U*[c,0,-d; 0,1,0; d,0,c]*V';
   t1 = e*v1+f*v3;
   t2 = e*v1-f*v3;
   if (n1(3)<0)  t1 = -t1; n1 = -n1; end;
   if (n2(3)<0)  t2 = -t2; n2 = -n2; end;
end;

% -- Translations in t'=-Rt form (c.f. P2 = R.(I|-t) = (R|t')):
% rt1 = -(b*U(:,1)+a*U(:,3));
% rt2 = -(b*U(:,1)-a*U(:,3));
% -- Direct form of Essential matrices:
% E1 = (a*U(:,1)-b*U(:,3))*V(:,2)'+U(:,2)*(f*v1-e*v3)'
% E2 = -(a*U(:,1)+b*U(:,3))*V(:,2)'-U(:,2)*(f*v1+e*v3)'
```

Although the above derivation uses $\boldsymbol{t} \wedge \boldsymbol{n}$ and friends, provided that $0 \ll \zeta = s_1 - s_3 \ll \infty$ (*i.e.* the plane is not too far away compared to the baseline $\boldsymbol{t}$, and the cameras stay on one side of it), the given routine is stable even if $\boldsymbol{t}$ is parallel to $\boldsymbol{n}$ (in which case $\boldsymbol{t} \wedge \boldsymbol{n} = \boldsymbol{0}$, $s_1 = s_2 = 1 > s_3$, $(\alpha, \beta) \sim (1,0)$, $\boldsymbol{t} \sim \boldsymbol{n} \sim \boldsymbol{v}_3$, and $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ can not be clearly separated), or orthogonal to $\boldsymbol{n}$ (in which case $s_1 > s_2 = s_3 = 1$, $(\alpha,\beta) \sim (0,1)$, $\boldsymbol{n} \sim \boldsymbol{v}_1$, $\boldsymbol{R}\boldsymbol{t} \sim \boldsymbol{u}_1$, but $\boldsymbol{v}_2, \boldsymbol{v}_3$ and $\boldsymbol{u}_2, \boldsymbol{u}_3$ can not be separated). For a distant plane $\zeta \to 0$, $\boldsymbol{S} \to \boldsymbol{I}_{3\times3}$ and α, β, $\boldsymbol{n}$, and $\boldsymbol{t}$ become unreliable but $\boldsymbol{R}$ should still be accurate. Of course, if the points used to estimate $\boldsymbol{H}$ are not well spread in the images, $\boldsymbol{H}$ and everything else can become inaccurate.