Parameter Estimation with Data Outliers

Goal: Discuss the use of RANSAC to fit parameterized models to data which includes outliers.

It is common in computational vision to attempt to fit a parameterized model to image features despite:

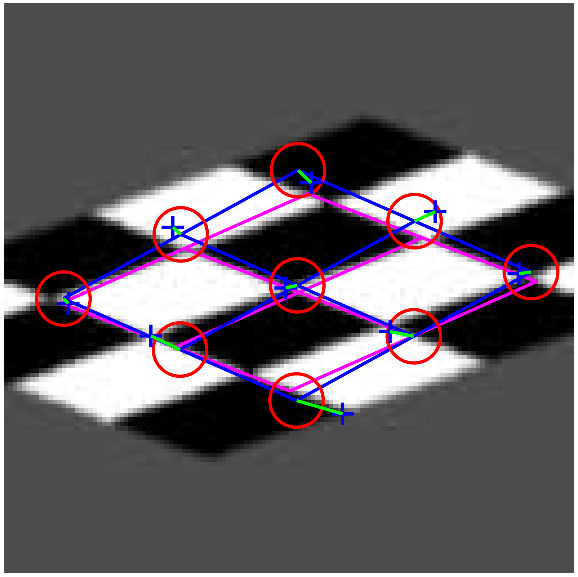
- Data arising from false positives of a feature detector (such as an edge or corner point);
- The presence of multiple objects in the image.

Both of these situations can lead to outliers in the data set for a parameter estimation problem.

Readings: Szeliski, Sections 6.1 and 6.2, Appendices B1 through B3.

Model Problem: Affine Pose Fitting

We return to the model problem of fitting the affine pose parameters for a calibration pattern given corner data with an outlier.



It is tempting to try approaches which consider using a LS fit, removing the point with the largest error, and refitting.

However, this type of approach is not generally successful since a least squares solution can be strongly perturbed by a single outlier, and the outlier(s) need not have the largest errors.

Instead, RAndom SAmple Consensus (RANSAC) is a common approach.

RAndom SAmple Consensus (RANSAC) Algorithm

Suppose we are given data points $\{\vec{z}_k\}_{k=1}^{K}$, which may include outliers.

Assume the following are given:

- Let J be the minimum number of data points needed to compute an approximate solution $\vec{q}$ (for affine fitting, $J = 3$ image points);
- Let $\epsilon > 0$ be an error tolerance (Typically $\epsilon = \rho\sigma$ where σ is an estimate for the standard deviation of the measurement noise and $\rho \approx 3$.);
- Let T be the number of random sampling trials to do.

RANSAC Algorithm (Continued)

Loop T times:

1. Randomly select J data points, $\vec{z}_{k(j)}$, $j = 1, \ldots, J$.
2. Minimize the least squares objective

$$\vec{q}^* = \arg\min \left[\sum_{j=1}^{J} ||\vec{z}_{k(j)} - \vec{f}_{k(j)}(\vec{q})||^2 \right], \quad \text{(note only the } J \text{ sampled points are used).}$$

3. Identify the inliers, $\mathrm{In} = \{k \mid ||\vec{z}_k - \vec{f}_k(\vec{q})|| < \epsilon, 1 \le k \le K\}$.
4. If the number of inliers, $|\mathrm{In}|$, is the largest seen so far, remember the current inlier set, In (and possibly $\vec{q}^*$).

End loop

5. Solve for $\vec{q}^*$ using all points $k \in \mathrm{In}$ (i.e., all inliers).
6. Re-solve for the inlier set In as done in steps 3 and 4 above.

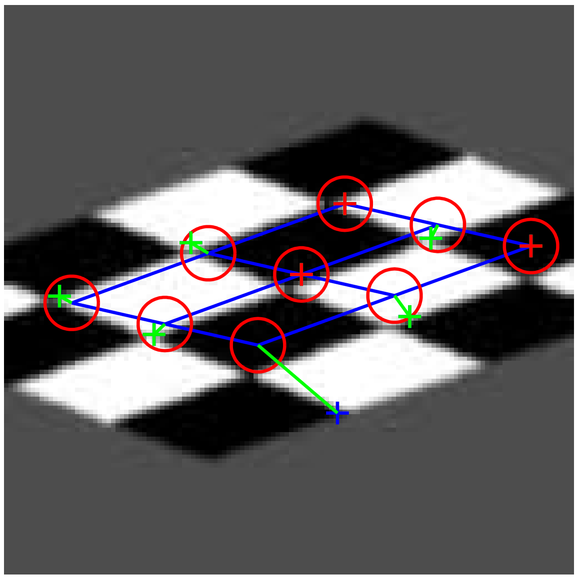
Can iterate steps 5-6 until the set of inliers In does not change substantially.

Steps 5-6 involve iteratively choosing weights, $w_k = 1(0)$ if $k \in In$ (or $k \notin In$, respectively). It is a form of *iteratively reweighted least squares* (IRLS).

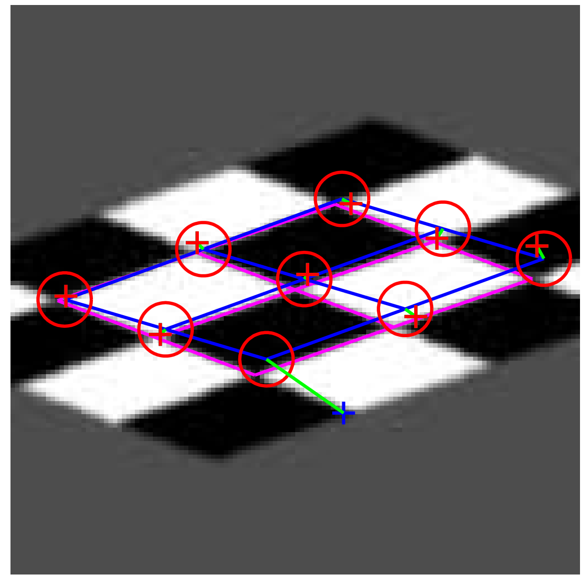
RANSAC: Affine Pose Example

At least six independent constraints are needed to determine the 2D affine pose parameters, $\vec{q} \in \mathbb{R}^6$. The minimum number of labelled image points, $\vec{z}_k \in \mathbb{R}^2$ is therefore $J = 3$.

Initial RANSAC Solution



RANSAC Solution



The pose parameters of the blue grid (above left) is solved using the $J = 3$ sampled points (red). The solution after IRLS (blue grid, above right) and inliers (red points). The recruitment of more inlier data points improves the solution. The mean solution (mauve grid) has negligible bias, despite the outlier.

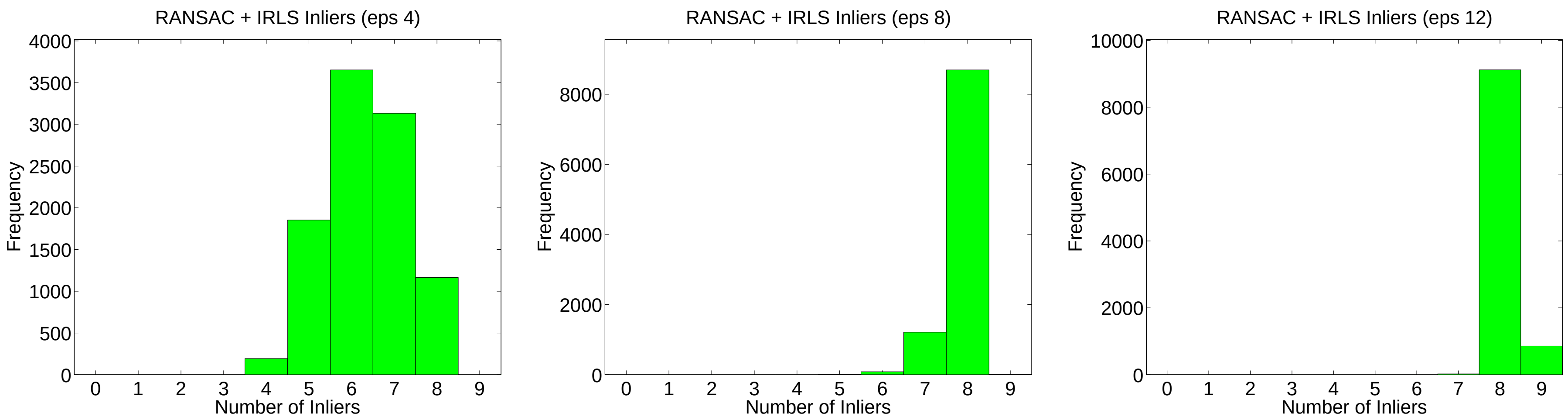
The RMS noise in the data points was 2.6 (pixels), we used $\epsilon = 8$ (pixels) (i.e., $\rho \approx 3$).

RANSAC: Choosing the Error Tolerance

The error tolerance ϵ for the inlier test: $||\vec{z}_k - A_k\vec{q}_0|| < \epsilon$ should be chosen so that:

1. data inliers typically pass the test. This suggests $\epsilon = \rho\sigma$ where σ is an estimate for the standard deviation of the noise and $\rho \approx 3$.

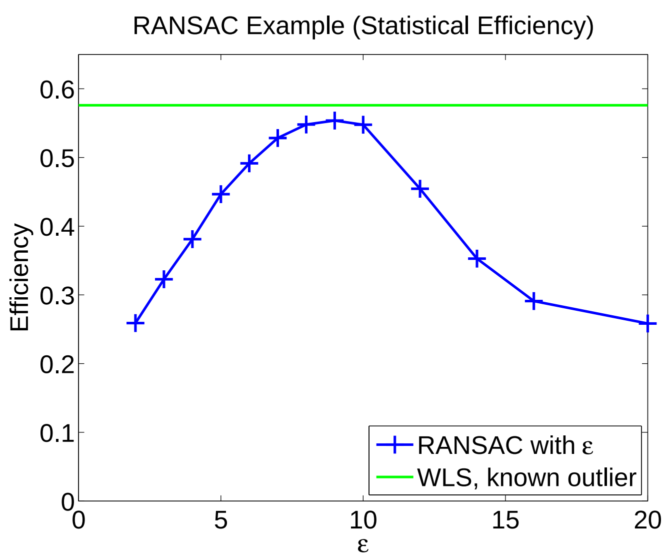
2. data outliers typically fail the test. This suggests choosing ϵ to be as small as possible (subject to the previous point).

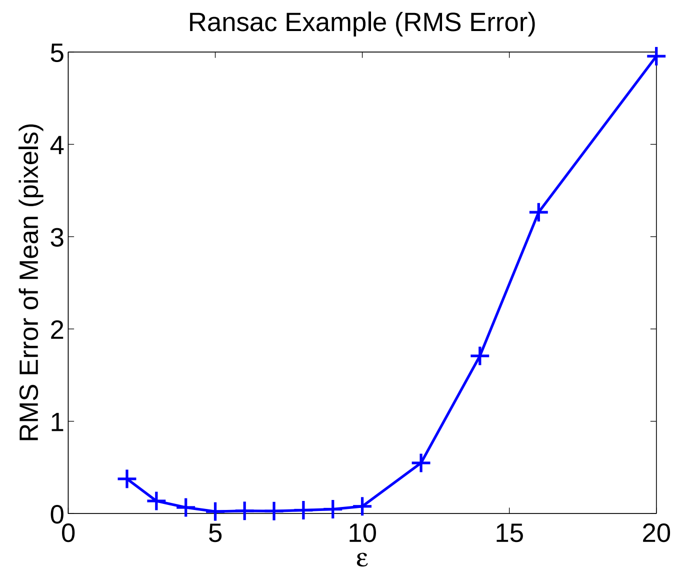


The number of inliers used in the IRLS solution is shown for $\epsilon = 4, 8, 12$ pixels, respectively. (Here the RMS inlier noise is $\sigma = 2.6$ pixels, so these correspond to $\rho = 1.5, 3, 4.6$, respectively.) Data inliers are being rejected for $\rho = 1.5$, $\epsilon = 4$, while the outlier is being accepted about 10% of the time for $\rho = 4.6$, $\epsilon = 12$. Using $\epsilon \approx 3\sigma$ is a reasonable compromise.

RANSAC: Statistical Efficiency

The statistical efficiency of RANSAC can be compared to the special case of using WLS where the data outlier is known and receives a weight of 0 (all other points get a weight of 1).





The above left plot shows that for ϵ either too small or too large the statistical efficiency suffers. The RMS inlier noise is $\sigma = 2.6$ pixels, and $\epsilon = 3\sigma$ is seen to provide reasonable performance (although this depends on the magnitude of the error in the outlier).

The above right plot indicates that RANSAC is unbiased for this problem for an intermediate range of ϵ, but a bias appears as ϵ increases and the outlier is accepted as an inlier.

RANSAC: How Many Trials?

Suppose our data set consists of a fraction p inliers, and $1 - p$ outliers.

How many trials T should be done so that we can be reasonably confident that at least one sampled data set of size $d = 3$ was all inliers?

The probability of choosing $d = 3$ inliers from such a population is roughly p^d when $K >> d$ (it is exactly p^d if we sample with replacement).

So the probability that a given trial of RANSAC fails to select d inliers is $1 - p^d$.

Therefore, the probability that RANSAC failed to have any of the T trials select d inliers is $(1 - p^d)^T$.

In other words,

$$P_0 = 1 - (1 - p^d)^T$$

is the probability that at least one of the RANSAC trials will select d inliers.

RANSAC: How Many Trials?

Given an estimate for the fraction of inliers p in the data set, we could then choose T such that $P_0 > 0.95$, say. That is,

$$T > \log(1 - P_0)/\log(1 - p^d).$$

For example, for 70% inliers and $d = 3$, we require $T > 7$.

Alternatively, if we only have 20% inliers, the same formula states that T should be chosen to be at least 373.

The number of samples required grows rapidly with the both the probability of outliers and with the number d of inliers required to compute a candidate solution.

RANSAC: How Many Trials? (Cont.)

Note T above is simply the number of samples needed to ensure that d inliers is chosen with probability P_0.

In practice, we need these d inliers to provide a good candidate solution for the parameters $\vec{q}$.

For example, the three data points for estimating the affine pose parameters $\vec{q}$ cannot be colinear, and preferably should be spread out in the image.

Planar Homography for a World Plane

The perspective view of any plane in the world (such as a checkerboard) is modelled as:

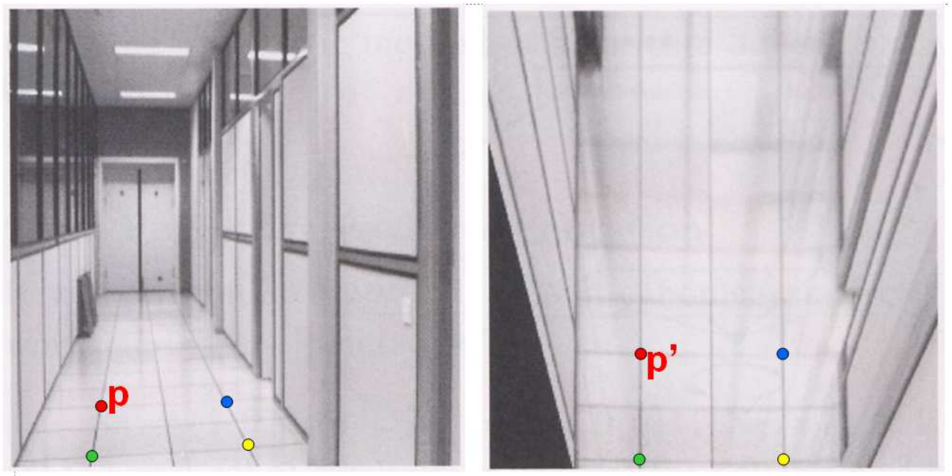
$$\vec{p}^{\,h} = M_{in} M_{ex} \begin{pmatrix} X_1 \\ X_2 \\ 0 \\ 1 \end{pmatrix} = H \vec{x}^{\,h}, \text{ where } \vec{x}^{\,h} = (X_1, X_2, 1)^T. \tag{1}$$

Here X_1, X_2 are world coordinates on the 3D plane (with $X_3 = 0$) and $\vec{p}^{\,h}$ are homogeneous pixel coordinates.

The mapping (1) is referred to as a *planar* or *2D homography*.

Planar Homography: Degrees of Freedom

Since the overall magnitude of H does not matter in the homography $\vec{p}^h = H\vec{x}^h$, there are effectively 8 unknown parameters. Four pairs of 2D image points are sufficient.



from Freeman and Torralba, 2011.

1. **Estimate the homography.** Solve the equations $\alpha_k \vec{p}_k^h = H\vec{x}_k^h$ for $k = 1, \ldots, 4$ for H, with $||H||_F = 1$, say.

2. **Back project.** Given a coordinate $\vec{r}^h$ on the ground plane, and the desired perpendicular mapping to pixels in the warped image, $M_{in}^r \vec{r}^h$, find corresponding pixel coordinates in the original image, namely $\vec{p}^h = H\vec{r}^h$.

3. **Interpolate.** The warped image at pixel $M_{in}^r \vec{r}^h$ is set to be the interpolated value of the original image at $\vec{p}^h$

Planar Homography: Normalization

It is convenient to normalize the third row of H, which provides the factor we need to rescale homogeneous vectors by.

One simple approach is to set $H_{3,3} = 1$. However, this normalization does not allow homographies with $H_{3,3} = 0$ to be represented (and provides a poorly scaled representation for those with $|H_{3,3}|$ relatively small).

Alternatively, we can normalize H by assuming that

$$H_{3,j} = 1, \quad \text{and } |H_{3,i}| \leq 2 \text{ for } i \neq j. \tag{2}$$

That is, the j^{th} element on the third row of H is set to one, under the condition that the other elements on this row are not too large in absolute value.

Fitting 2D Homographies

Given observed image points $\{\vec{z}_k\}_{k=1}^K$ from known 3D planar points $\{\vec{x}_k^{\,h}\}_{k=1}^K$, we then wish to minimize the sum of squared standard errors

$$S^3E(\vec{q}) \;=\; \frac{1}{2}\sum_{k=1}^{K}(\vec{z}_k - \vec{f}_k(\vec{q}))^T\Sigma_k^{-1}(\vec{z}_k - \vec{f}_k(\vec{q})). \tag{3}$$

Here the parameterized model for the k^{th} point is

$$\vec{f}_k(\vec{q}) = \frac{1}{\vec{h}_3^T\vec{x}_k}H_1\vec{x}_k^h, \tag{4}$$

where

- $\vec{x}_k^h = (X_{1,k}, X_{2,k}, 1)^T$ corresponds to the world coordinates of the k^{th} point in the plane $X_3 = 0$;
- H_1 denotes the 2×3 matrix consisting of the first two rows of H;
- $\vec{h}_3^T$ denotes the third row of H.

The unknown vector $\vec{q}$ consists of the eight free coefficients of H, remembering that $H_{3,j} = 1$.

Minimizing (3) is therefore a nonlinear least squares problem. We need to generate an initial guess for the parameters, and deal with outliers.

Algebraic Linearization

If we reweight the objective function (3) by multiplying each term by

$$w_k = \left[\vec{h}_3^T \vec{x}_k^h\right]^2, \tag{5}$$

then we find the weighted least squares objective

$$\begin{aligned} WLS(\vec{q}) &= \frac{1}{2}\sum_{k=1}^{K} w_k(\vec{z}_k - \vec{f}_k(\vec{q}))^T \Sigma_k^{-1}(\vec{z}_k - \vec{f}_k(\vec{q})), \\ &= \frac{1}{2}\sum_{k=1}^{K}((\vec{h}_3^T \vec{x}_k^h)\vec{z}_k - H_1\vec{x}_k^h)^T \Sigma_k^{-1}((\vec{h}_3^T \vec{x}_k^h)\vec{z}_k - H_1\vec{x}_k^h) \end{aligned} \tag{6}$$

is quadratic in the unknowns $\vec{q}$. Hence minimizing $WLS(\vec{q})$ is a linear least squares problem.

This reweighting raises two issues:

1. It emphasizes the noise in some observations over others, decreasing statistical efficiency;
2. The weights *depend on the unknowns* $\vec{q}$, perhaps leading to biased estimates. (For example, all else being equal, $WLS(\vec{q})$ is reduced if the weights w_k can be reduced.)

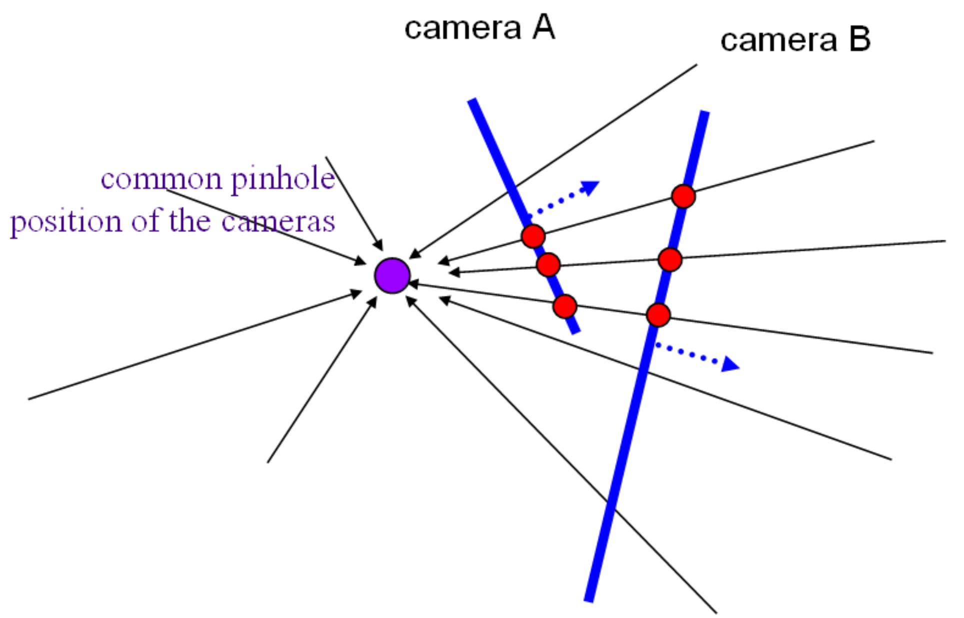
Basic Approach for Fitting Planar Homographies

Planar Homography Estimation Algorithm:

1. Estimate/Guess the (isotropic) standard deviation, σ_0, of the noise in the data points $\vec{z}_k$.
2. Use Ransac to solve the linear WLS problem (6), identifying the set of data inliers, $k \in In$. Use an isotropic, identically distributed noise model, $\Sigma_k = \sigma_0^2 I$.
3. Use the solution $\vec{q}$ from step 2 as an initial guess for nonlinear optimization software applied to the nonlinear least squares problem (3). Restrict the sum in (3) to the inliers identified in step 2, namely $k \in In$. Use $\Sigma_k = \sigma_0^2 I$.
4. Check the solution from step 3. For example, re-estimate σ_0 by assuming the inlier errors $\vec{e}_k = \vec{z}_k - \vec{f}_k(\vec{q})$ are roughly Normal. In that case, $\sigma_0 \approx 1.5\,\text{median}\,\{||\vec{e}_k||$ for $k \in In\}$. Recompute the inlier set $In = \{k \mid ||\vec{e}_k|| < \rho\sigma_0\}$. If there is a substantial change in the inlier set, repeat steps 3-4.

Camera Rotation

Suppose the camera rotates about the (front) nodal point, i.e., the center of projection in world coordinates.



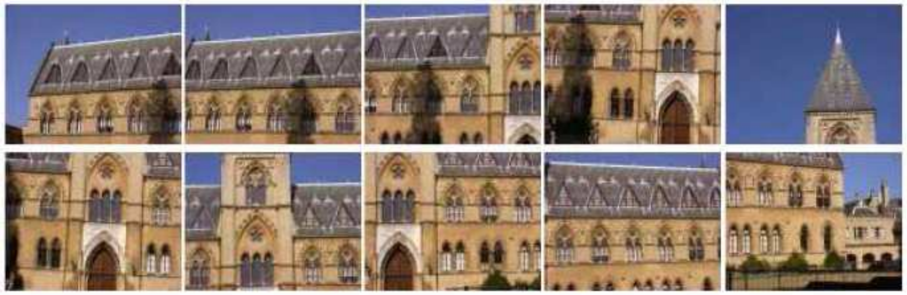
from Freeman and Torralba, 2011.

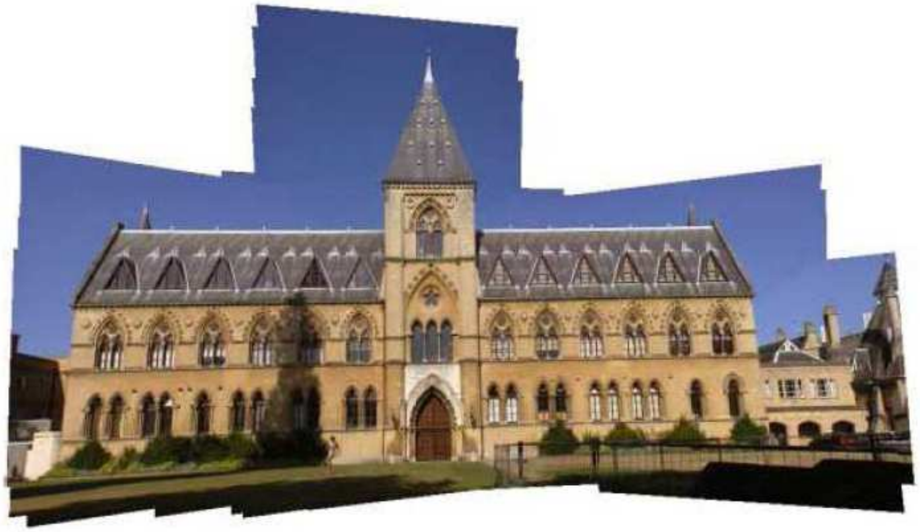
Any image can be thought of as being printed on a 3D world plane (i.e., the camera's image plane reflected to be in front of the center of projection).

From a second orientation, this first plane can be mapped to the current image plane using a homography.

Image Stitching

Given a collection of images obtained by rotating the camera about the front nodal point, they can all be mapped back to a single image plane. To do this a homography has to be estimated for each image, and that image then needs to be warped and blended into the final mosaic.





See http://www.cs.bath.ac.uk/brown/autostitch/autostitch.html.

References

The original paper on RANSAC:

M.A. Fischler and R.C. Bolles, Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, Communications of the ACM, 24(6), 1981, pp. 381-395.

Two papers on algorithmic speed-ups for large data sets:

Daniel Nistér, Preemptive RANSAC for live structure and motion estimation. In Ninth International Conference on Computer Vision (ICCV 2003), pp. 199-206, Nice, France.

O. Chum and J. Matas, Matching with PROSAC - progressive sample consensus, In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), pp. 220-226, San Diego, CA.

For models with a few parameters, such as image lines or circles, the *Hough transform* is an alternative approach that can tolerate outliers. It is based on voting in parameter space. See Section 4.3.2 of Szeliski's book for an introduction.

The RANSAC algorithm discussed here discretely categorizes data into inliers versus outliers, and uses the associated binary weights. Robust M-estimation is an alternative approach for which the weights are smooth functions of the unknowns $\vec{q}$. See:

Allan Jepson, Lecture Notes on Robust Estimation, Univ. of Toronto, 2004 (available from `http://goo.gl/NCu5z`).

For more information about image stitching see:

M. Brown and D. Lowe. Automatic Panoramic Image Stitching using Invariant Features. International Journal of Computer Vision, 74(1), pages 59-73, 2007.

M. Brown and D. G. Lowe. Recognising Panoramas. In Proceedings of the 9th International Conference on Computer Vision (ICCV2003), pages 1218-1225, Nice, France, 2003.