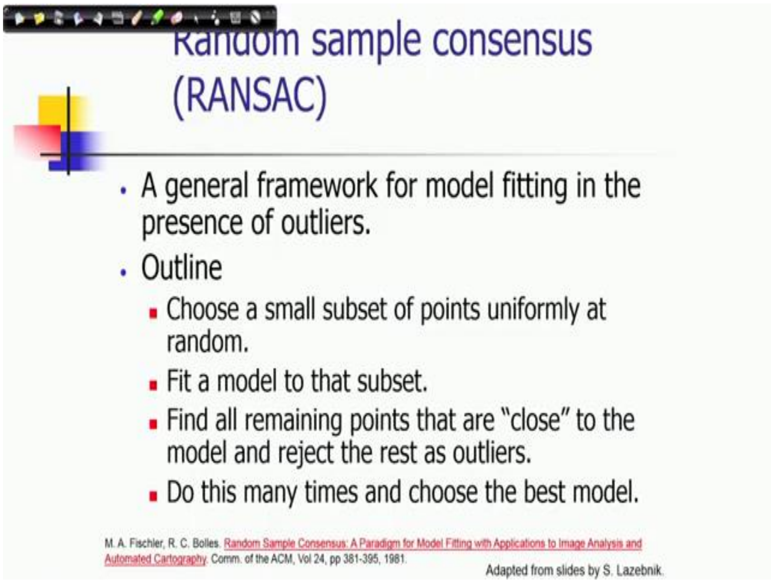
Computer Vision

Prof. Jayanta Mukhopadhyay
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

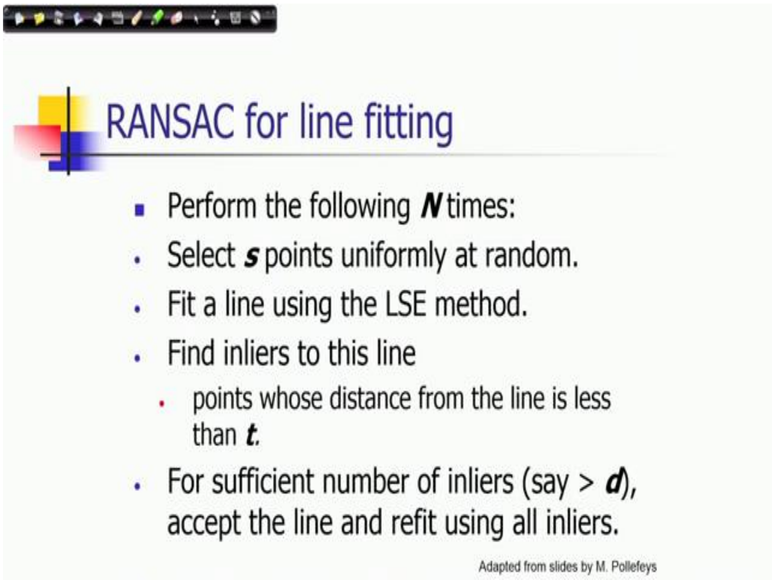
Lecture – 33
Feature Matching and Model Fitting Part – V

(Refer Slide Time: 00:19)



We are discussing about the technique of random sample consensus. And in the previous lecture, I have given a general idea how this technique works while fitting a straight line. Now, we will elaborate it is algorithm with respect to line fitting.

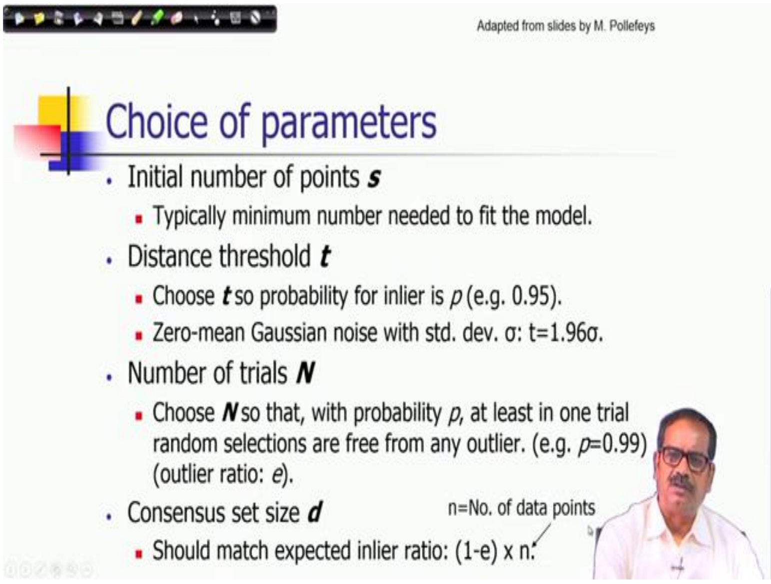
(Refer Slide Time: 00:37)



So, the steps are that as I mentioned that we have to perform N number of times the operations for choosing a good set of inlier points for fitting and initial model. And then based on an initial model and then perform refining over that set of inlier points. So, this is the step that you should select s points uniformly at random.

In the previous example, minimally I have chosen two points for fitting a straight line, but it could be more number of points. And then apply a fitting technique, for example, you can use any least square estimate method which we discussed earlier and then you find inliers to this line, the method is that the points whose distance from the line is less than a threshold value t, then that point is declared is inlier point. And if you have a sufficient number of inlier points, for example, we can have another parameter d, then you should accept the line and then again refine your model by refitting those, refitting the model with those lines with those points only.

(Refer Slide Time: 01:51)



So, there are various parameters those are involved in this process and that will affect your estimation of straight line parameters or fitting the models. So, initial number of points, for example, and as I mentioned you can choose a small number of points, randomly you have to sample those points from your data. Typically, minimum number of points that is require needed to fit the model that at least you have to choose.

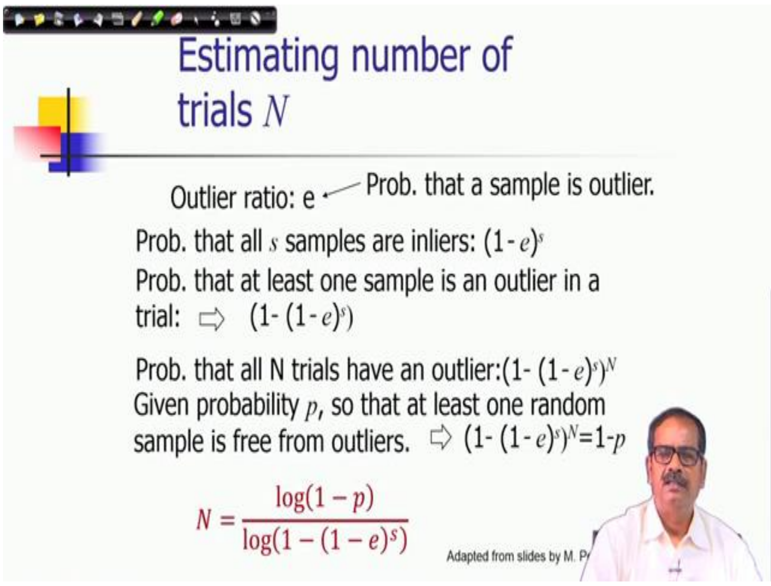
So, sometimes you may work on those that minimum number of model sorry, minimum number of sample points and apply the model fit precise model fitting over those sample point. And ones you get a model then you have to compute the distance from that model distance of other points from your straight line, so distance threshold t in this case you should choose this threshold in such a way that probability of that point should be in inlier if it is less than that threshold value that probability should be high.

For example, say 0.95 could be your target probability. In that case, if I assume that its say your error in the data is a 0 mean Gaussian noise then you can ensured if its standard deviation is σ then this threshold should be 1.96σ. Then you can ensure that this probability of no getting an inlier with this policy if this 0.95.

Similarly, number of trials that how many times you should repeat these operations for getting a model and then if you do not get any good set of inlier points then you can you then you can terminate the process. So, this number of trials you have to choose in such a way that probability of at least in one trial you should get all the points as inlier that probability should be very high. Given some data condition, suppose you know that in your data certain fraction of data is there forming outliers and you can have an estimate of that fraction. Say outlier ratio which is given in a symbol e here it could be say 0.1, 0.2 something like that.

And then the consensus set size d that is also important that how many points declared as inliers are sufficient for deciding about a model. So, that also is decided by this outlier ratio. It should match the expected inlier ratio. That means, if there are e fraction of outlier ratios outlier points then there is (1-e)*N number of inlier point is there. So, your d should be close enough to that value, d should not be small, d could be very near to that value. It should not be no significantly less than this expected number. So, these are certain thumb rules by which this parameters are selected as it is shown the N is number of data point.

(Refer Slide Time: 05:12)



So, one particular issue I will be discussing in details here, that how to estimate the number of trials, how it is related with this outlier ratio e and that indicates what the probability that a sample is outlier is. So, I can make probabilistic analysis in this way.

Suppose, we have s samples all of them are inliers. So, in that case probability that all s samples are inliers would be

$$(1-(1-e)^s)^N = 1-p$$

say individual sample is inlier probabilities (1-e) and all s samples together they would be all are inliers would be product of $(1-e)^s$. So, that is what you get here. So, which means that there exist at least one outlier in your set s. So, in a trial that would be $(1-(1-e)^s)$. So, this is what. So, which means a trial, we consider a trial is unsuccessful if there exist at least one outlier that means, I need all the points of sample should be inliers that minimally you should satisfy. So, the probability of in one trial that it is unsuccessful that means, there is an outlier in the set of s points is s shown here.

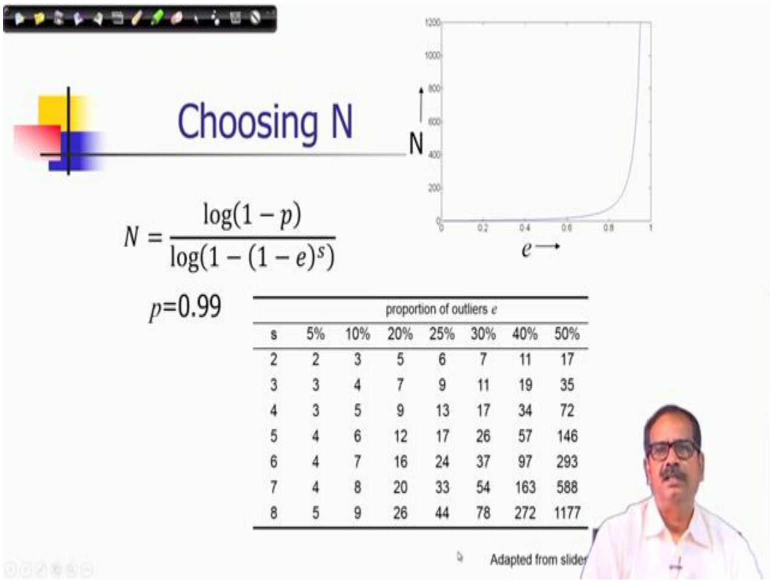
Then for N trails, that none of them contains any successful. It is a product of this N times. So, it is raise this, raise to the power N should give you that probability and 1 minus of that would give you the probability that there is at least one trial where you get all the environments. So, that is what that probability that all N trails have an outlier as I mentioned that value should be raised. So, at least one random trial, one trial is free from any outlier, if I consider that probability p that is the desired feature of your algorithm say p which could be very high value say 0.99, that should be, so your N is related to that value in this fashion.

So, the probability of failing in all N trails is $(1-(1-e)^s) = 1-p$. So, this is how the theoretical estimate of N could be found

$$N = \frac{\log(1-p)}{\log(1-(1-e)^s)}$$

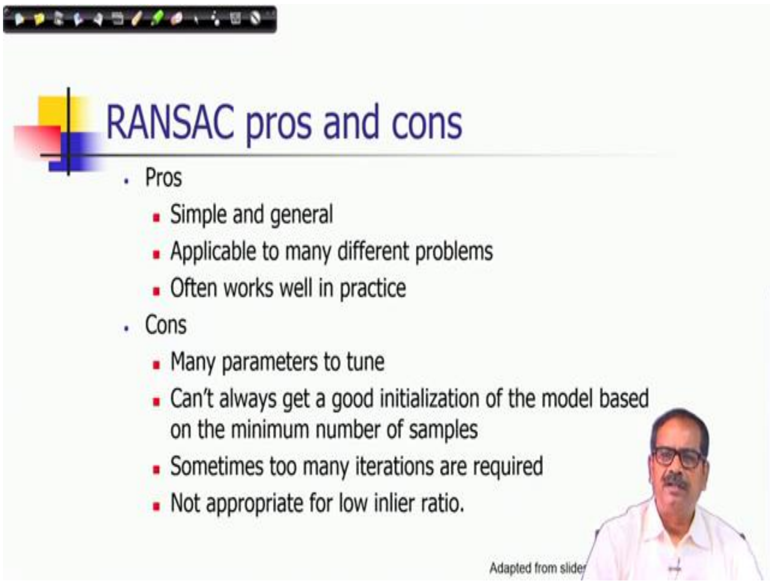
Given the data conditions where outlier issue is given and all other parameters are you know algorithms are also specified.

(Refer Slide Time: 07:59)



If I typically look at those values of N for a very high probability 0.99, we can see that as your samples s varies, the number of trial also varies and if your outlier ratio is very small then you get less number of trials. But if it is large then you get more number of trials, you need to have more number of trials. In fact, it exponentially rises with the rising fraction of outlier layers.

(Refer Slide Time: 08:33)

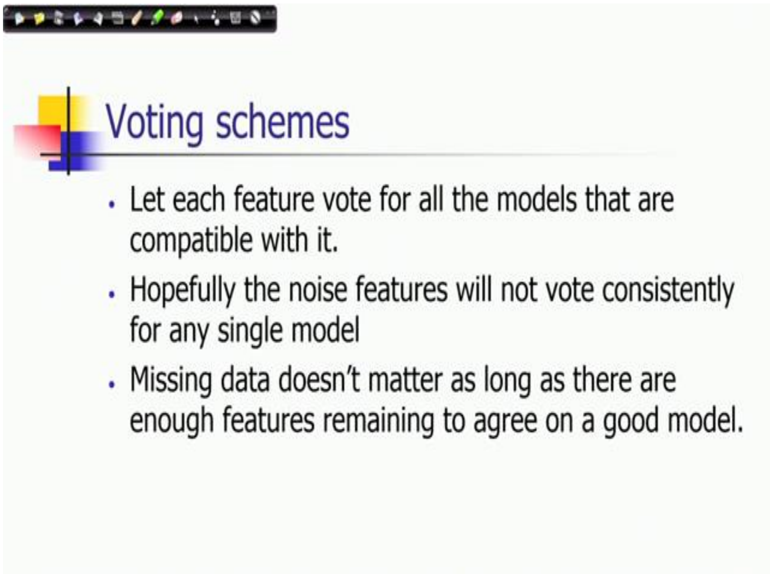


So, there are certain advantages and disadvantages of this RANSAC algorithm. Like, it is very simple and it is general. As you can see that this technique can be extended to other model estimations. For example, homography matrix or fundamental matrix, there also you when you are selecting a set of data points for fitting a model you should apply this strategy you have choosing a set of inlier data points by rejecting those data points which are not well fitted by the estimated model, initially estimated model.

And then look at the fraction of data points which are validated by model, and then again refitting it with the expanding number of inlier points. So, that is a general approach and that you can apply for any other kind of model fitting which considers now fitting data points. And applicable to many different problems often works well in practice, but there are some of the disadvantages also.

First thing as we have seen there are many parameters to tune. I have already given 4 parameters. So, tuning this parameters as you increase the number of parameters tuning becomes it tricky and non-trivial job, and you may not get a good initialisation of the model based on the minimum number of samples. So, to choose how many samples that s, value of s itself that itself is critical and that influences the performance of the model, performance of this algorithm. Sometimes too many iterations are required that means, you have lot of outliers and your N could be a large number and that is why it is not appropriate for low inlier ratio.

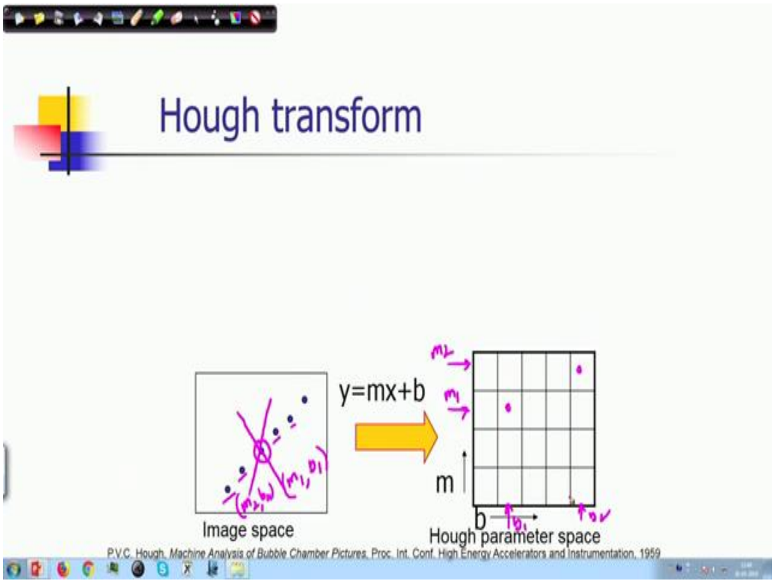
(Refer Slide Time: 10:21)



So, let us discuss the other technique flying fitting, where you have multiple lines. And then in this approach will be considering voting schemes for getting a line out of the data point and in fact, multiple instances of straight multiple instances of similar models out of a data point. In this case, these are all straight line.

So, the idea the general approach of this voting scheme is that let each feature vote for all the models that are compatible with it. So, if the future is noisy, then the consistence consistency of voting owned be there. So, only voting from the less noisy features they would be quite consistent and overall affect would be that the voted, the mostly voted no models they will consistently describe those good set of points. And also in this case, missing data, if it is missing data is also there that can be also handle because no though from the you do not get any vote for the missing data, since you are getting vote from other data and by accumulating this votes you may get a good model.

(Refer Slide Time: 11:40)



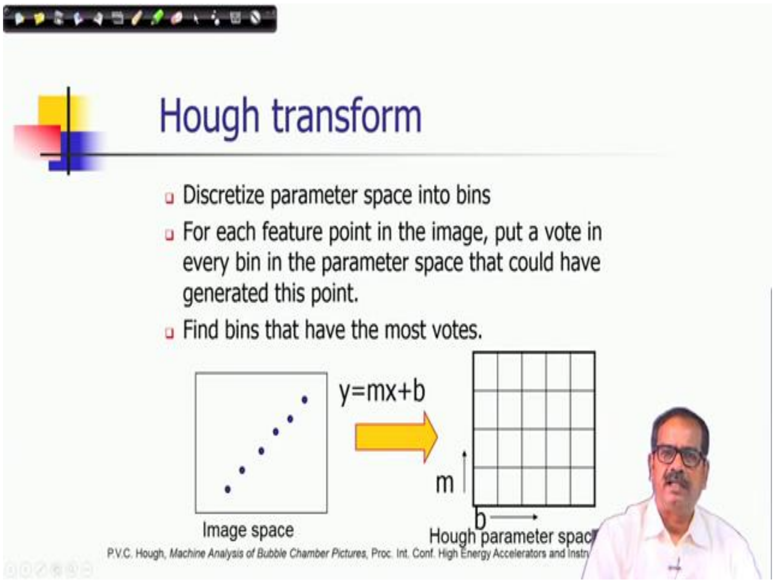
So, let me describe this particular technique with respect to line fitting using Hough transform which follows this approach. You consider a set of points which has been shown here in the image space by circular dots and as we can understand that in fact, there is a straight line passing through these dots.

So, what we are trying to do here that for every point we would like to see that what are the possible straight lines that could pass through that point, which means now if I go to the space of straight lines representations that parametric space of m and b then which pairs of m and b will be describing those straight lines which passes through any point. Just to elaborate that concept consider a particular point say this point.

Now, with respect to this point there are straight lines passing through them say this is described by in the parametric space by (m1, b1), (m2, b2) like this. So, any one parametric space suppose this cell is m1 and say this is b1, and then this cell has a vote for this point. Similarly, say this cell is m2 and say this is b2, so this also has a vote. In this way, in the parametric space you find out all possible combinations of m and b which can provide you those straight lines.

So, we will see mathematically how this could be computed. But this is idea. So, you accumulate votes for all instances of this point sets. You accumulate and you get a distribution of votes in the parametric space. So, the parameter values where these votes are quite high, those are the possible descriptions of a model through this data point. So, this is the idea and that is what we will be considering.

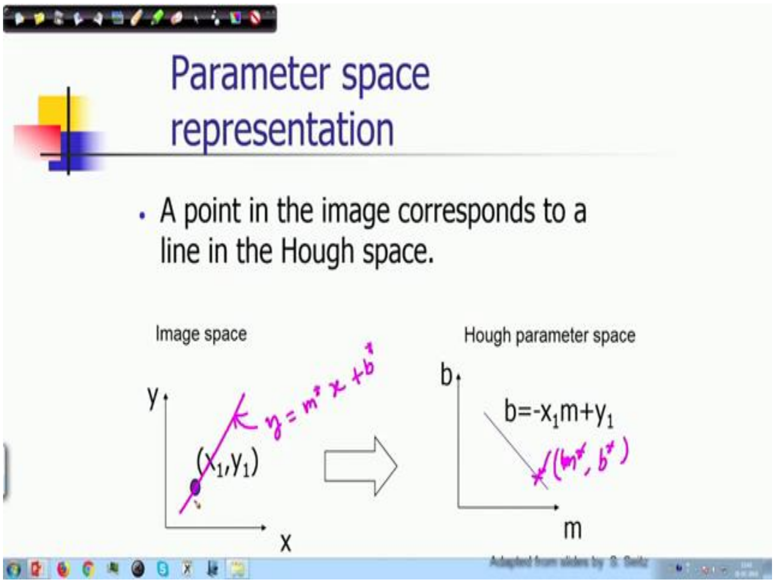
(Refer Slide Time: 13:55)



So, these are the operations which are needed when you perform this particular computations. First thing, we have to discretize the parameter space into bins, because in they continuously are from the continuous domain, but in your computations you have to discretize them, so that you can accumulate votes for each discretized cells.

Then for each feature point in the image put a vote in every bin in the parameter space that could have generated this point. So, that is what I explained here. And we have to find the bins that have the most votes.

(Refer Slide Time: 14:33)

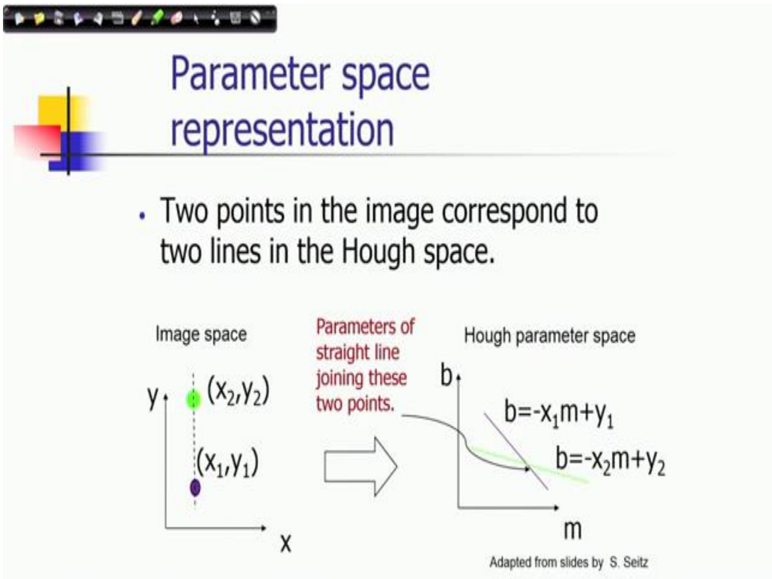


So, this will be explaining further. So, as I was telling that how straight line in the image corresponds to a point in Hough space. So, in the image if you have a straight line as it is shown here y = m1x + b1, you know that exactly it can be represented by a point in the Hough space by (m1, b1). We assume that it is it is corresponding to some discretize bin of m n.

Similarly, if I consider a point in the image, then in the Hough space it will be represented by a line because that is the relationship, say if I consider a line in the Hough space with the parameters b and m expressed in this way b = - x1m + y1, we can see that combination of b and m will give you all lines passing through (x1, y1).

With respect to our previous discussion it means that if I consider any point in this straight line it has a corresponding representation of a straight line in the image space these values, say these values some value corresponds to say m* and b*. And there exist a straight line equation y = m *x + b * which passes through this point (x 1, y1). So, that is an implication of this particular analysis.

(Refer Slide Time: 16:03)

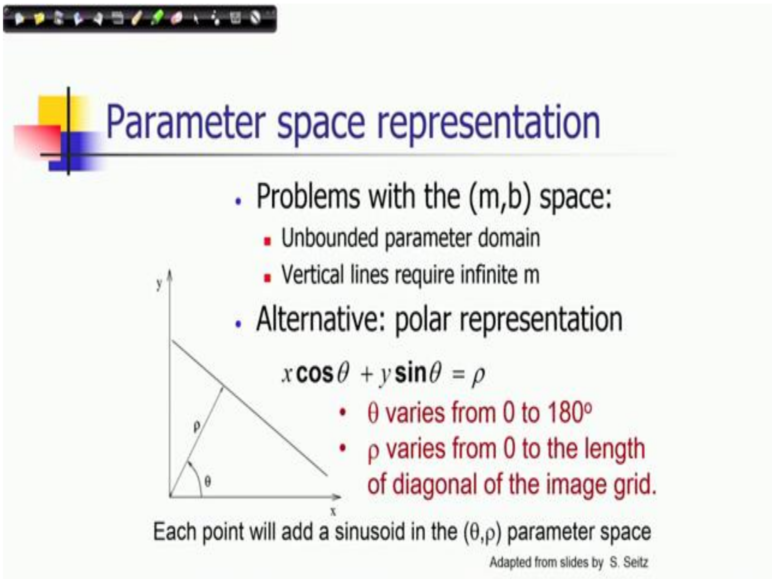


Consider another point (x2, y2) and that would also give you another straight line in the Hough parameter space which has been shown here for the green points I have shown the straight line with green colour which is given by b = - x2m + y2.

So, now you understand the importance of voting's. You can observe that the intersection point of this two straight lines in the Hough parameter space that would be the common parameter of a straight line passing through both the points and which could be a single straight line only geometrically.

Which means, if I use this the intersection of these two straight lines as a parameters of a straight line then that straight line would be joining these two points and that is how the voting comes, voting becomes importance. So, if I collect the votes you will find this intersection point will have more number of votes and since it has more number of votes you may choose that point as a solution. So, this is the general idea.

(Refer Slide Time: 17:05)



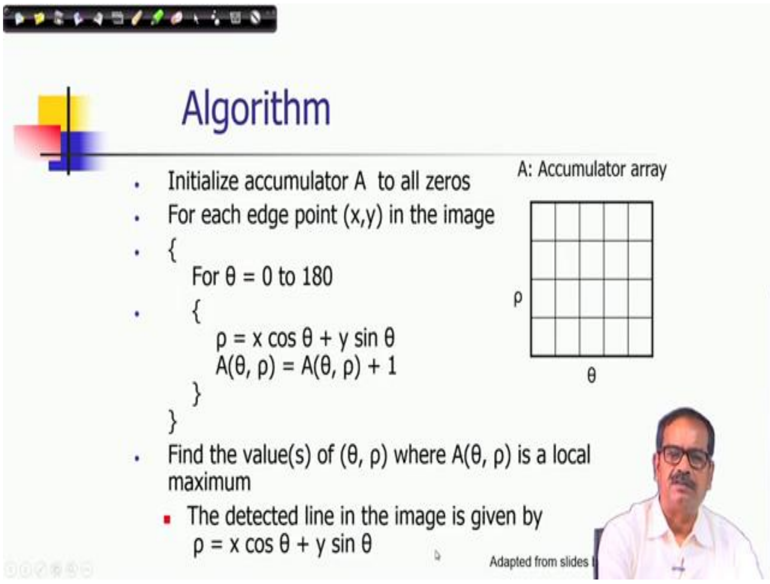
So, there are some problems with this parametric space representation. So, in we are discretizing m and b, because the values of ranges of m and b, they are bit unbounded, so particularly for m. And vertical lines require infinite m there is a problem. So, there are other representations of straight line which is more convenient like polar representation which is given in this form.

So, it is the perpendicular distance from the origin is given by ρ of the straight line and you can write it as

$$x cos\theta + y sin\theta = \rho$$

So, you see here you have parameters θand ρ instead of m and b, and the ranges of θand ρ will be in a finite range. So, like θvaries from 0 to 180 degree for an image and ρ varies from 0 to the length of the diagonal of the image grid. So, we have a very finite range. Discretization becomes easier within this finite range and you can design an algorithm.

(Refer Slide Time: 18:10)

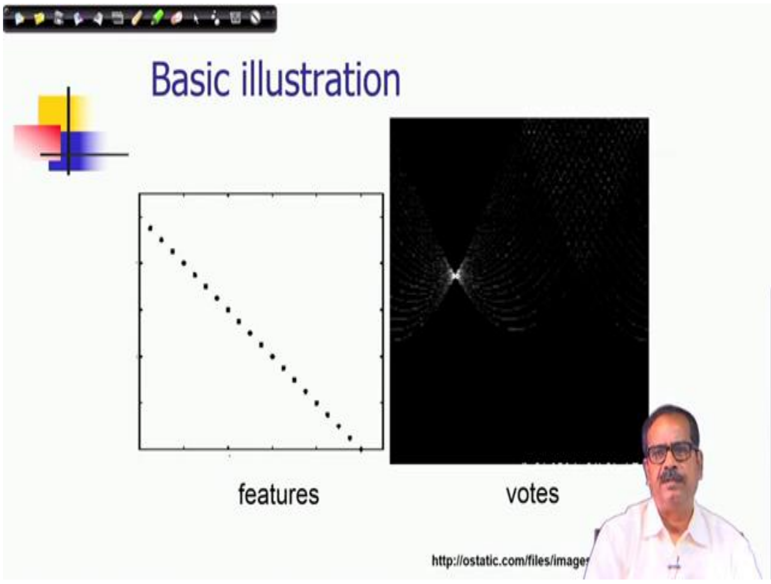


So, the algorithm would be like this that now you consider the discretized space of parameters ρ and θ which is represented by an array here all the cells of an array and let us considered name that array as an accumulated array.

So, you initialize the accumulator a to all 0s, then for each edge point (x, y) in the image you increase the counts of those accumulator cells of those arrays which denote a straight line passing through x y, which means that would be also a straight line which can be given in this form. It is not straight line in rho theta it is given in this form. So, range is theta equal to 0 to 180 degree, then $\rho = x\cos\theta + y\sin\theta$. And in that way the ρ can be found for every θ, then $A(\theta,\rho)$ should be the cell that should be increased value of that it should be accumulated.

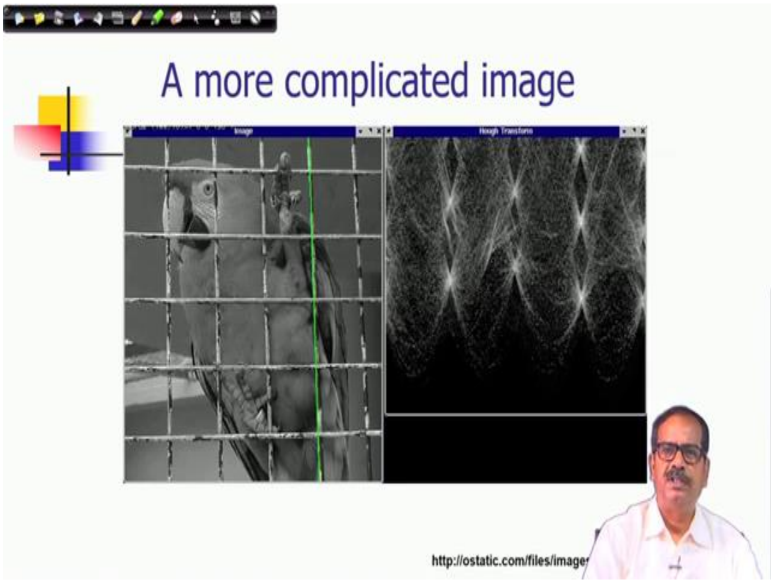
And finally, you need to find the values of θ and ρ which are local maxima because that gives you that shows that there are more votes, there are more votes in those cells considering their surroundings and that is one possible description of a straight line one possible straight line passing through some set of points in your image. And you can, in this way we can get a multiple number of lines, you can detect multiple number of lines those are passing through this set of data point. So, detected line in the image is given by this equation.

(Refer Slide Time: 19:50)



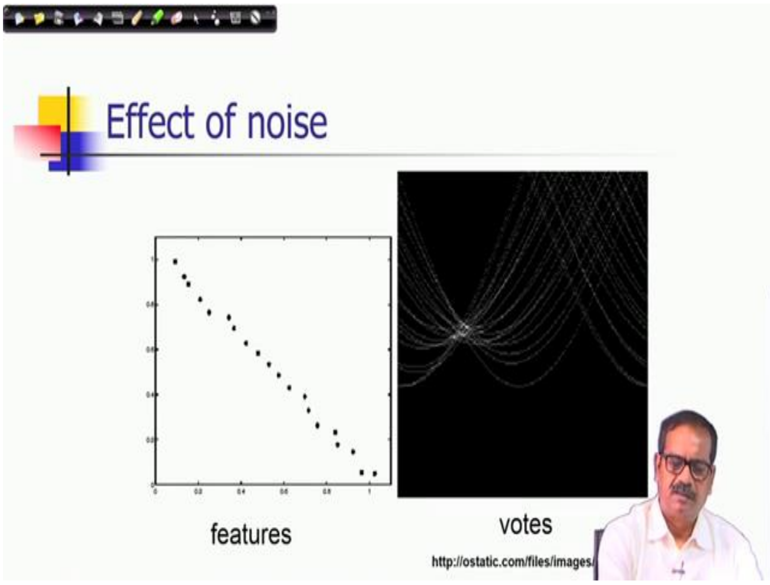
Some illustration that how this votes they look. If you have a very precise straight line passing through all the set of points only single straight line, then you can see there is only one point where you get a very high illuminations. Say, here the brightness value shows the amount of voting and since it is a single line and a very short peak you are expecting in the space of parametric space of Hough transform. So, you will get a very short point there.

(Refer Slide Time: 20:22)



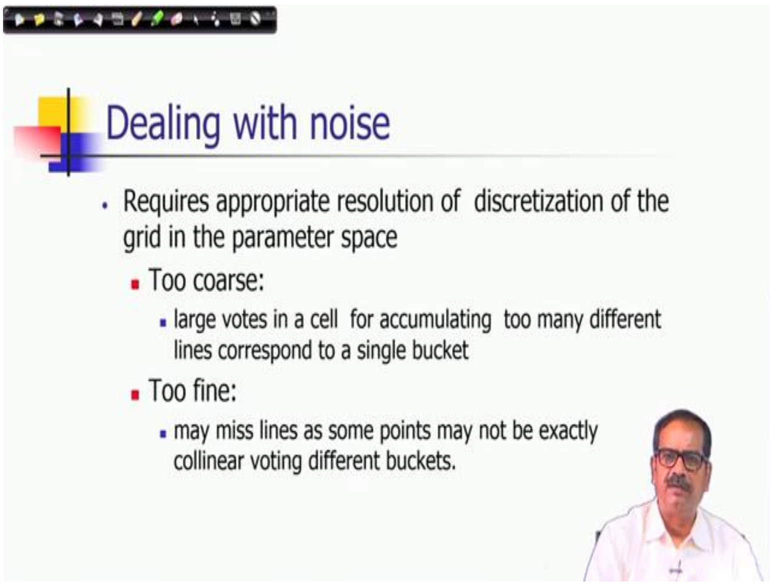
But if you have a more complicated image that means, there are too many number of lines, so each will give you some local maxima our all those voting would be, many voting's would be there from the points lying on their straight line. For example, the line showing on by the colour green would be one of this points in this case.

(Refer Slide Time: 20:45)



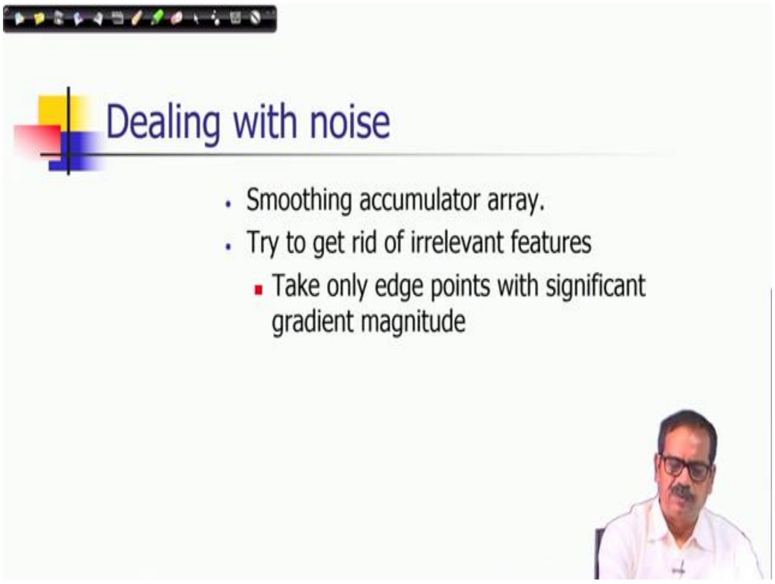
If you have noise then the shortness of that local peak would be lost, would be blurred. See, we can get some near about local maxima. So, there will be cluster of local maxima in a small space, that would be kind of nature of this.

(Refer Slide Time: 21:03)



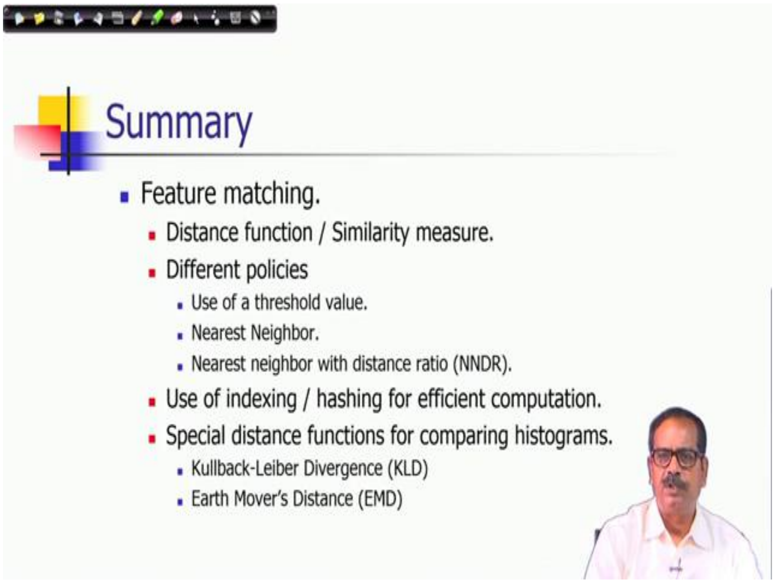
So, these are some situations and one of the major problem or major issue in applying of transform that is it how to deal with noise. In that case, you should require an appropriate resolution of discretization of the grid because if you have two coarse grid in the parametric space then large votes in a cell for accumulating too many different lines correspond to a single bucket. If it is too fine, then you a miss lines as some points may not be exactly collinear and voting different buckets.

(Refer Slide Time: 21:38)



So, you should have an appropriate resolution, and other than this you should smooth the accumulator array to reduce the effect of ripples, effect of rippling local maximas and sometimes you try to get rid of irrelevant features that means, the outliers which may cause problem. So, you fit and you take the edge point with significant. So, in this case, you consider those points which have a strong gradient magnitude and use them for your fitting.

(Refer Slide Time: 22:13)

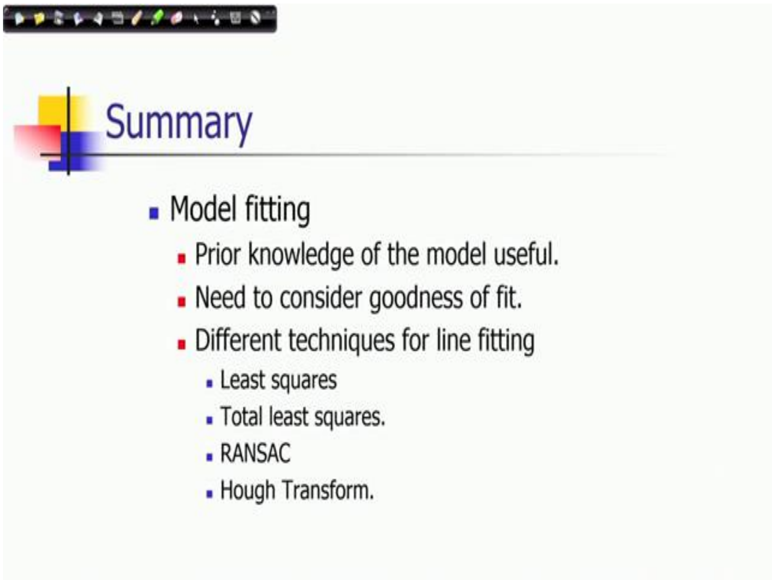


So, here we come to the end of this particular topic and let me summarise what we discussed in this topic of feature matching and model fitting. So, in feature matching we have discussed about different distance functions and similarity measure, and then we also considered different policies of matching like you can use a fix threshold value to report the matching points and within that neighbourhood of a point. Then, you can also choose in precise nearest neighbour that is as a matching point matching feature or you can use nearest neighbour with distance ratio that policy used from the query point you can find out, you can use this measure to compute a matching feature point.

The indexing and hashing to make your efficient your computation efficient and in particular with respect to these feature vectors you can exploit the geometry and you can use the indexing schemes like K-D tree or for hashing locality sensitive hashing schemes that we discussed. For comparing histograms we discussed about different some of the special distance functions other than using the norms, Lp norms with the histogram, correspondingly histogram bins values. We can use the special measures like Kullback Leiber divergence assuming histogram is a probability density function. And then, this measure is used to compare between compare two distributions and if there are close this value should be less.

And earth mover's distance also we discussed, that is a special distance function and its computation is also quite involved computations.

(Refer Slide Time: 24:14)



In the model fitting approaches, we considered only of course, here the issues of fitting straight lines given two-dimensional points, but we discussed some of the general issues a model fitting like prior knowledge of the model is useful, then we should consider the goodness of fit when you fit a model. And we discussed about different techniques for line fitting, like techniques of least squares, then total least squares, and the random sampling consensus technique or RANSAC, and also Hough transform technique. So, with this, let me conclude this lecture and this topic. In our next topic would be on colour processing.

Thank you very much for your attention.

Keywords: RANSAC, outliers, hough transform, parametric space.