ECE 661
PURDUE U.
AVI KAK

Lecture 11

Robust Homography Estimation with the RANSAC Algorithm

- The Linear Least-Squares methods I presented in the last lecture can tolerate only a small amount of noise in the $(\mathbf{x}, \mathbf{x}')$ correspondences. That is, if the measured coordinates for $\mathbf{x}$, and $\mathbf{x}'$ are off by no more than a couple of pixels from their true values, the algebraic minimization of $\|A\vec{h}\|$ subject to $\|\vec{h}\| = 1$ would still get you a good estimate for the homography.

- However, when one tries to automate all of the steps required for calculating a homography (that is, when one also tries to figure out the $\mathbf{x}_i \leftrightarrow \mathbf{x}_i'$ correspondences automatically), the set of correspondences you feed into a homography calculator is likely to contain several false pairings. If the correspondence data you feed into a linear least-squares calculator includes false pairings, you'll surely end up with a strange result for the homography.

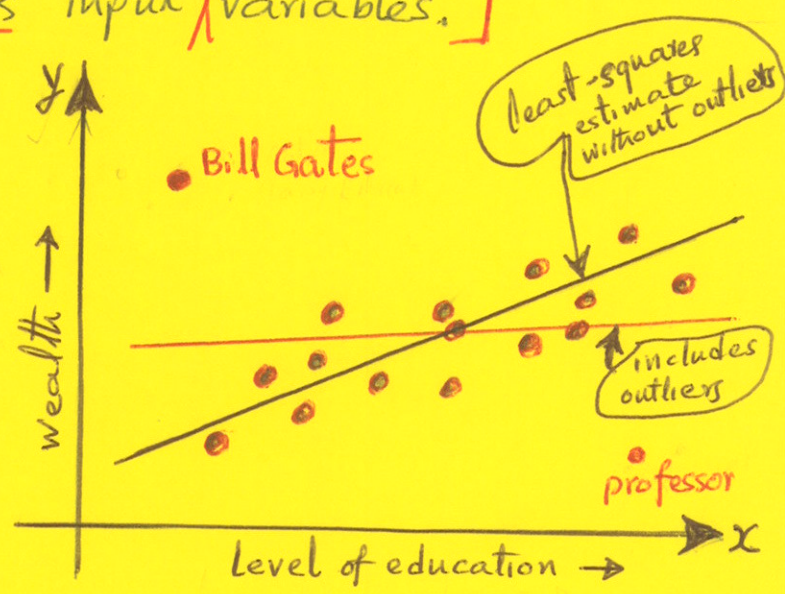
- You can think of the merely noisy $(\mathbf{x}, \mathbf{x}')$ correspondences as constituting the inliers, and the false $(\mathbf{x}, \mathbf{x}')$ correspondences as constituting the outliers for the calculation of the homography.

- So what we need is an outlier rejection mechanism before we invoke the linear least-squares method on the inliers. And that takes us to the subject of robust estimation with the RANSAC algorithm. RANSAC stands for "Random Sample Consensus". I think a better expansion of RANSAC is "Random Sampling And Consensus". The goal of robust estimation is to reject outliers.

- Fundamental to the RANSAC algorithm is using randomly-selected least amount of data to construct an estimate and to then ascertain the extent of support that the rest of the data provides to the estimate. We accept the estimate only if the support exceeds a threshold. When an estimate made in this manner is accepted, the supporting data constitutes the inliers and the rest of the data the outliers. After an estimate is accepted, it can be refined using all the inliers.

- To illustrate RANSAC with a simple example, consider the problem of linear regression. **Linear regression** refers to estimating a linear relationship between an input random variable and an output random variable, both assumed to be continuous. [For an example of another kind of regression, we can think of **logistic regression**, which refers to estimating the relationship between a binary output (random variable) and a set of continuous input (random variables).]

- Consider x and y as two random variables, with x standing for the level of education of an individual and y as his/her wealth. Linear regression in this case amounts to estimating a 1-D affine affine transformation between the two variables: $\boxed{y = ax + b}$



In general, any relationship like $\vec{y} = A\vec{x} + \vec{b}$ can be expressed as $\begin{bmatrix} \vec{y} \\ 1 \end{bmatrix} = \begin{bmatrix} A & \vec{b} \\ \vec{0}^T & 1 \end{bmatrix} \begin{bmatrix} \vec{x} \\ 1 \end{bmatrix} = H \begin{bmatrix} \vec{x} \\ 1 \end{bmatrix}$ where H is obviously an affine transformation.

- Since the least number of points needed to form a line estimate is 2, RANSAC will randomly pick two of the data points, construct a line on the basis of those two points, and then measure the support the rest of the data provides for that line. The support is measured by the number of points that lie within a distance threshold of the line. The points within the distance threshold constitute the inlier set (it is also referred to as the consensus set).

- The random selection of two points, the formation of a line estimate, and the measurement of support for that line is repeated a number of times and the line with the most support is accepted as the robust fit.

- It follows from the last two bullets that the following three parameters are important to the working of the RANSAC algorithm:

  δ ① We need a decision threshold to construct the inlier set. When we compare a data point with the estimate, if the distance between the two is less than the threshold, we place it in the inlier set.

  N ② N for the number of trials to conduct. It is often computationally infeasible to attempt all possible trials.

  M ③ A minimum value for the size of the inlier set for it to be acceptable.

  In most cases, these parameters are set by experimenting with the data.

- In some cases, it may be possible to set the distance parameter δ with the help of a statistical model for the inlier noise. Earlier I said that we consider the noisy (x, x') correspondences to constitute the inliers and the false (x, x') correspondences to constitute the outliers. In most cases, we would expect the inlier noise to consist of a displacement of the true location of a point by no more than a couple of pixels.

- It would be common to assume that, for the inliers, the noise-induced displacement of the true location of a pixel would be modeled by the Gaussian $g(\Delta x, \Delta y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(\Delta x)^2 + (\Delta y)^2}{2\sigma^2}}$ where σ is set to a small number between 0.5 and 2.

- The distance d between the true location of the correspondence and its measured location, with $d^2 = (\Delta x)^2 + (\Delta y)^2$, is governed by a chi-squared distribution with 2 degrees of freedom. We want to set the threshold δ so that 90% of the inliers would be accepted. We can consult the published tables of cumulative chi-squared distribution for determining that threshold. More commonly, though, we set $\delta = 3\sigma$.

We model the inlier noise by assuming there are no errors in x coordinates and that all the errors are in x' coordinates. Let $\tilde{x}'$ be the location of the true correspondence for x while x' is the measured correspondence. We can write

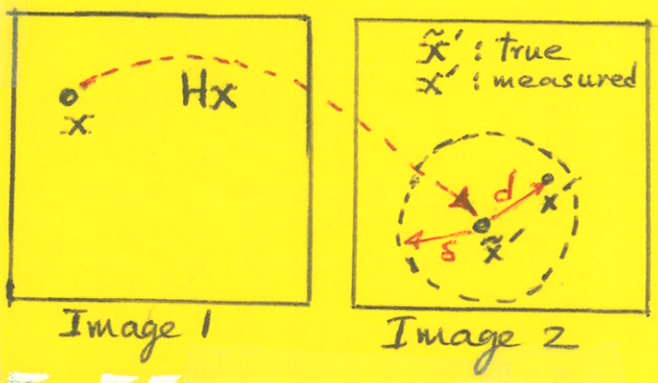
$$\text{Prob}(x') = \frac{1}{2\pi\sigma^2} e^{-\frac{d^2(x', \tilde{x}')}{2\sigma^2}}$$

To make explicit the dependence on the homography H, we write:

$$\text{Prob}(x'|H) = \frac{1}{2\pi\sigma^2} e^{-\frac{d^2(x', Hx)}{2\sigma^2}}$$

Obviously, $d^2 = (\Delta x)^2 + (\Delta y)^2$

- Other heuristic considerations used for setting δ are dictated by the spatial resolution in one or both of the images. Let's say you are constructing a homography from an NLCD (National Land Cover Database) to a photograph. The δ you choose must take into account the fact that each NLCD pixel represents a 30 m x 30 m square of the ground.



- I'll now talk about how to set the parameter N for the number of trials. We want to set N large enough so that in at least one trial the minimal set (of size n) of the correspondences chosen randomly to construct the homography estimate H does not include any outliers.

- Let ϵ be the probability that a randomly chosen (x, x') is an outlier. [If you know that roughly 10% of the correspondences are false, $\epsilon = 0.1$.] Then $1-\epsilon$ is the probability that any single correspondence is a true inlier. Therefore, the probability that all n correspondences chosen for estimating H in a given trial are all inliers is $(1-\epsilon)^n$. This implies that, in a given trial, the probability that at least one of the n correspondences used for calculating H is an outlier is $1-(1-\epsilon)^n$. Therefore, the probability that every one of the N trials will involve at least one outlier in the calulation of H is $[1-(1-\epsilon)^n]^N$.

Therefore, the probability that **at least** one of the N trials will be free of outliers in the calculation of H is $p = 1 - [1 - (1-\epsilon)^n]^N$. We set N so that $p = 0.99$.

- Since $[1 - (1-\epsilon)^n]^N = 1 - p$, we can also write $N = \log_b(1-p)$ with $b = 1 - (1-\epsilon)^n$.

  Equivalently, $N = \dfrac{\ln(1-p)}{\ln[1-(1-\epsilon)^n]}$

- To be able to conduct N trials by selecting a fresh set of n correspondences for calculating H in each trial places constraints on the minimum number of correspondences that must be available between the two images. Let n_{total} be the total number of available correspondences. Choosing n correspondences at a time for each trial means that you'll be able to carry out at most $\dfrac{n_{total}!}{(n_{total}-n)!\,n!}$ trials.

- That takes us to the third parameter of the RANSAC algorithm, M. Recall that M is the minimum value for the size of the inlier set for it to be considered acceptable. We want M to be roughly the same as the number of true inliers in the data. So if ϵ is the probability that a correspondence is an outlier, and if n_{total} is the total number of correspondences between the two images, then $(1-\epsilon)\cdot n_{total}$ is the number of inliers in the data. We set $M \simeq (1-\epsilon)\cdot n_{total}$.

Automatic Computation of Homography

- The following steps go into automatic calculation of homography between two images:
  - You first extract interest points and their descriptors from each image. If your interest point extractor is relatively simple, such as the Harris corner detector, you can directly use the gray levels in a WxW window around as its descriptor.
  - You compare each interest point in one image with all the interest points in the other image using an appropriate similarity criterion. With SIFT and SURF interest points, the similarity criterion can be the Euclidean distance between their descriptor vectors. With Harris corner points, similarity can be established with either the SSD or the NCC criterion (the latter is better).
  - All the (x, x') pairs that pass the similarity test constitute your putative set of correspondences. This set will usually be a mixture of noisy and false correspondences.
  - Apply the RANSAC algorithm to find the best inlier set of correspondences. For each trial, use more than 4 but less than 10 randomly chosen correspondences to calculate a homography $(4 < n < 10)$ using the Linear Least Squares method.
  - After going through N trials, retain the homography with maximal inlier support and refine it with the inliers.