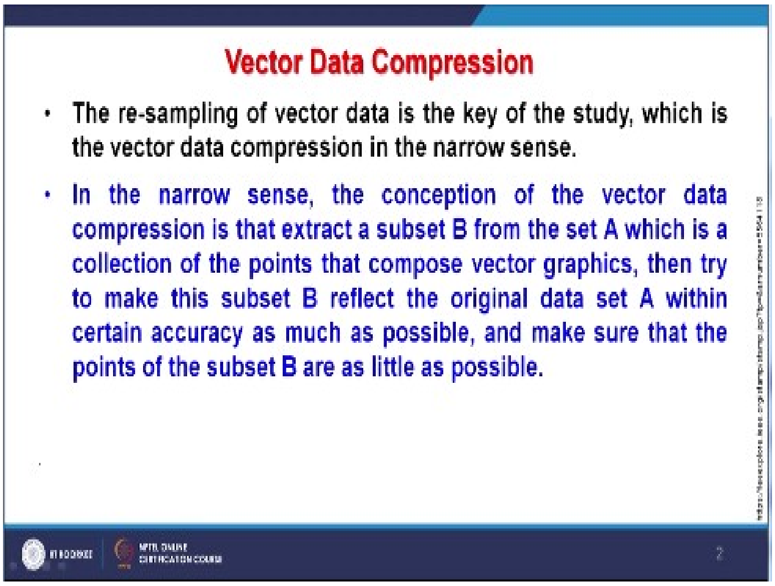
Geographic Information Systems
Prof. A. K. Saraf
Department of Earth Science
Indian Institute of Technology – Roorkee

Lecture – 09
Vector Data Compression Techniques

Hello everyone! and welcome to a new topic today, we are going to start discussion on Vector Data Compression Techniques. And though, we know that there is hardly any redundancy in the vector data but still sometimes which we will see that how redundancy also exists in the vector data and what are the concepts so far available with vector data compression techniques. So, this discussion we are going to have in much detail.

(Refer Slide Time: 01:05)



Vector Data Compression

- The re-sampling of vector data is the key of the study, which is the vector data compression in the narrow sense.
- In the narrow sense, the conception of the vector data compression is that extract a subset B from the set A which is a collection of the points that compose vector graphics, then try to make this subset B reflect the original data set A within certain accuracy as much as possible, and make sure that the points of the subset B are as little as possible.

IIT ROORKEE NPTEL ONLINE CERTIFICATION COURSE 2

As you know that re-sampling of vector data is the key for the compression and basically that leads us to the vector data compression in really narrow sense. So, in this concept of vector data compression if we extract a subset from a set A which is a collection of points that compose vector graphics. What does it mean? It is basically that if I am having village's level data of entire India that might be few lakhs points.

Now, I am not really working on the entire villages data set but I am working on, for example only for Uttarakhand state. So, why to keep the entire data set with us? and therefore we can do the subsetting of the data and try to make this subset of course which will reflect the original data set and very accurate as far as possible when we do this extraction from the original data set.

And all these points will be possible to use later on in any other our discussion or in analysis. So, that is also one way of achieving indirectly a compression. Similarly, we can take example of line data. For example if I am having a stream network for entire India; suppose if you are looking stream network for India, you may find on the internet the stream network for the entire Asia which is available and has been generated from SRTM digital elevation model and through surface hydrologic modeling.

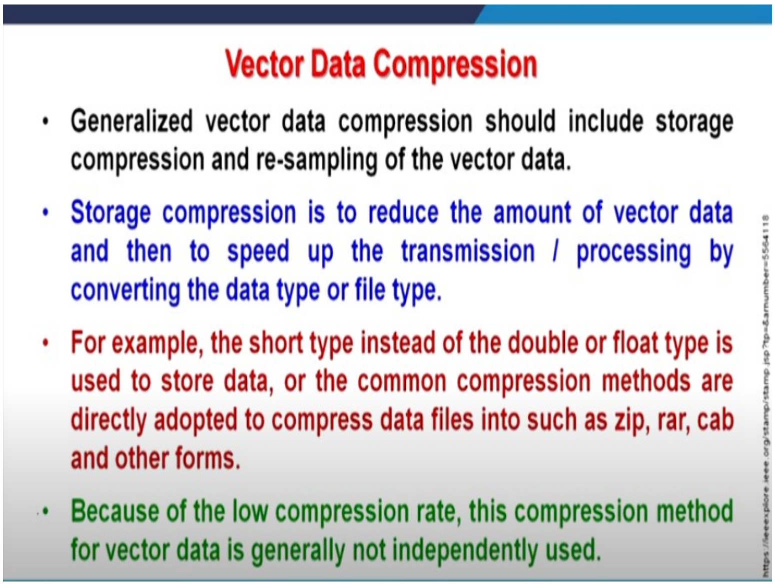
Now, whereas I might be working only in a small basin or any large basin like Ganga and I want only the stream network for Ganges. So, why I should go for the entire Asia data set? Though, I have to download and then extract the subset and of course there should not be any issue related with the accuracy. And this subset we should be able to use it and process it whenever we require for different projects.

So, vector data that means indirectly not in real sense but we can achieve certain compression but the real compression which we will see that by some means if we can reduce the number of these internodes or a number of x and y coordinates.

Suppose, I am having a boundary of India which was digitized or which was created at 1 million or 250,000 scale that means the boundary will have very fine details about boundary of India especially in coastal regions or in other places also. Whereas my target to use that boundary and I might be preparing a map in which I would like to show the boundary of India at 1 million or 5 million scale.

And therefore, if I keep using that data which has got detailed boundary of India or a polyline or polygon then there is no use. So by generalization method, I can reduce internodes and make it a simpler boundary because ultimately I am going to produce on a small scale instead of a 250,000, I am going to make on 1 million or 5 million scale. There I do not require much details about the India boundary. So, that is also we can consider as vector data compression.

(Refer Slide Time: 05:46)



Vector Data Compression

- Generalized vector data compression should include storage compression and re-sampling of the vector data.
- Storage compression is to reduce the amount of vector data and then to speed up the transmission / processing by converting the data type or file type.
- For example, the short type instead of the double or float type is used to store data, or the common compression methods are directly adopted to compress data files into such as zip, rar, cab and other forms.
- Because of the low compression rate, this compression method for vector data is generally not independently used.

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5564118

So, generalization as vector data should also include storage compression and re-sampling of the vector data. This we put as a generalization which I have just mentioned. And of course, it will produce the storage compression as well because less number of nodes when they are stored in a system compared to the original data set then obviously it is going to provide a storage compression as well.

Storage compression means here is that less space will be required to store that data. Also do not think about only storing the data; ultimately whatever the data which will reside in the GIS database, it has to be used. So if data itself is having a lot of redundancy and when I am using that data, it will take a lot of my processing time or RAM memory even for display of that data.

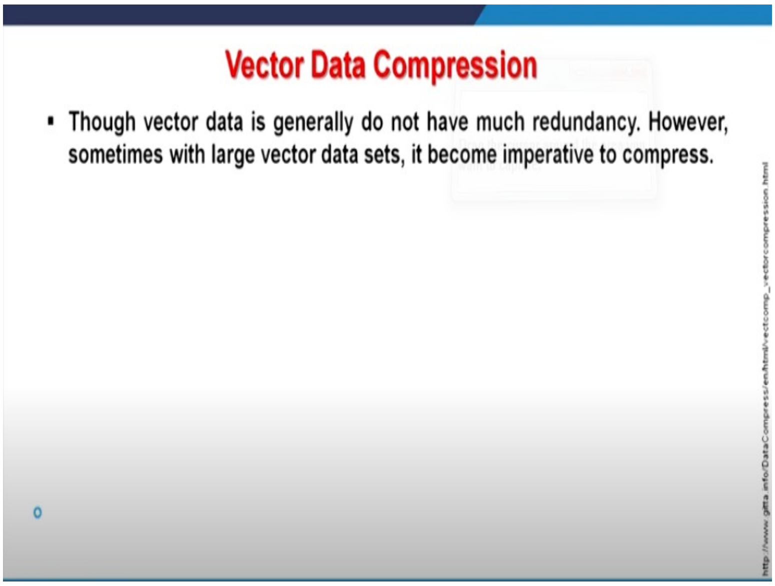
And if I zoom it or do it some operations then again it will take a lot of time too. So, it is always advisable in real practical GIS approaches or projects that first we should decide how much area I am going to cover in a particular study or project and once that is decided, area of interest; a polygon boundary, maybe a natural boundary, maybe a political boundary should be used and data using that boundaries should be extracted and should be kept separately so that I will not be using the big data file.

But I will be using a subset of original data set. Compare to a raster data compression because in raster you know that sometimes there may be very high redundancy in the data means many cells or pixels may have the same value. And therefore by some means, we can achieve better compression which we will be discussing also about the raster data compression and different technologies or methods which are available to compress raster data.

But redundancy in case of vector data is relatively less and therefore the same compression may not be achieved. For example I am having a 10 MB raster file and if I apply a compression technique whichever is available, which we will be discussing later. So, I apply x compression technique for raster data compression, instead of 10 MB that file may become 5 MB.

It depends on redundancy. Every file may not become 5 MB but a particular file; I am just taking an example. Same time if I am having a vector data of 10 MB and if I apply compression or some other tool which are available to us then probably I may not achieve 5 MB, I may get even 9 MB or 7 MB because vector data is having a less redundancy all the time.

(Refer Slide Time: 09:01)



Vector Data Compression

- Though vector data is generally do not have much redundancy. However, sometimes with large vector data sets, it become imperative to compress.

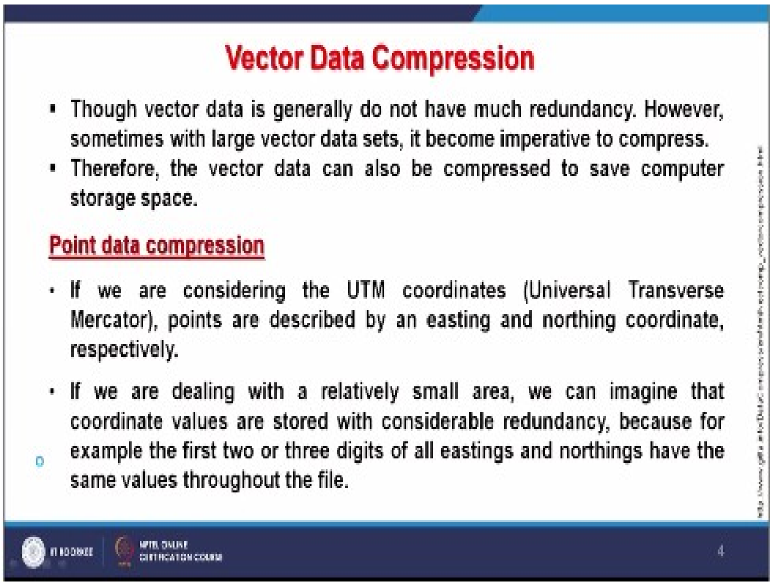
http://www.gitta.info/DataCompress/en/html/vectcomp_vectorcompression.html

So, vector data generally do not have much redundancy as I have already discussed. However sometimes with large vector data, I gave the example of a stream network of entire India which was created at 30 meters spatial resolution digital elevation model and it is having too details of any network of entire Asia. So, indirectly we can find the redundancy instead of using large file, I will use a subset of the file which belongs to my area of interest.

Anyway, whatever the compression one can think of that and try to achieve if possible so that our processing along with other data sets become more efficient. Because if I have to retrieve certain data or query the system and if it is large data for entire villages of India; first of all, it may take a lot of time. And secondly, sometimes villages may have same names.

For example, one name is Rampur which is in Roorkee or in Uttarakhand and there might be 10s of Rampur in entire India. So if I have taken a subset of Uttarakhand out of that village data then chances of getting errors or getting extra information would be reduced. So, that is why this kind of subsetting is suggested.

(Refer Slide Time: 10:40)



Vector Data Compression

- Though vector data is generally do not have much redundancy. However, sometimes with large vector data sets, it become imperative to compress.
- Therefore, the vector data can also be compressed to save computer storage space.

Point data compression

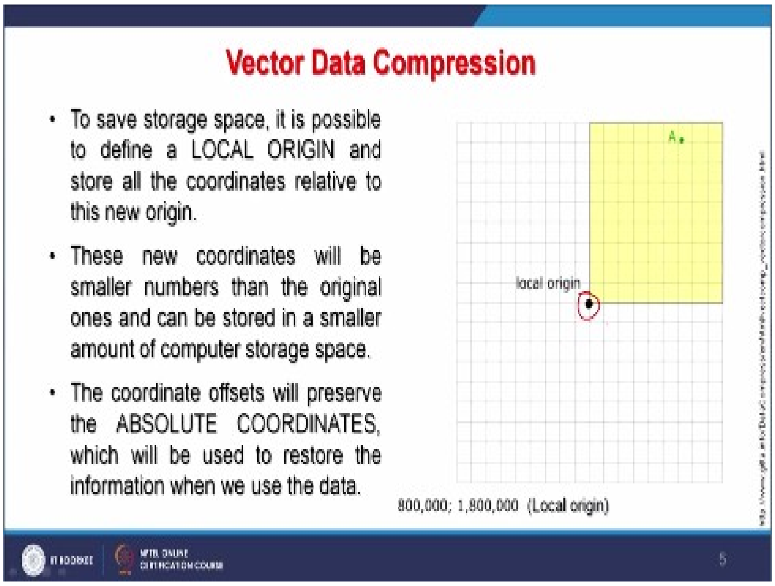
- If we are considering the UTM coordinates (Universal Transverse Mercator), points are described by an easting and northing coordinate, respectively.
- If we are dealing with a relatively small area, we can imagine that coordinate values are stored with considerable redundancy, because for example the first two or three digits of all eastings and northings have the same values throughout the file.

In case of point data if we consider the UTM projection (Universal Transverse Mercator projection) and then this data is described in terms of easting and northing coordinates that is they are in meters whereas in case of non-projected data that is in DD (Degree Decimal) and this is what we get in latitude, longitude. So, both ways are possible when we go for measurements.

And if a country is located somewhere near equator, then UTM works very well and it may give better results about the measurements like area, length and perimeter etc. So, that is why UTM is important here. Now when we are dealing with a small area and our area of interest is not big then we can imagine that coordinate values are restored with considerable redundancy because if too detail information is stored if I stored a track recorded via GPS.

Now, GPS by default is said that every second, it will take a coordinate and my movement; my velocity, my speed may not be that high. So that means I am unnecessarily storing lot of coordinates and which is not really required for my project or those track details are not required. So, again I can go for some kind of compression and can reduce the internodes and can get more smooth line and compressed vector data.

(Refer Slide Time: 12:30)



Vector Data Compression

- To save storage space, it is possible to define a LOCAL ORIGIN and store all the coordinates relative to this new origin.
- These new coordinates will be smaller numbers than the original ones and can be stored in a smaller amount of computer storage space.
- The coordinate offsets will preserve the ABSOLUTE COORDINATES, which will be used to restore the information when we use the data.

A
local origin
800,000; 1,800,000 (Local origin)
IIT ROORKEE
NPTEL ONLINE CERTIFICATION COURSE
5

So, here is how to achieve storage space or save a storage space or storage compression. Then define the local origin like here, it has been defined local origin and then store all coordinates relative to this region. And in that way also, this is basically going back to geometric system but only for a small area and this is what sometimes civil engineers or surveyors do.

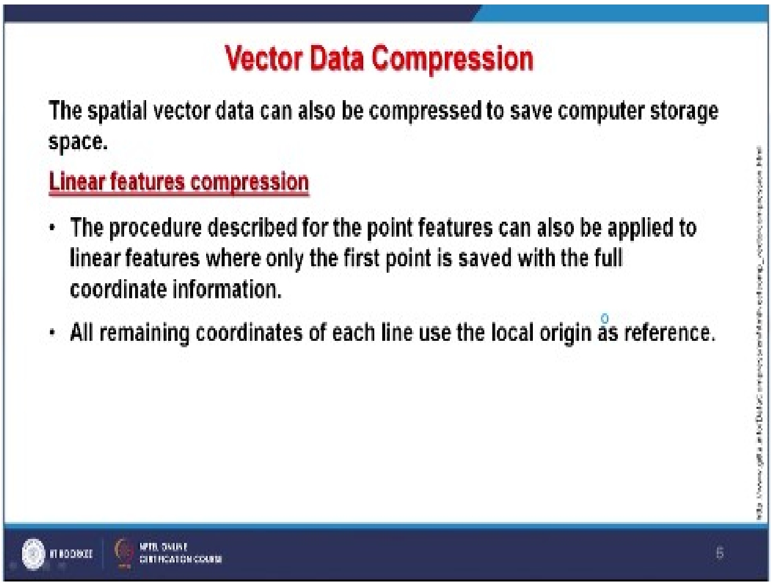
Instead of using easting-northing in case of UTM projection or latitude-longitude in terms of degree minute seconds, they are having their own coordinate system because the area maybe 1 square kilometer or maybe 5 square kilometers. So instead of writing degree, minutes that may be the same for a small area and that means I am introducing the redundancy.

So instead of that, we can have a local origin and then with reference to the local origin everything can be surveyed and plotted. So, these new coordinates having local origin are the smaller numbers than the original ones because original ones will have degree and minutes but here only with seconds or minutes-seconds, it can be work.

And then of course, it will require less storage space. So, by which we are indirectly achieving vector data compression. Now, coordinate offset will preserve absolute coordinates which will be used to restore the information when we use this data. That means I can store somewhere the offset about this information.

And then whenever I want those easting-northing or geographic coordinates then it may be converted easily but originally the data can be stored using a local origin and by which I can achieve vector data storage compression.

(Refer Slide Time: 14:46)



Vector Data Compression

The spatial vector data can also be compressed to save computer storage space.

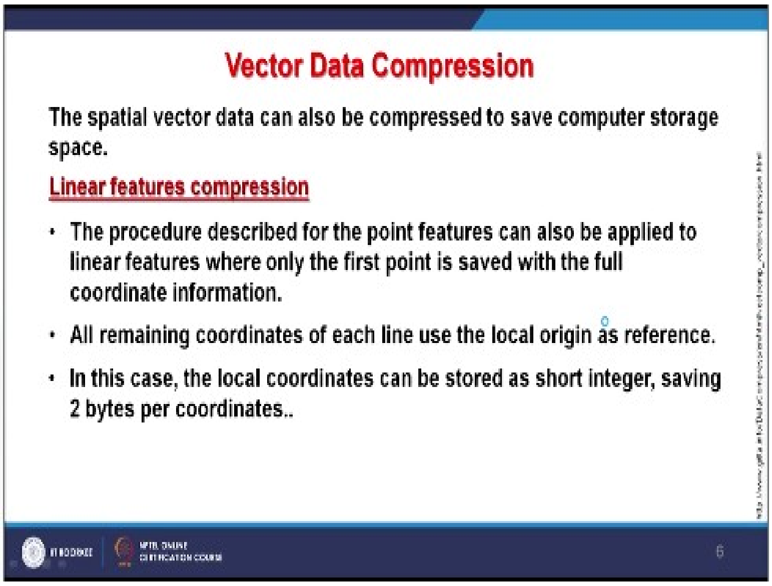
Linear features compression

- The procedure described for the point features can also be applied to linear features where only the first point is saved with the full coordinate information.
- All remaining coordinates of each line use the local origin as reference.

Now for point data we have discussed. Similarly, for line features compression and this procedure for point features can also be applied in a similar manner to line features or poly-line features where only first point is saved with the full coordinate's information and that way all remaining coordinates of each line use the local origin as reference.

So instead of using easting-northing or geographic coordinates, local coordinates can be created and they can be used in terms of distance like in meters or centimeters depending and you can store.

(Refer Slide Time: 15:34)



Vector Data Compression

The spatial vector data can also be compressed to save computer storage space.

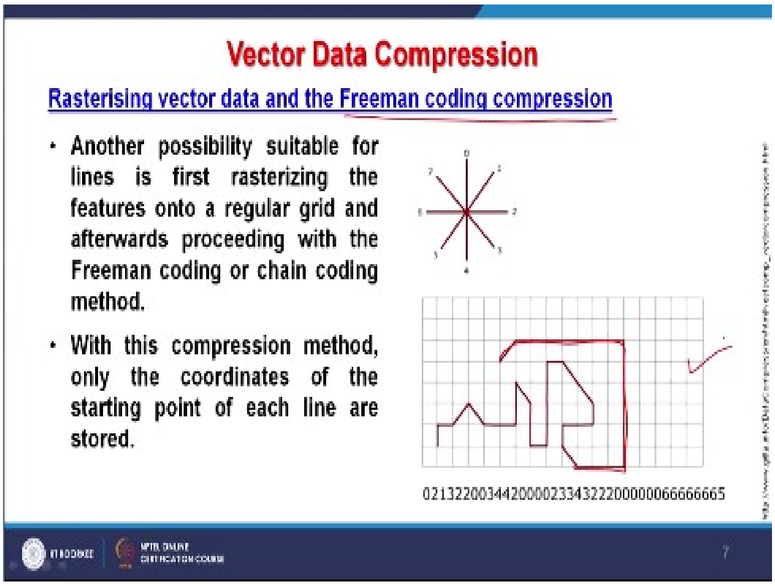
Linear features compression

- The procedure described for the point features can also be applied to linear features where only the first point is saved with the full coordinate information.
- All remaining coordinates of each line use the local origin as reference.
- In this case, the local coordinates can be stored as short integer, saving 2 bytes per coordinates..

IIT ROORKEE NPTEL ONLINE CERTIFICATION COURSE 6

So, as per point if you can save few bytes and if you are having such 1000s of points, definitely you will save some megabytes of data and that is what it is important.

(Refer Slide Time: 15:45)



Vector Data Compression

Rasterising vector data and the Freeman coding compression

- Another possibility suitable for lines is first rasterizing the features onto a regular grid and afterwards proceeding with the Freeman coding or chain coding method.
- With this compression method, only the coordinates of the starting point of each line are stored.

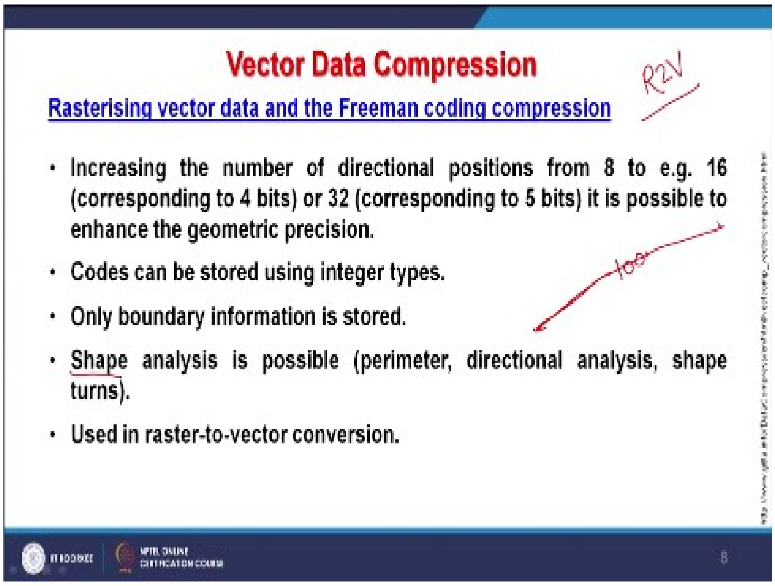
IIT ROORKEE NPTEL ONLINE CERTIFICATION COURSE 7

So, when we go for the data compression for line data then there is a technique which is known as the Freemen coding compression technique which is achieved by rasterizing vector data. Now, it is a kind of reverse process but nonetheless it will depend on the project and on the redundancy. If not much redundancy is there, one does not have to bother about the compression of vector data.

But, if there is a redundancy and it is taking a lot of time then we should go for such compressions. So, the possibility here for first to rasterize the features that is line feature in our discussion, on to a regular grid as shown here that these are the lines which are there and in background we are having raster grid and afterward this proceeding with the Freemen coding or chain coding method.

Chain coding method has also been implemented for raster data compression. So when we will discuss that part that is also will be discussed in detail. So this compression method that is chain coding method, only the coordinates of the starting point of each line are stored because rest of the things are stored as a regular grid that way also compression can be achieved.

(Refer Slide Time: 17:20)



Vector Data Compression

R2V

Rasterising vector data and the Freeman coding compression

- Increasing the number of directional positions from 8 to e.g. 16 (corresponding to 4 bits) or 32 (corresponding to 5 bits) it is possible to enhance the geometric precision.
- Codes can be stored using integer types.
- Only boundary information is stored.
- Shape analysis is possible (perimeter, directional analysis, shape turns).
- Used in raster-to-vector conversion.

100

IIT ROORKEE NPTEL ONLINE CERTIFICATION COURSE 8

And the increasing number of directional positions and from 8 to for example, 16 (corresponding to 4 bits) or 32 (corresponding to 5 bits), it is possible to enhance the geometric precision and by which also we can achieve better compression. So, codes can be stored using integer types as you know that storing integers will require less space than storing in real data or real type or in floating points.

That is why whenever we choose the precision even if we have to store data for certain analysis in real numbers then we should choose very carefully about the precision. For example if I am storing the pH value of the soil or water in my database and by default my database is having 4 precisions after decimal but we know that our instrument and the technology which we are having, we can only measure up to 1 decimal place.

So why to keep 3 decimal places extra in our database for storage? We are not going to have any values or significant value there. So, through better management of attribute data or characteristics of the field or format of the column, again we can achieve better compression even that attribute table belongs to a point data or line data. So on all fronts, we should work to save space.

This is not only saving space; it improves the quality of our data. It improves our query system. It improves our analysis also. So, redundancy in the vector data or raster data or any data like non-spatial or attribute data may give problems at later stages. So always it should we avoid and optimum things should be kept whether it is precision for the real numbers or number of nodes and internodes for a line or point data also.

So in this Freeman coding, only the boundary is stored and that is why it is called chain code kind of thing. Now when we say shape, it means we are talking about the vector data and once we are having the data in that format then perimeter directional analysis and shape turns. It is possible to analyze or estimate their length and perimeter, etc.

And this concept can also be used to convert raster to vector conversion. As we know that when we discussed this raster to vector conversion, I also mentioned that currently that technology is not available yet which can automatically do this conversion from raster to vector. Though, your software may convert from raster to vector but it will end up with very poor results.

So, whatever the methods which are available so far are semi-automatic with lot of human interventions and very well-known software which is called R2V that means raster to vector software can do a conversion but it has to be trained. Suppose I am having a survey of India topographic map in which I am having multiple layers in different colors and I want to extract only contours.

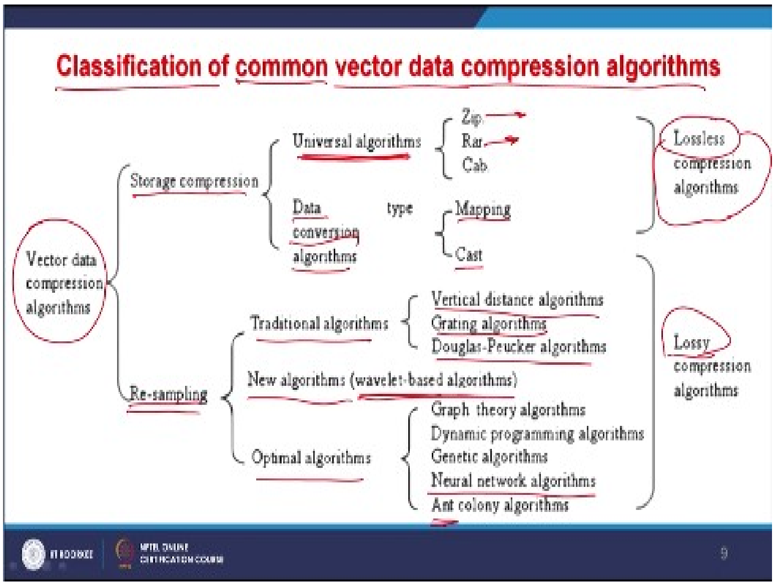
So, when I will scan that using a large format scanner basically, I am generating an image of the toposheet that is nothing but the raster and if I submit that scanned image of the toposheet to the software which is R2V then what is going to happen? It will convert but it will give us very erroneous results. So, what we need to do basically we train each and every contour line.

So we give a start to the software. This is the start of a contour line having value 100 and then the software itself by identifying pixels having same color characteristics, it will follow and digitize or vectorize and wherever there is confusion, it will stop. Now human interventions are required to go and to proceed further, you have to connect by yourself.

So if there is a contour and 100 values written here and it is there so, if I have trained this one, it will keep digitizing till it will stop here because here it is confused. So then I have to say that just ignore whatever written and restart from here and it will reach to this end. Therefore there is no fully automatic raster to vector conversion technology; neither software available but by doing this raster to vector conversion obviously; we are achieving compression because raster will always occupy the space.

So if at all it is required then with some limited options, we can convert and if possible that we always require vector data rather than initial raster data then anyway you will achieve the compression. Now, here the boundaries in this Freeman encoding compression will be stored twice.

(Refer Slide Time: 23:43)



Classification of common vector data compression algorithms

Vector data compression algorithms

Storage compression

Universal algorithms

Zip

Rar

Cab

Data conversion algorithms

type

Mapping

Cast

Lossless compression algorithms

Re-sampling

Traditional algorithms

Vertical distance algorithms

Grating algorithms

Douglas-Peucker algorithms

New algorithms (wavelet-based algorithms)

Optimal algorithms

Graph theory algorithms

Dynamic programming algorithms

Genetic algorithms

Neural network algorithms

Ant colony algorithms

Lossy compression algorithms

Now for vector data compression, so far the algorithms which are available, this is the classification of that one. So vector data compression algorithms either targets for the storage compression or re-sampling techniques and that also belongs to the generalization kind of thing. So, there is a universal algorithm. This one; you already know about these 2 are very famous zip and rar. These are compression techniques.

And not only they compress the vector data, raster data but also any data, image, video or textual data also but lot of details about these compression techniques are not available. They are copyright protected. So, we do not know much about that. Anyway, they are basically based on this algorithm which is universal algorithm.

Sometimes if you try on a particular raster file, you may achieve very good compression if you zip it but on another same size original raster file if you try, you may not achieve that compression. For example the original raster file or vector file is having 10 MB space or 10 MB memory requirement if I zip it, it may reduce to 8 MB but I am having another vector file for the same area having different theme and if I compress it, I may not achieve even 8 MB, I may achieve just 9.5 MB.

So depending on the redundancy, in case of vector; depending on the redundancy of internodes and points and through which only you achieve the compression. The good part about these zip and rar and lot of utilities win zip and win rar such softwares are available. Most of them are free or some license are also there but the best part here is that there is no loss of data and that is why they are very popular.

I mean they are non-destructive compression technique. Today I compress a 10 MB file to 8 MB and then I send via internet or email to some friend and he can unzip it and the original quality will be stored as it is. So, this is the biggest advantages by these techniques or tools which are available through this universal algorithm and that is why, they are very-2 popular.

Now, there is another way of compressing which can allow us better storage compression is data conversing algorithms. So data conversing algorithms are used for mapping and other cast type of these methods and all these are as mentioned here; lossless, non-destructive compression techniques and most of us should always check before compressing any data and deleting the original one.

Always check whether it is a lossless or really destructive; lossy. For example if I convert a raster data into JPEG then JPEG is a lossy data compression technique rather than lossless or I say is non-destructive or destructive compression technique. So, before choosing any compression techniques make sure that it is lossless.

Unless you are having the backup of original data and you do not care about the quality of the data because of certain purposes just you want to send to somebody, quality is not important then you can go for lossy compression techniques but the advantage with lossy compression techniques is that they will allow us to achieve much more compression then lossless compression techniques.

And best example I have given about the JPEG. JPEG is a file format as well as a lossy compression technique. Now for re-sampling methods, there are traditional algorithms; new algorithms especially wavelet based algorithms are there which are becoming very powerful, very useful one and then optimal algorithms are there and if we go for traditional algorithms then vertical distance algorithm, Grating algorithms and Douglas-Peucker algorithms.

And our discussion on GIS, basically we do not want to go in much more detail about these algorithms but the tools techniques which are available, they are being implemented like for example wavelet based or neural network based algorithms. So, they are everywhere and also Ant colony algorithms. So these are the tools which people are trying in almost all domains in GIS, in digital image processing or in computer to train the computers to do certain tasks in a much better efficient way.

So, these things are being done. But the important point before I go to the next slide is that lossy compressions are not preferred at all. I will give you one example though that example from raster data compression technique which is based on this wavelet based algorithm and which is called Mr.Sid and which may allow you to compression of up to 50 times.

What I mean is that if I am having a raster file of 50 megabytes using that wavelet based algorithm, I can reduce to 1 MB. But interesting part here for in case of raster that is lossless technique. And Google Earth has implemented that one and that is why we get a lot of data very easily coming on our screen through net and getting exploded on our system rather than at the host level and so that we get a fast display of big-big images or big data sets.

So, lot of such developments are taking place. I do not want to go further on this. And this brings to the end of this discussion about the vector data compression technique just to recap that in anyway vector data generally do not have much redundancy. But if your project requires that you should have a very efficient vector data too then one should explore the possibilities of 2 ways of compression.

Either going for storage compression or going for a kind of generalization, if detailed boundaries or detailed point data is not required and by which directly or indirectly you achieve quite good compression. Thank you very much.