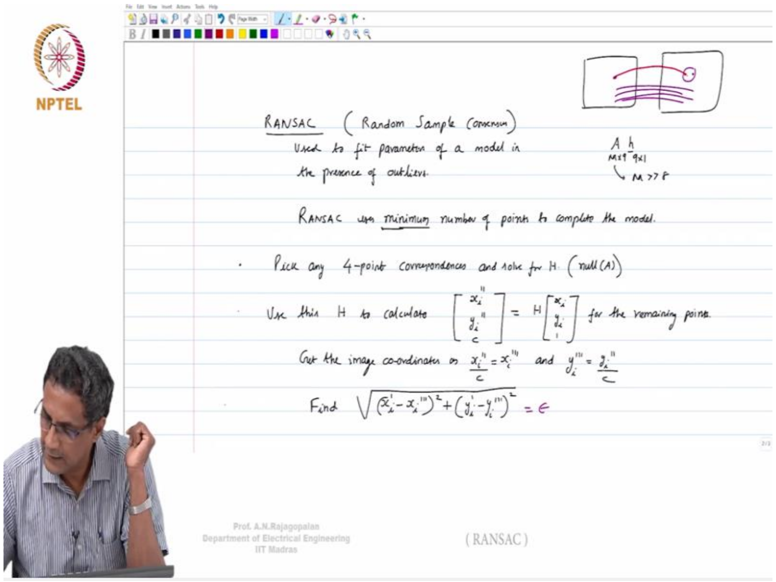
Image Signal Processing

Professor A. N. Rajagopalan
Department of Electrical Engineering
Indian Institute of Technology, Madras

Lecture - 14
RANSAC

Now you might ask, now what if I have, if I have more, more such point correspondences.

(Refer Slide Time: 00:23)



RANSAC (Random Sample Consensus)

Used to fit parameters of a model in the presence of outliers.

$A_{M\times 9}\, h_{9\times 1}$

$\searrow M >> 8$

RANSAC uses minimum number of points to complete the model.

- Pick any 4-point correspondences and solve for H. (null(A))
- Use this H to calculate $\begin{bmatrix} x_i'' \\ y_i'' \\ c \end{bmatrix} = H \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}$ for the remaining points.

Get the image co-ordinates as $\frac{x_i''}{c} = x_i'''$ and $y_i''' = \frac{y_i''}{c}$

Find $\sqrt{(x_i' - x_i''')^2 + (y_i' - y_i''')^2} = \epsilon$

Prof. A.N.Rajagopalan
Department of Electrical Engineering
IIT Madras

(RANSAC)

So, what is, what is done is normally, people use what is called a RANSAC, this is not the ransack that we use in English. So this stands for Random Sample Consensus, Consensus. And it is not really anything to do with homography. RANSAC is a, is basically an algorithm developed in the CS community, and you can use, you can use it wherever you want. It is supposed to be, it is supposed to give you robustness against outliers.

So what it basically means is, if you have data points and if you want to kind of fit some kind of a model, so, and if there are outliers in that particular, particular say, data points that you have, there are outliers, then you want to kind of be aware of the outliers and you want to come up with some kind of a robust model fitting, then RANSAC you can use and we will show how you can use it in a homography setting.

But this is not developed for homograph and anything. But then people in, people that do computer vision use RANSAC in order to be able to arrive at a robust homography, more so because when you do this point correspondences, if you, if you just take all of them, see one way to do would be to simply take all of them and then write in which case you will get A h.

So suppose you were to build this matrix A h and suppose there is some M cross 9 where, where M is a number much larger than 8. So, let us say I have got 40 point correspondence or 50 point correspondences, I can just put them all as rows. But if you directly solve for this if some of those point correspondences are wrong, which can happen, because it is only some algorithm that it is trying to figure out which point has gone where.

Suppose it tells you a wrong correspondence, then it can actually completely fail your homography estimation. Your h will kind of, say take a hit if there are outliers. So, the idea is that, idea is that you find as many safety features as you can, let the algorithm, that SIFT code, let it run, let it sort of return to you as many features as it can find. Out of those, you want to find out homography in a kind a robust way, rather than throwing all of them into some A h and trying to solve for some.

So ideally, you will ask for that h such that norm of A h is as small as possible, because you may not even be able to satisfy A h equal to 0. So you will ask for some h such that norm of A h is as small as possible. But, but then one actually ask that eventually, but not straightaway, one first accumulates all the inliers.

Now, that accumulation of inliers is actually typically done through RANSAC and I know I will, let me just write down. At this point of time do you have any this one, you have anything to ask? Now this is used to fit parameters of a model, parameters of a model in the presence of outliers, presence of outliers. And let me write down a few points. So what this also does is RANSAC uses minimum number of points, uses minimum, minimum number of points to actually complete the model, to complete the model.

So now the way it works here is, see, for example, we know that in our case to compute a homography, the minimum number that I need are four-point correspondences. So the way this algorithm works is there is something that you will actually implement. Let me just write down the steps. So what you will do is, you will pick any four, four-point correspondences at random, that is what is called Random Sample Consensus.

So four, pick any four-point correspondences and solve for h, and solve for h. So this you will do using a null of A, null of A. A is that matrix that had those spatial coordinates, which you know because somebody has already told you what those point correspondences are, so the SIFT has told you what four points so you just pick them and then it is often null of A.

Now, what you do is from this you get your h. Now use this h, use this h to calculate, h to calculate, let us say, and then I am going to use some funny sort of a notation here, but you will know why. So, xi double dash yi double dash and some let us say c, when h is acted upon, upon xi yi 1 for the remaining points. So what this means is if I had M points, then M minus 4 on those you apply this h, that is why we have a consensus. A consensus means what? Agreement.

So you are trying to see whether the other points are agreeable to this homography or not for the, the remaining points, remaining means the n minus 4 or whatever, other than excluding these four guys. Now, what do you do, get the image coordinates as, because it is still in the homogeneous form, get the image coordinates as xi double dash by c, let us call this, I am calling this as xi triple dash, hopefully, there is no xi four dash, and yi triple dash is equal to yi double dash upon c.

Then what do you do? You find again this root of xi minus xi triple dash square because yi minus yi triple dash square. See, what is actually means is the following. What it is saying is there is somebody that has already told that for, that for this xi yi, wait a minute, this should be xi dash, I kind of sort of wondering. This should be xi dash and yi dash, not xi yi.

So see, your SIFT code has already told that xi matches with xi dash, xi goes to xi dash, yi goes to yi dash. Now you have actually computed some homography and now on the remaining points you are applying h on xi and, and let us say, so when it x acts on it, it need not give you back xi dash, it will give you something, it could give you something else xi triple dash, yi triple dash.

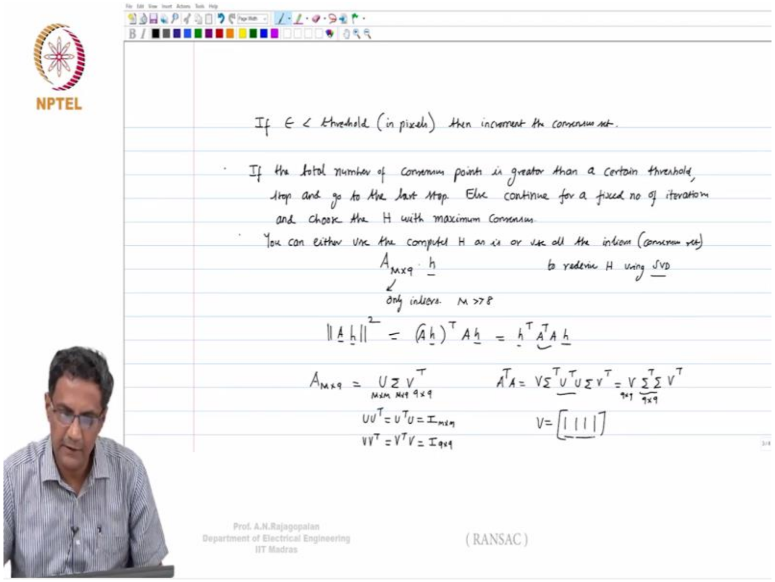
You are trying to see whether, whether you are landing close enough to where the SIFT said this point matches. Are you able to see that? So it is like saying that you have this image and SIFT has said that, let me give some color here, so let us say that SIFT says that this point matches with that, now your computed homography from some other four guys, we have not used this at all. You have used this, this, and something else. So we have used these four points to compute our h.

Now we want to know whether this h is agreeable for the other points or not. So what we are doing is, we are applying h on this guy which is your xi yi, it will take it somewhere, but no it could, so what will happen? What will happen is it could land somewhere around this point, we hope it will, we hope it will land on xi dash yi dash, which is what our SIFT test said it should be.

But then this homography need not be the right one, we do not know which is the correct one. So what will happen? It will land somewhere near xi dash yi dash, so that is a coordinate xi triple dash yi triple dash, where the homography takes it. The SIFT says it is this, the homography probably says it is going there. So now we are trying to find out how far away these two points are and as long as these two points are close enough, you say that okay, this point, this homography is in agreement for that point.

And then you start incrementing the consensus set, so the consensus is like, how many points are in-line with this homography. That is what you, so then what do you do? Then if, now let us call this as epsilon. Let us say that, let us say it is equal to some epsilon.

(Refer Slide Time: 09:36)



If $\epsilon <$ threshold (in pixels) then increment the consensus set.

- If the total number of consensus points is greater than a certain threshold, stop and go to the last step. Else continue for a fixed no of iterations and choose the H with maximum consensus.
- You can either use the computed H as is or use all the inliers (consensus set)

$A_{M\times 9}\,\underline{h}$ to rederive H using $\underline{SVD}$

$\swarrow$ only inliers. $M >> 8$

$$\|\underline{A}\,\underline{h}\|^2 = (A\underline{h})^T A\underline{h} = \underline{h}^T \underbrace{A^T A}\,\underline{h}$$

$$A_{M\times 9} = \underset{M\times M}{U}\,\underset{M\times 9}{\Sigma}\,\underset{9\times 9}{V^T} \qquad A^T A = V\Sigma^T \underline{U^T U}\Sigma V^T = \underset{9\times 9}{V}\,\underbrace{\Sigma^T\Sigma}_{9\times 9}\,V^T$$

$$UU^T = U^T U = I_{M\times M} \qquad V = [\,|\;|\;|\;|\,]$$

$$VV^T = V^T V = I_{9\times 9}$$

Prof. A.N.Rajagopalan
Department of Electrical Engineering
IIT Madras

(RANSAC)

Then if, again there will be some kind of heuristics here. You will have to choose a threshold, and typically, this will be like five pixels or something normally, but it can change, this is in pixels. This is not some intensity, we are not going to be looking at intensities, we are looking at where the coordinate is going, in pixels, then increment the consensus set, then increment the consensus, consensus set.

Then what do you do? If the total number, the total number of consensus points, of consensus points, is greater than a threshold, again that will be a threshold that you have to kind of see, decide, people typically think 80 percent is okay, but again, know this can change, it is greater than a certain threshold to be chosen by you.

So in that sense, all of you need not arrive at the same answer, because somebody might have used 70 percent thresholding, somebody might have set 80 percent, so a certain threshold. So stop and go to 8, what is that, go to the last step, go to the last step.

Else, so suppose, suppose you have still not, so for that homography you have not been able to reach this kind of a threshold, so what do you do, else continue for a fixed number of iterations. That means now you pick another, another four-point correspondences, compute a homography, run. Else continue for a fixed number of iterations, number of iterations.

So, normally what basically people will do is, since they want to, since they do not want to be trying out everything exhaustively, they will just fix a certain number of iterations till which they will wait, either the iterations will get over so you come out or you set or at some homography, you may be able to meet the consensus, consensus threshold and then you come out and choose the h with max.

That that anyway, you would have done here itself, whenever your consensus, whenever you reach the consensus thing you will anyway come out. If not, you just continue for a fixed number of iterations, if you are never able to get that consensus, it is okay. You just choose the one with the maximum consensus, choose the H. Among those, so none of them has been able to meet your criterion of 80 percent let us say, but among those whichever gave you the maximum consensus, let us say you had 75 at something, 75 percent, just take that. H with maximum consensus, maximum consensus.

Now, in some place they will simply say that what you can do is now, whichever H gave you the consensus that was maximum, you can simply use that H, that means what? That you that H was computed with only four correspondences, you can use that, but then you might ask when there are so many other inliers sitting, now what are the inliers. Those are the other points, that are actually sitting in that set that we know have come reasonably close to their feature correspondences. So you could actually use all of them, because now you know that they are all inliers.

Now, you do not have any outliers sitting there because those are all, those all belong to an inliers set now. So what is normally done, you do not want to throw them away. So you will recompute your homography using all the points that are inliers. Not all the points, if you had 50 points, maybe for this homography you got about let us say, whatever depending on 80 percent, let us say that we have got 40 points. So those 40 points will be the only ones that will be used to re-compute our homography, not all 50 because we got outlier sitting there outside of the set.

So what you will do? So you will do A, let us say you got some M number of inliers so, so this is only inliers. So now let me write, when you can either, let me right here, you can either use the computed H, computed H by which we mean only with those four correspondences or as is, or use all the, all the inliers, that is the consensus set, consensus set in order to re-derive H, to re-derive. I mean, it may not matter so much but we should know that such a thing is possible.

In which case you will do AM cross N, multiply it. M cross 9 by h, where only inliers and we know that M is probably much, much larger than 8. Now if let us say, see this is all assuming that A has still some noise and so on. See, there may still be some noise, these may be inliers but does not mean that everything is exact because you still allowed for some sort of the threshold, you still said that as long as we are within, within a radius you are okay. So that mean what can happen is, this A can still end up, end up with a rank 9. A can have a rank 9 because, because of noise in, noise near the data.

If it so happens that all of your data is completely clean, then it does not matter whether you use 8 or use more. A will have rank 8 and there is only one homography. But normally, A will have some noise in it, nothing is exact. So, what you do is you look for an h such that norm of A h is as small as possible, let us put it this as square. So you want to pick an h such that norm A h is as small as possible because then know, you may not be able to get A h equal to 0. So the next best is look for a norm A h such that it is as small as possible.

Now which is equal to saying that you are looking at something like A h, I have wrote here, SVD. Now Singular Value Decomposition, I am sure you have all done. You will anyway do when we do image transforms but we just need a little of that now, so I will just introduce you to that. So A h transpose A h, that is what you are looking for an h such that this number, number is as small as possible and this is nothing but h transpose A transpose A h.

Now, A we know is not really a square. A is actually a rectangular matrix. So when you have a rectangular matrix, you can, you know that singular value decomposition allows you to decompose a rectangular matrix in terms of what are called unitary matrices and sort of a, sort of a rectangular, it is not strictly diagonal. So you get something like U sigma V transpose. This is the standard sort of a notation that people use for, for SVD.

So what it means is that U is, U is such that if A is complex and all, U has to be U Hermitian, we have talked about, let us leave all that. Let us simply say that UV transpose is equal U transpose U is equal to identity. And of course, in this case, the dimensions will change. M, U will be M cross M; then V is going to be 9 cross 9 and M sigma is going to be M cross 9.

So sigma is like almost diagonal matrix but except that it is not square. It is like M cross 9 and it has what are called singular values. That is why we call it a singular value decomposition. We will talk about it later. But just the way people use it to compute homography is because singular, SVD is so powerful it will keep on coming. There are so many applications of it, this is just one application.

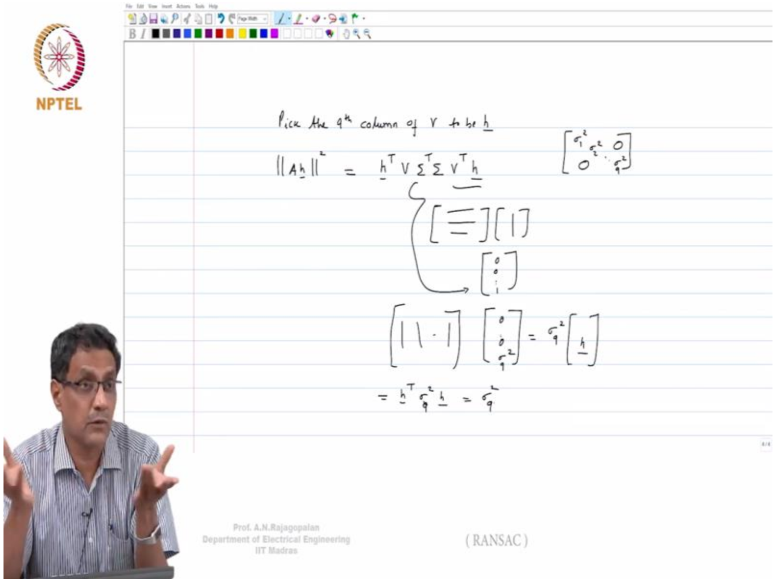
And of course, similarly, VV transpose is V transpose V is equal to identity. Of course, this I is of a different dimension, this is like M cross M, this is 9 cross 9. So the two Is are not the same and UV transpose not by identity and so on. There is if is U orthogonal, V is orthogonal. And, so therefore if you look at this A transpose A that we have here, we can use singular value decomposition to our, to our advantage now.

So A transpose A. So A transpose A will then equivalent yield you V sigma transpose U transpose; and then A, which is U sigma V transpose. Now, U transpose U, I told you is identity, so you will end up with V sigma transpose sigma V transpose. And sigma transpose is what? 9 cross. So this whole thing is 9 cross 9, which is what it should be because A transpose is 9 cross M; so 9 cross M, M cross 9, 9 cross 9; so whole thing will be 9 cross 9. V is also 9 cross 9.

So, you can think of V as consisting of these columns. Each has, each column is like it has a length 9. So now what you do is you want to you want to kind of pick an h such that this norm is as small as possible. So in order to do that, what do you do? You pick and, yeah. One more thing I should tell you that let us assume that these are arranged in an order in terms of the significance of the, of their singular values.

That means the last, let us say this guy has the highest singular value, the next highest, the next highest and last guy has a lowest singular value in the sense that whatever is that, you know singular value is denoted by sigma 1, sigma 2, so if you are looking at sigma 9 and we are saying that sigma in a corresponding to sigma 9, there is a ninth column in B. So this is like ordered.

(Refer Slide Time: 19:46)



Pick the 9th column of V to be $\underline{h}$

$$\|A\underline{h}\|^2 = \underline{h}^T V \Sigma^T \Sigma V^T \underline{h} \qquad \begin{bmatrix} \sigma_1^2 & & 0 \\ & \sigma_2^2 & \\ 0 & & \ddots \ \sigma_9^2 \end{bmatrix}$$

$$\begin{bmatrix} \equiv \end{bmatrix} \begin{bmatrix} | \end{bmatrix} \rightarrow \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} | & | & \cdot & | \end{bmatrix} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \sigma_9^2 \end{bmatrix} = \sigma_9^2 \begin{bmatrix} \underline{h} \end{bmatrix}$$

$$= \underline{h}^T \sigma_9^2 \underline{h} = \sigma_9^2$$

Prof. A.N.Rajagopalan
Department of Electrical Engineering
IIT Madras

(RANSAC)

So now if you assume that things are ordered, then what we will do is let us pick the, what do we call, let us pick the ninth, pick the ninth column. Column, not row; column of V to be h. Why? Suppose you do, suppose you pick the ninth column of V to be h, now go back to that norm A h square. So we were here at h transpose, what was that, V sigma. So V sigma transpose sigma, V sigma transpose sigma V transpose, and then h. Because A transpose A is this guy now.

Now, because of the fact that V transpose V is identity. So, what this means is that, when you multiply, so V transpose is like you can think of all these rows. There are 9 rows in V and you and you pick the ninth column of V to be h, so then what it will mean is, you will get then a vector such that, because V transpose V is identity. That means, that means it is actually so each of those vectors is orthonormal.

So which then means that what you will get as a as a vector? You will get 0 0 0. The last entry will be 1. So we will get 0 0 and the ninth entry will turn out to be 1. Sigma transpose sigma, so sigma, so this will be like, this will be a square matrix now, which values like sigma 1, square sigma 2 square, all the way up to sigma 9 square, and this is all 0. So, you can think of this sigma as sigma 1 0 0 0 0 0 0 sigma, so on. Then what will happen? So you are multiplying, so you are multiplying what?

What are you multiplying now is, you are multiplying this diagonal guy with this. So then, then after you multiply this with this, now if you multiply this with this, what you will get now? You will give you another vector, which will be 0 0 0 all the way up to sigma 9 square. Then you are multiplying it with V, which is a set of columns. We said that V has 9 columns in it. Now when you multiply a matrix like this with a vector, the I mean, you can look at the result as 0 times the first column plus 0 times the second column plus plus plus all the way up to the last entry times the last column.

So this will give you simply a vector, which is sigma 9 square, the last column. But the last column was, what do, I mean I do not know what we called it. Well, that is it, that is actually that is h by the way, that is your h. We picked the last column as h of V. So h transpose, now what have we got now? h transpose sigma 9 square h. But h transpose h is nothing but that it is again, it is orthonormal, no? It is simply 1. Sigma 9 square is only a scalar, it can be pulled out of this. Sigma 9 square, so this is like sigma square 9.

And that we know is the kind of, is the smallest singular value. This will be the smallest, if there is noise, this will be the smallest non-zero singular value. Corresponding to this, that h that you had, there was a ninth column of V that will be your h. So that h will be such that norm of A h is the smallest given this, given the set of inliers.

So basically this is the way, this is the way one finds out the homography. And now that you know how to compute homography using a RANSAC, now you can actually take images and you can stitch them.