Autocalibration from Planar Scenes

Projective Direction Frames

- Simpler, more direct formulation of theory of autocalibration

Planar Autocalibration

- Camera calibration + Euclidean structure + motion from a ***moving projective camera*** viewing a ***planar scene***

  — unknown scene & motion, unknown but constant intrinsic parameters

- ≥ 5 images needed for a full projective camera model (f, a, s, u_0, v_0)
- ≥ 3 images suffice if only focal length f is estimated
- Numerical optimization based algorithm, choice of initialization methods

Why Use Planes?

1. A common, easy to recognize primitive — “every wall is a calibration grid”
2. Feature extraction & matching are relatively simple
3. A singular case for projective cameras, but not for calibrated ones
   - ***no*** camera motion information can be extracted
   - ***no*** 3D projective structure (only 2D planar structure)
4. ***Effective*** scene planarity is surprisingly common
   - distant scenes, small motions, dominant ground plane, . . .

Direction Frames

Direction Frames are just ***orthonormal sets of 3D directions***

— *i.e.* “orthonormal bases for the plane at infinity”

- In homogeneous Euclidean coordinates they become 4×3 orthogonal matrices

$$\boxed{\mathbf{D} \;=\; \begin{pmatrix} \boldsymbol{R} \\ \mathbf{0} \end{pmatrix} \quad \text{where } \boldsymbol{R} \text{ is a } 3\times 3 \text{ rotation}}$$

- In projective 3D coordinates they become ***arbitrary 4×3 rank 3*** matrices

- They are defined only up to orthogonal mixing of their columns

$$\boxed{\mathbf{D} \longrightarrow \mathbf{D}\,\boldsymbol{R} \quad \text{where } \boldsymbol{R} \text{ is a } 3\times 3 \text{ rotation}}$$

Direction Frames and the Plane at Infinity

- The ***null space*** of a direction frame matrix is the ***plane at infinity***

$$\boxed{\mathbf{p}_\infty \, \mathbf{D} \; = \; \boldsymbol{0}}$$

- In projective coordinates where the first camera matrix is $\boxed{\mathbf{P}_1 \;\simeq\; (\,\boldsymbol{I}_{3\times 3} \,|\, \boldsymbol{0}\,)}$

and the plane at infinity is $\boxed{\mathbf{p}_\infty \;=\; (\boldsymbol{p}^\top \;\; 1)}$, we can choose

$$\boxed{\mathbf{D} \;=\; \begin{pmatrix} \boldsymbol{I} \\ -\boldsymbol{p}^\top \end{pmatrix} \boldsymbol{K}}$$

where $\boldsymbol{K}$ is the ***internal camera calibration matrix***

$$\boxed{\boldsymbol{K} \;=\; \begin{pmatrix} f & f\,s & u_0 \\ & f\,a & v_0 \\ & & 1 \end{pmatrix} \;=\; \begin{pmatrix} 1 & s & u_0 \\ & a & v_0 \\ & & 1 \end{pmatrix} \begin{pmatrix} f & & \\ & f & \\ & & 1 \end{pmatrix}}$$

Basic Autocalibration Constraint

Orthogonal 3D directions project to orthogonal directions in the camera frame

The calibrated projection of a direction frame is orthogonal

$$\boldsymbol{K}^{-1}\,\mathbf{P}\,\mathbf{D} \;\simeq\; \textbf{3}{\times}\textbf{3 rotation}$$

where

$\boldsymbol{K}$ = internal camera calibration

$\mathbf{P}$ = 3×4 camera projection

$\mathbf{D}$ = 4×3 direction frame matrix

Derived Autocalibration Constraints

- To eliminate the unknown rotation, multiply by the transpose on left or right

$$\mathbf{D}^\top \mathbf{P}^\top \boldsymbol{\omega}^{-1} \mathbf{P}\,\mathbf{D} \simeq \boldsymbol{I}_{3\times 3}$$

$$\boldsymbol{K}^{-1} \mathbf{P}\,\boldsymbol{\Omega}\,\mathbf{P}^\top \boldsymbol{K}^{-\top} \simeq \boldsymbol{I}_{3\times 3}$$

$$\mathbf{P}\,\boldsymbol{\Omega}\,\mathbf{P}^\top \simeq \boldsymbol{\omega}$$

where $\boldsymbol{\Omega} = \mathbf{D}\,\mathbf{D}^\top =$ Dual Absolute 3D Quadric

$\boldsymbol{\omega} = \boldsymbol{K}\boldsymbol{K}^\top =$ Dual Absolute Image Conic

- 5 constraints/image on 8 projective structure d.o.f. + $\leq$5 unknown camera pars.

$\Rightarrow$ need $m \geq$3 images for full projective camera, $m \geq$2 if only f is estimated

- Resolve by ***numerical optimization*** — ***much*** stabler than algebraic elimination

Planar Autocalibration Constraint

- Choose two vectors $\mathbf{x}, \mathbf{y}$ of the direction frame parallel to the 3D plane

The calibrated projections of $\mathbf{x}, \mathbf{y}$ are ***orthonormal up to scale***

$$\begin{aligned} \|\boldsymbol{u}\|^2 &= \|\boldsymbol{v}\|^2 \\ \boldsymbol{u} \cdot \boldsymbol{v} &= 0 \end{aligned} \qquad \text{where} \quad (\boldsymbol{u}, \boldsymbol{v}) \equiv \boldsymbol{K}^{-1} \mathbf{P} (\mathbf{x}, \mathbf{y})$$

- Equivalently, the plane's two ***circular points*** project onto the ***image of the absolute conic*** ω^{-1}

$$(\mathbf{P}\,\mathbf{x}_\pm)^\top \boldsymbol{\omega}^{-1} (\mathbf{P}\,\mathbf{x}_\pm) = 0 \qquad \text{where} \quad \mathbf{x}_\pm \equiv \mathbf{x} \pm i\,\mathbf{y}$$

Planar Autocalibration from Homographies

- In practice, the plane is represented projectively by one of its images (say image 1)
- The projections $\mathbf{P}_i$ become ***inter-image homographies*** $\boldsymbol{H}_{i1}$

$$\begin{aligned} \|\boldsymbol{u}_i\|^2 &= \|\boldsymbol{v}_i\|^2 \\ \boldsymbol{u}_i \cdot \boldsymbol{v}_i &= 0 \end{aligned} \qquad \text{where} \quad (\boldsymbol{u}_i, \boldsymbol{v}_i) \equiv \boldsymbol{K}_i^{-1} \boldsymbol{H}_{i1} (\boldsymbol{x}, \boldsymbol{y})$$

Algorithm

- Minimize the ***autocalibration constraint residuals*** over the 4 d.o.f. of $(\boldsymbol{x}, \boldsymbol{y})$ and the $n \leq 5$ free calibration parameters

- ***Statistically motivated*** error model

$$\text{error} \;=\; \sum_{i=1}^{m} \left(\frac{(\|\boldsymbol{u}_i\|^2 - \|\boldsymbol{v}_i\|^2)^2}{\|\boldsymbol{x}\|^2\|\boldsymbol{a}_i\|^2 + \|\boldsymbol{y}\|^2\|\boldsymbol{b}_i\|^2} + \frac{(\boldsymbol{u}_i \cdot \boldsymbol{v}_i)^2}{\|\boldsymbol{x}\|^2\|\boldsymbol{b}_i\|^2 + \|\boldsymbol{y}\|^2\|\boldsymbol{a}_i\|^2} \right)$$

$$\text{where} \quad (\boldsymbol{a}_i, \boldsymbol{b}_i) \;\equiv\; \boldsymbol{K}_i^{-\top}(\boldsymbol{u}_i, \boldsymbol{v}_i)$$

- Can also include a ***weak prior distribution*** on calibration

— *e.g.* gives default values for degenerate cases

How to Choose an Algebraic Error Model

- Poorly weighted algebraic error measure $\Rightarrow$ ***seriously biased results!!***
- ***"Normalization"*** (preconditioning) helps, but is not the whole answer

A better method

1. Start with an arbitrary algebraic error — here $\boldsymbol{e} = (\|\boldsymbol{u}\|^2 - \|\boldsymbol{v}\|^2,\ \boldsymbol{u} \cdot \boldsymbol{v})$
2. Use ***covariance propagation*** to estimate covariance of $\boldsymbol{e}$

$$\boxed{\mathrm{Cov}(\boldsymbol{e}) \approx \epsilon \cdot \begin{pmatrix} \boldsymbol{x}^2\boldsymbol{a}^2 + \boldsymbol{y}^2\boldsymbol{b}^2 & (\boldsymbol{x}^2 - \boldsymbol{y}^2)\boldsymbol{a}\cdot\boldsymbol{b} \\ (\boldsymbol{x}^2 - \boldsymbol{y}^2)\boldsymbol{a}\cdot\boldsymbol{b} & \boldsymbol{x}^2\boldsymbol{b}^2 + \boldsymbol{y}^2\boldsymbol{a}^2 \end{pmatrix}}$$

3. The statistically correct error metric is $\boxed{\chi^2 \approx \boldsymbol{e}^\top \mathrm{Cov}(\boldsymbol{e})^{-1}\, \boldsymbol{e}}$
4. If this is too complicated, approximate — you now know what to aim for!

Numerical Method

- Work with respect to a ***nominal calibration*** $(f, a, s, u_0, v_0) = (1, 1, 0, 0, 0)$
- Stabilized Gauss-Newton iteration — converges quickly and reliably

  . . . but sometimes to the ***wrong solution !***

Initialization

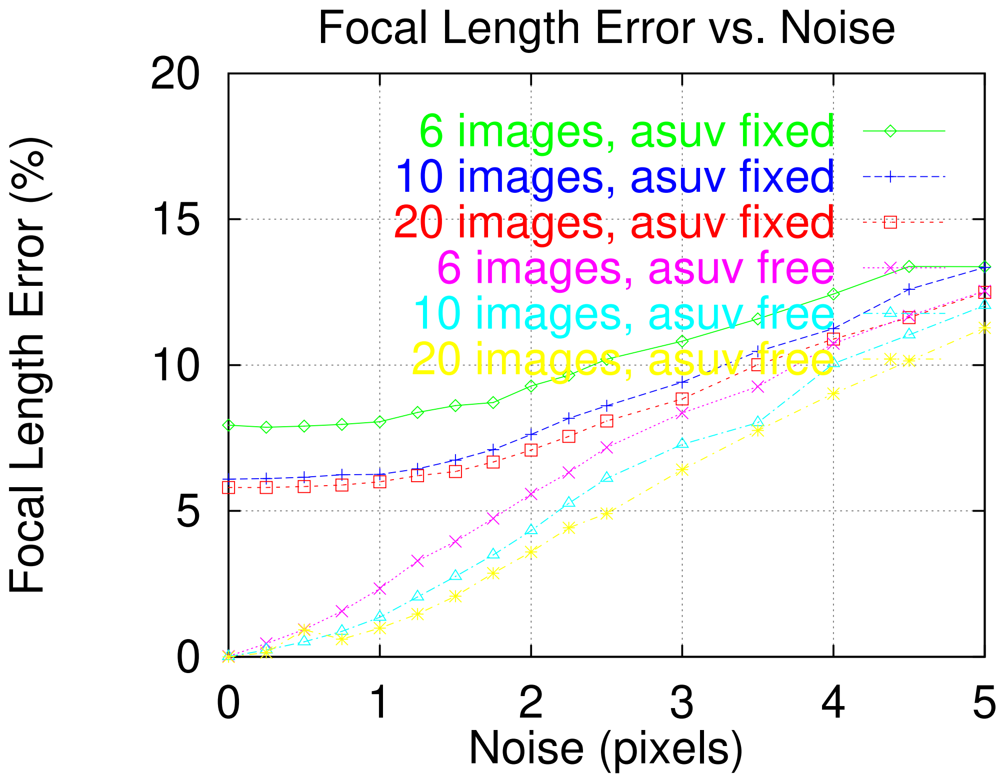
- Algebraic initialization seems difficult — too many images & variables
- Instead use one of three ***numerical initialization methods:***

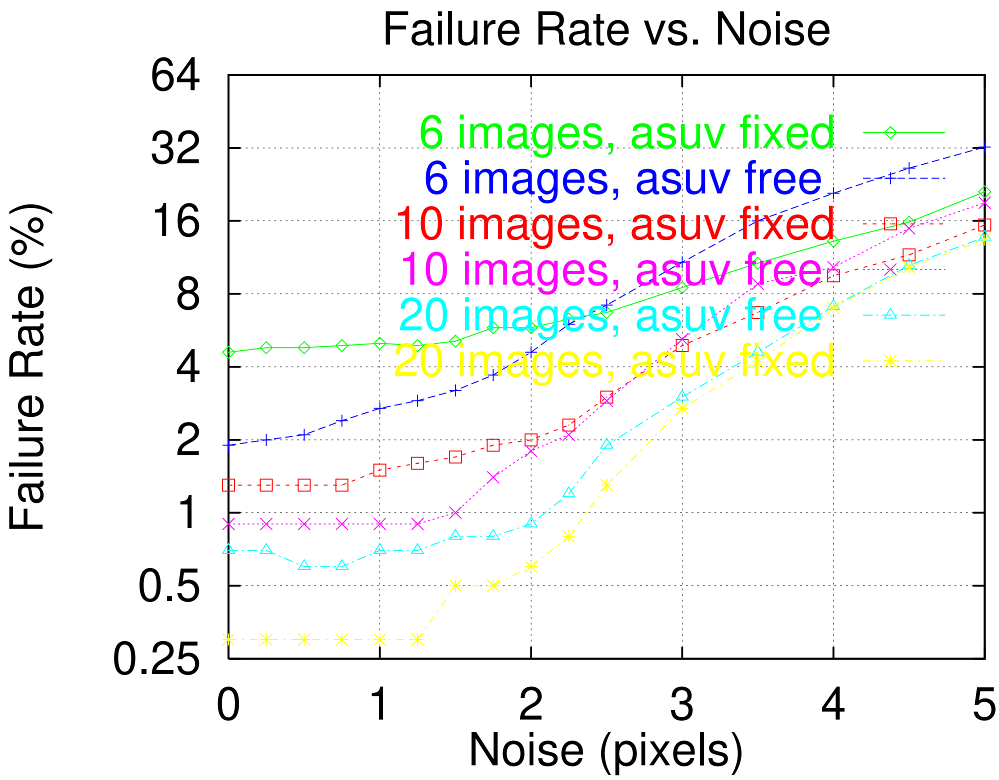
$\boxed{1}$ Start search at nominal calibration

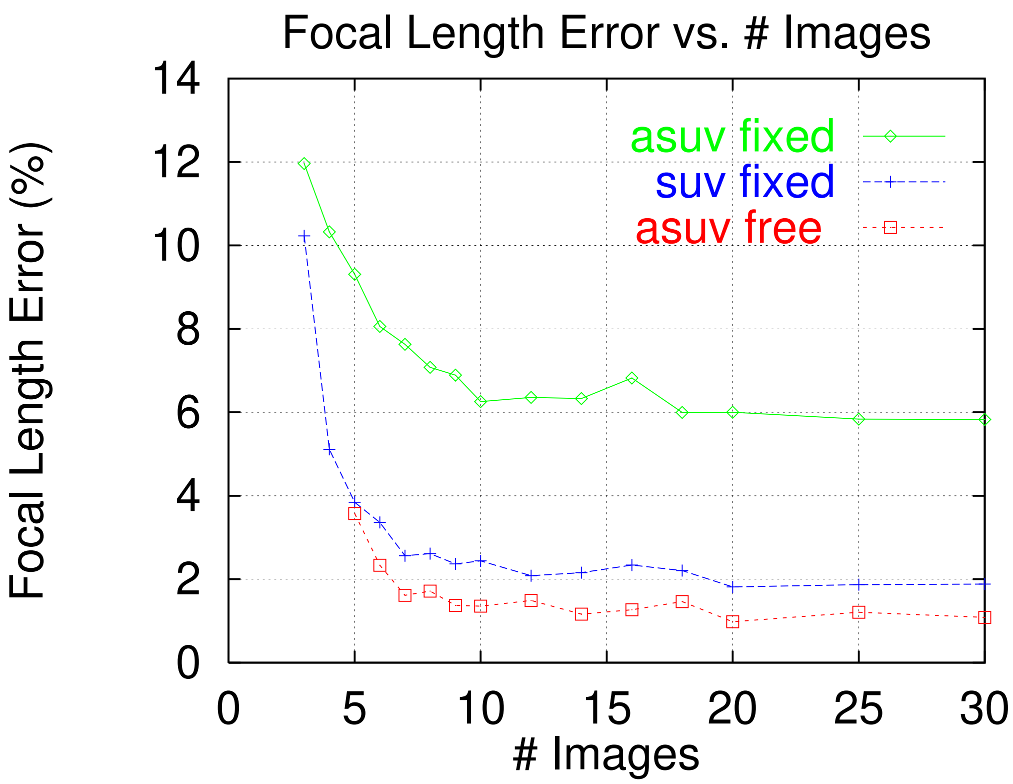
$\boxed{2}$ Line search over f, with nominal (a, s, u_0, v_0)

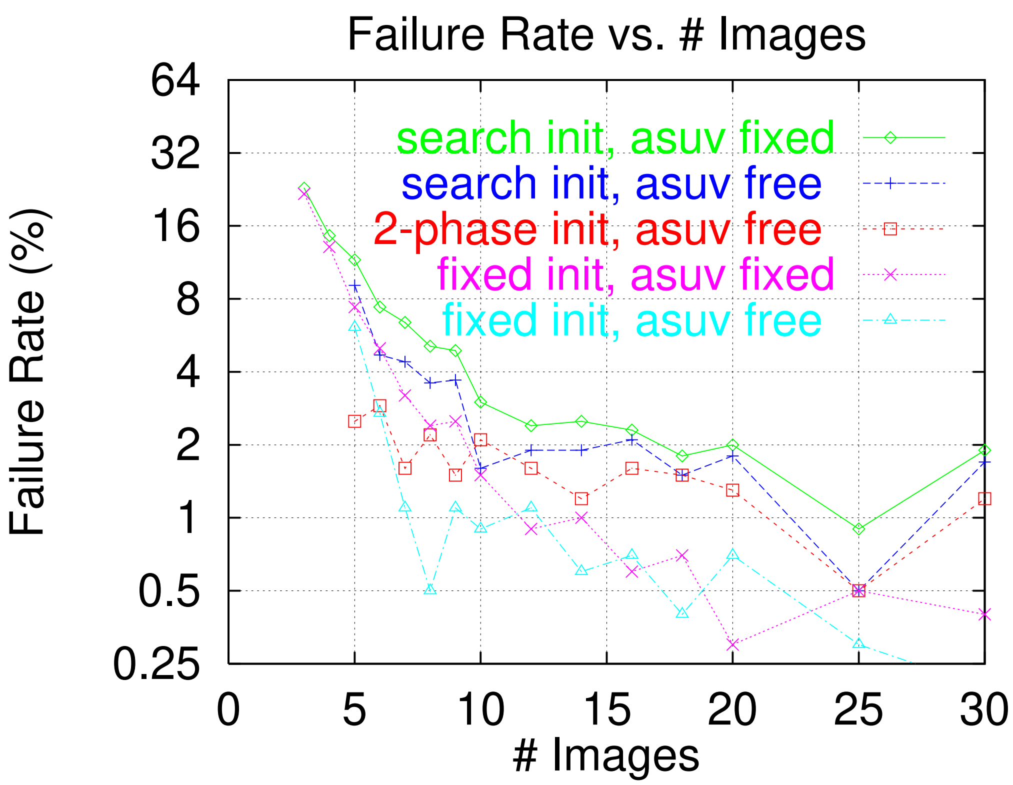
— for each f, the problem reduces to relative orientation of two calibrated cameras from a planar scene (stable new SVD based method for this)

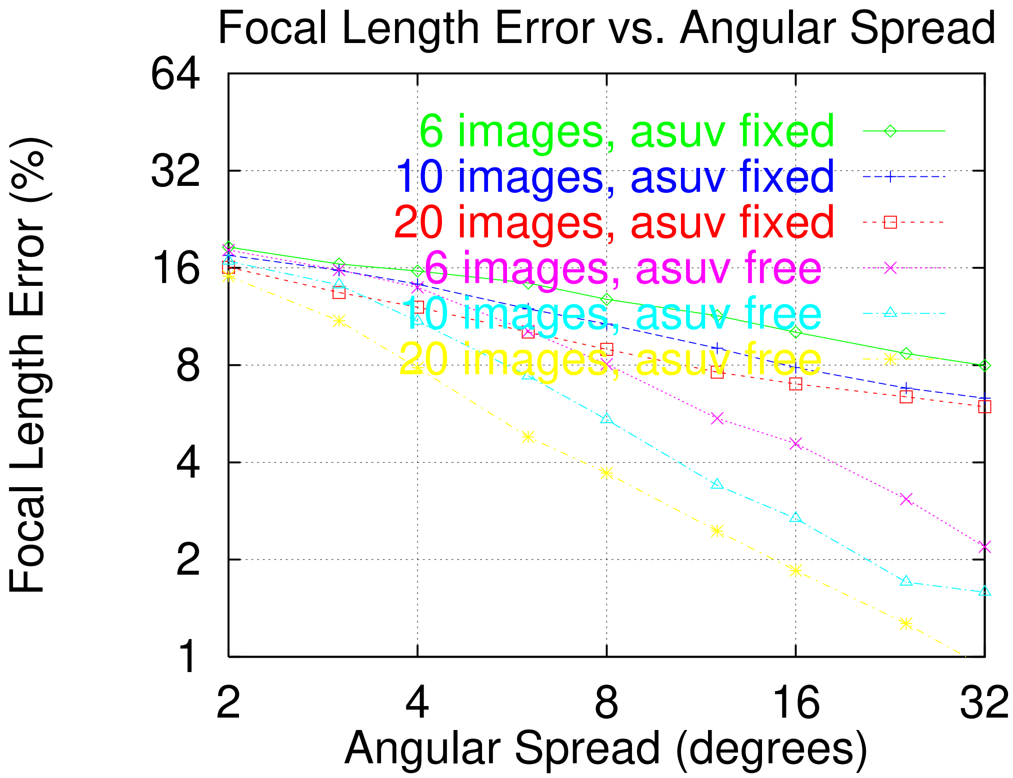
$\boxed{3}$ Hartley's 'rotating camera' method

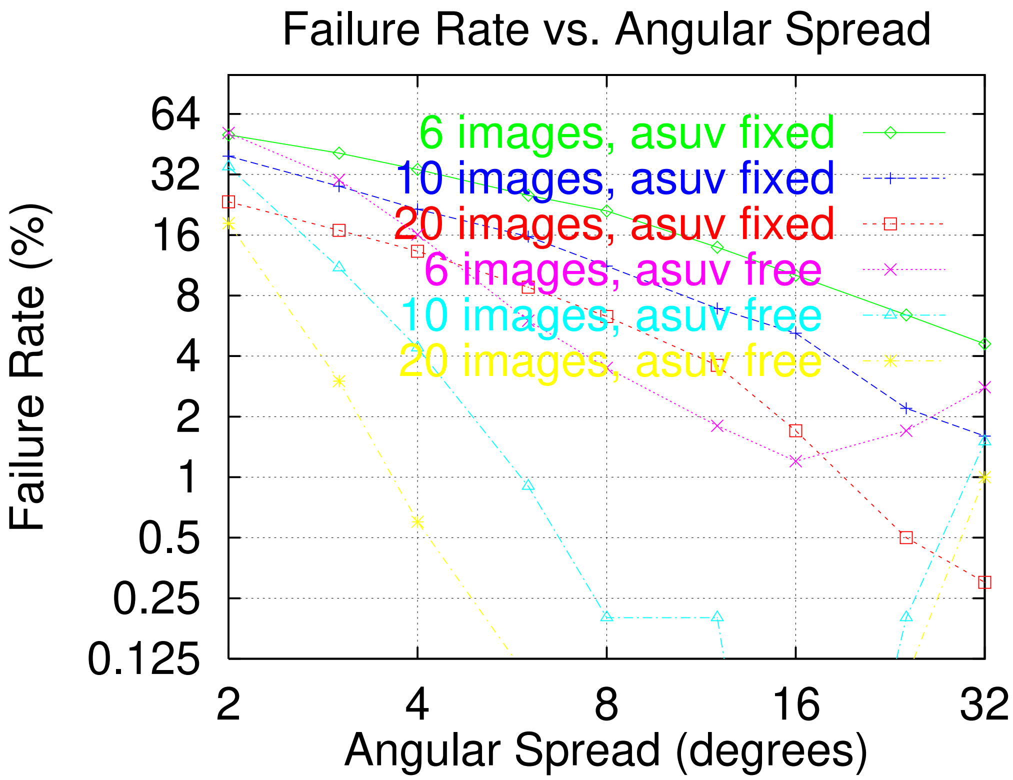

Focal Length Error vs. Noise
Focal Length Error (%)
Noise (pixels)
6 images, asuv fixed
10 images, asuv fixed
20 images, asuv fixed
6 images, asuv free
10 images, asuv free
20 images, asuv free


Failure Rate vs. Noise
Failure Rate (%)
Noise (pixels)
6 images, asuv fixed
6 images, asuv free
10 images, asuv fixed
10 images, asuv free
20 images, asuv free
20 images, asuv fixed


Focal Length Error vs. # Images
Focal Length Error (%)
Images
asuv fixed
suv fixed
asuv free
0
2
4
6
8
10
12
14
0
5
10
15
20
25
30


Failure Rate vs. # Images
Failure Rate (%)
Images
search init, asuv fixed
search init, asuv free
2-phase init, asuv free
fixed init, asuv fixed
fixed init, asuv free
0.25
0.5
1
2
4
8
16
32
64
0
5
10
15
20
25
30


Focal Length Error vs. Angular Spread
Focal Length Error (%)
64
32
16
8
4
2
1
6 images, asuv fixed
10 images, asuv fixed
20 images, asuv fixed
6 images, asuv free
10 images, asuv free
20 images, asuv free
2
4
8
16
32
Angular Spread (degrees)


Failure Rate vs. Angular Spread
Failure Rate (%)
64
32
16
8
4
2
1
0.5
0.25
0.125
6 images, asuv fixed
10 images, asuv fixed
20 images, asuv fixed
6 images, asuv free
10 images, asuv free
20 images, asuv free
2
4
8
16
32
Angular Spread (degrees)

Rules for Calibration Accuracy

- The ***natural geometric error measures*** are

$\|\Delta f\|/f$	relative focal length error
$\|\Delta u_0\|/f, \ \|\Delta v_0\|/f$	principal point error relative to focal length
$\|\Delta a\|$	absolute error in dimensionless aspect ratio
$\|\Delta s\|$	absolute error in dimensionless skew

- For most auto/calibration methods with reasonably strong geometry

$$|\Delta f|/f \ \approx \ |\Delta u_0|/f \ \approx \ |\Delta v_0|/f$$

$|\Delta a| \ \approx \ |\Delta s|$ are about one order of magnitude smaller

- If these rules fail, ***the geometry is probably weak*** (or the algorithm!)

— *e.g.* insufficient camera rotation during autocalibration.

A Nice Idea, but . . .

- All homographies are with respect to a ***key image*** — this could bias results
- Use **homography factorization** to choose a more neutral frame

— analogous to depth + factorization based projective reconstruction

— the key is to find ***consistent scale factors*** for the homographies

- A nice idea, but in the end it ***didn't improve the results***

. . . this suggests that using a key image doesn't cause too much bias

Homography Factorization Method

1. Estimate homographies between all image pairs $(\boldsymbol{H}_{ii} = \boldsymbol{I}, \boldsymbol{H}_{ij} = \boldsymbol{H}_{ji}^{-1})$
2. Find self-consistent relative scalings $\boldsymbol{H}_{ij} \to \lambda_{ij}\, \boldsymbol{H}_{ij}$:

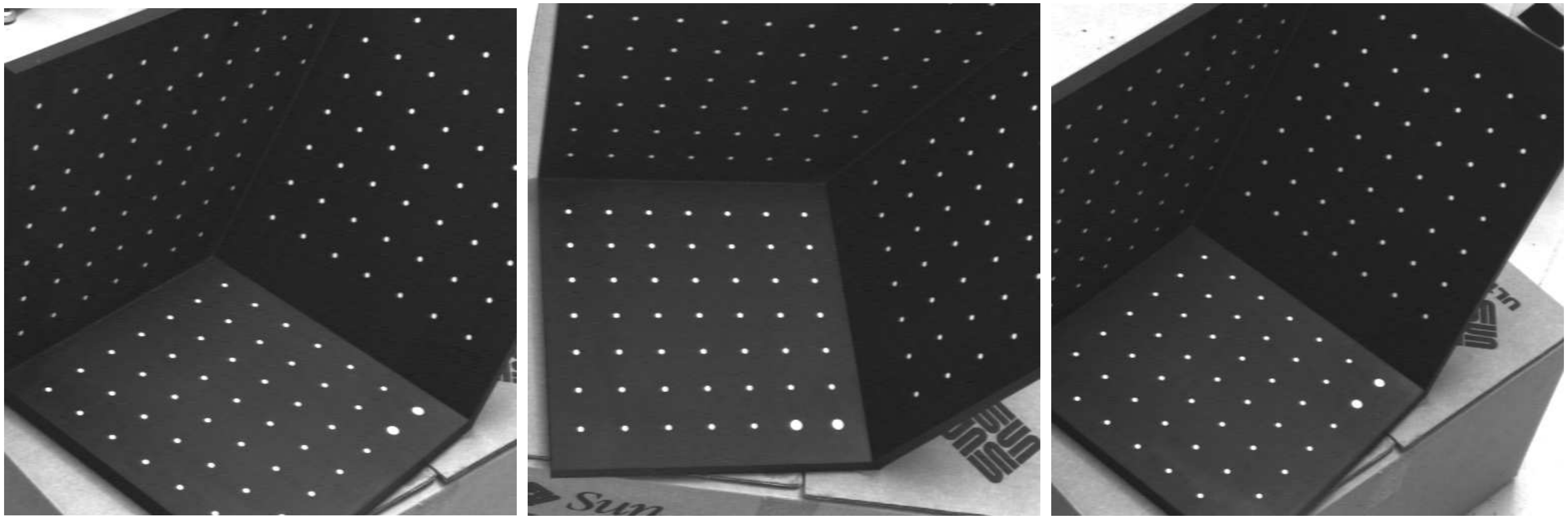
— choose a key image 1 and define $\lambda_{i1} = \lambda_{1j} = 1$

— enforce $\boxed{\boldsymbol{H}_{ij} \;\approx\; \boldsymbol{H}_{i1}\, \boldsymbol{H}_{1j}}$ by choosing $\boxed{\lambda_{ij} \;=\; \frac{\mathrm{Trace}\left(\boldsymbol{H}_{i1}\, \boldsymbol{H}_{1j} \cdot \boldsymbol{H}_{ij}^{\top}\right)}{\mathrm{Trace}(\boldsymbol{H}_{ij} \cdot \boldsymbol{H}_{ij}^{\top})}}$

— ***balance*** the scales for numerical stability

3. Rescale and factorize to rank 3

$$\boxed{\begin{pmatrix} \lambda_{11}\boldsymbol{H}_{11} & \cdots & \lambda_{1m}\boldsymbol{H}_{1m} \\ \vdots & \ddots & \vdots \\ \lambda_{m1}\boldsymbol{H}_{m1} & \cdots & \lambda_{mm}\boldsymbol{H}_{mm} \end{pmatrix} \approx \begin{pmatrix} \boldsymbol{H}_1 \\ \vdots \\ \boldsymbol{H}_m \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{H}}_1^{-1} & \cdots & \tilde{\boldsymbol{H}}_m^{-1} \end{pmatrix}}$$



	f only	f	a	s	u_0	v_0
calibration	-	1515±4	0.9968±0.0002	-	271±3	264±4
6 images	1584±63	1595±63	0.9934±0.0055	0.000±0.001	268±10	271±22
8 images	1619±25	1614±42	0.9890±0.0058	–0.005±0.005	289±3	320±26
10 images	1612±19	1565±41	1.0159±0.0518	–0.004±0.006	273±5	286±27

Summary

Direction Frame approach to Autocalibration

- ⊕ Conceptually simpler — avoids most of the abstract algebraic geometry
- ⊕ Much stabler & fewer degeneracies than Kruppa approach

Planar Autocalibration

- ⊕ Accuracy seems reasonable
- ⊕ Degeneracies seem to be those of 3D autocalibration
- ⊖ Many images required — 8–10 recommended in practice
- ⊖ Occasional convergence to false solutions remains a problem

http://www.inrialpes.fr/movi/people/Triggs/home.html