

**Remote Sensing Essentials**  
**Prof. Arun K. Saraf**  
**Department of Earth Sciences**  
**Indian Institute of Technology Roorkee**

**Lecture - 38**

**Basic Image Compression Techniques and Different Image File Formats**

Hello everyone and welcome to new discussion which we are going to have one basic image compression techniques and different image file formats. Because as we have been discussing high spatial resolution data and other things, so sometimes these images acquire a lot of space on our hard disk on our systems and therefore, and we need to compress the these images and sometimes we want to keep them in compressed mode and it still would like to use.

So, there are many basic image compression techniques which have been developed so far and have been implemented in digital image processing. So, in this discussion, we are going to have that the discussion and also we simultaneously we will be also seeing and what are the different image file formats available to us in satellite in case of satellite images.

**(Refer Slide Time: 01:31)**

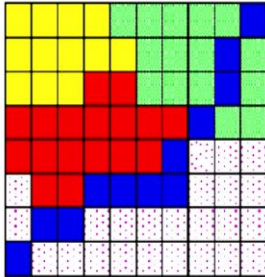
### Image

Stores images as rows and columns of numbers with a Digital Value / Number (DN) for each cell

Units are represented as square grid cells that are uniform in size

Data is classified as "*continuous*" (such as in an image), or "*thematic*" (where each cell denotes a feature type)

Numerous data formats (TIFF, GIF, ERDAS.img etc)



IT RoorkeeNPTEL ONLINE CERTIFICATION COURSE

As you know that and these images are stored as a 2 dimensional matrix and in rows and columns and the digital number of pixel value is there for each pixel and this units is always is square in shape. Whereas overall shape of an image can be square or rectangular by I am repeating this because when we go for compression techniques and then this shape of overall image will play some role there.

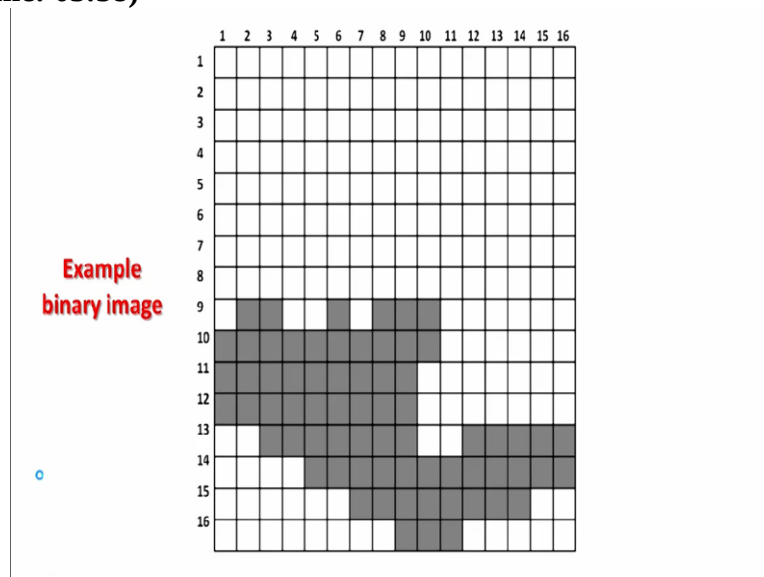
Therefore, and they mentioned to this was necessary here, whereas the square that means the unit will always remain is square and you will remain uniform throughout in terms of size throughout the image. That means, if I am handling the 10 meters space and resolution satellite image, then all pixels will represent 10 by 10 meter and grounded that means, 100 square meter media and of course.

Satellite images are described as a classified as a continuous data and sometimes also as a thematic layer and there are various image formats which are very popular all are having a specific purpose is like example, very popular one is the TIFF and TIFF Which is stands for Tagged Information File Format that means the information about the file that means, some sort of metadata is also tagged with the image file itself or image data itself.

So, it is through generally we write just TIF rather than TIFF but here it is stands for tagged information file format. So, information about the file is tagged with the image data that is what is TIFF very popular format. Another format is the GIF that is a graphic Interchange format, very popular in animations and we are sequence of images are put in one file that is a GIF file.

And it has image in very popular image processing software uses its own image format and data format that is IMG format. But it also allows us to handle other datasets like TIFF or JPEG and other formats.

**(Refer Slide Time: 03:58)**

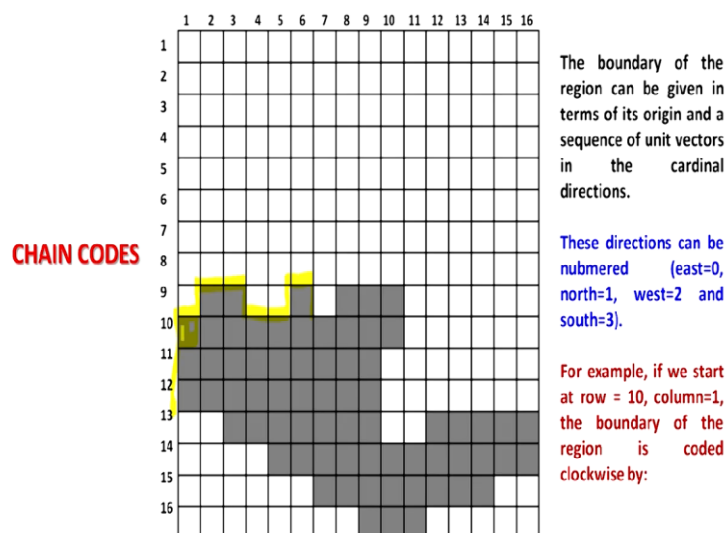


Now, in order to understand 4 basic kinds of data compression techniques. We will be using this binary image as an example, the example which I am going to take is the most simple example and be pitches in binary. But in real images of course, things will become complicated. So, in order to understand the data compression techniques basically 4 basic kinds of data compression techniques a very simple.

Our simplest possible example has been taken and these rows and lines which are you know showing rows and columns are just for our understanding in the real image. These lines are not there only the pixels are arranged in a 2 dimensional matrix. So, this is just for our understanding and there are 2 main areas are here. One is we can see is a blinker 0 area or white areas and another one is the grey areas which we can say as black areas and might be having it since is a binary.

So, it might be having value pixel value 0, other values are 1 so just binary image. So, 2 areas and just 2 major areas one is having value 1, another one having value 0 black and white that is binary image.

**(Refer Slide Time: 05:23)**



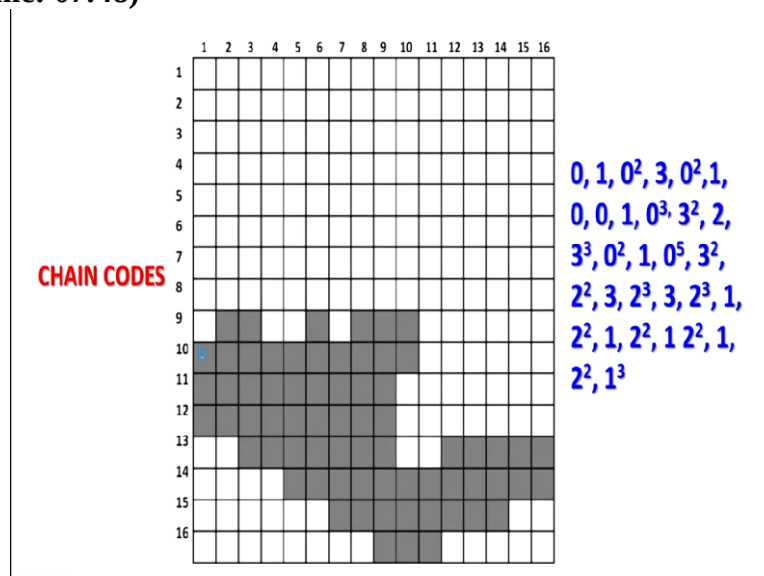
So, when we go for the first type of data compression technique, which is called chain codes, in this what we basically do that the boundary of the region, so first, it is assess that which part is less. So in this our example, though, the dark part or the black part is covering the less part of the total input image, so that boundary of that black part is coded in terms of its origin, and the sequence of unit squares.

Because here pixels are square and vector units in cardinal direction that is in north south east west. So, what we say we start from say, and this corner and goes like this like this like this and this and every part of this black part is coded as part and that origin and sequence of unit 2 factors in the cardinal direction and the cardinal direction or the codes which have been assigned may be assigned in chain codes is like for the north, it is for the north it is 1 for the east it is 0 and for west it is 2 and for south it is 3.

So starting see from east 0, and then south 3 or clockwise anticlockwise, so, 0 east, north 1, west 2 and south 3. So, anticlockwise can be assigned these are just codes for our understanding and different kinds of codes say for example, north 1 and the east 2 can be assigned, then south will become 3 and then west can become you know if 0 starts and then maximum we can go for 3 if 1 is start and we give code 1 then it can go maximum to 4. So, these are the directions which are used.

So, for example, if we started the row 10 row 10 here and go and column 1 which is column 1, because this pixel has to be coded first and the boundary of then why clockwise we will record.

(Refer Slide Time: 07:48)

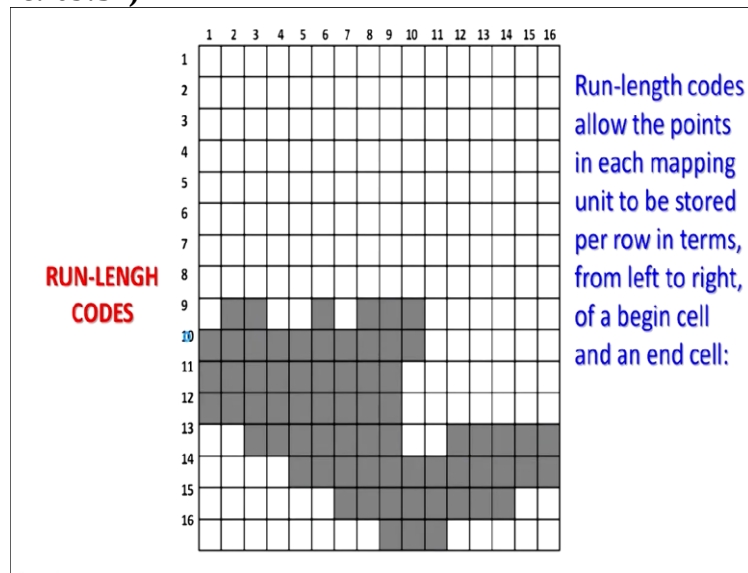


So, that means the, we start from here and this is the boundary here is coded first. That is a 0 and these are just codes these are not true, you know typical numeric values otherwise one can say how you can have a square of 0 these are just codes you know, for our understanding, so, we move one direction in cardinal direction and our east direction is 0 therefore 0 1 cell or 1 pixel here, then we 1 we go 1 up, because for the north, we have given code 1.

So, 1 up then 2 towards east and therefore, and for east direction, we are having 2 0 and 2 pixels have to be covered and therefore, 0 square in this 1, then 1 down that means towards the south and south is having code 3, only one pixel, so, like this and then again 2 pixels towards the east. So, 0 is square then 1 pixel towards the north for north we have a sign code 1. So, that is there and then go towards the east 0 and then again and likewise we cover the entire region like this.

So, this way we will cover and let us see what it in the ends it when we reaches here, then in north direction we go for the like this that mean say north direction 3 pixels or 3 bit length of 3 pixels have to be covered and that way we will cover the entire and black part or shaded part in this example putting these codes. So, then I can just store these number of codes rather than storing for each cell or each pixel of input image. So, this is what it called chain codes.

**(Refer Slide Time: 09:51)**



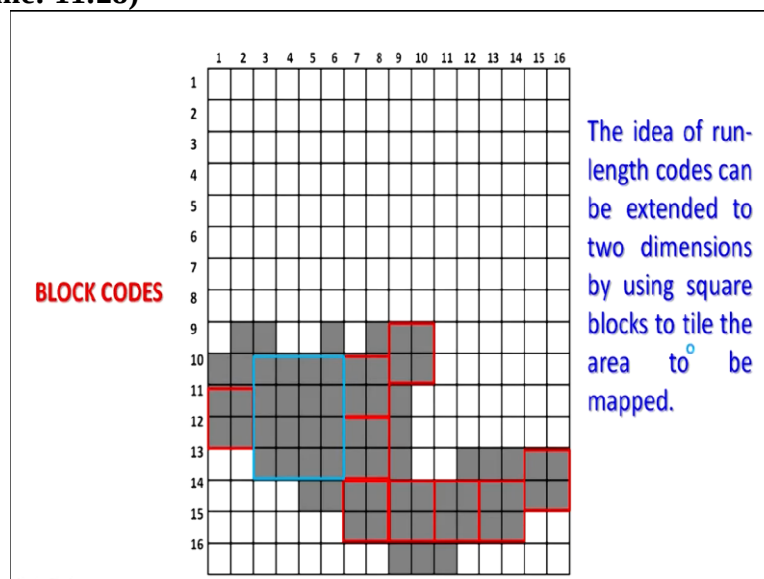
Now, let us see another data compression technique which has been implemented in many software's and sometime it is called RLE Run Length Encodes or Run Length Codes. So encodes RLE is also equally popular term here in run length code what basically is done is that, you know we run along the length basically. So the codes allow the points in each mapping used to be stored per row in terms of a left to right.

From a begin cell and to the end cell likewise, and that first row number 9, 2 cells have to be stored here, then another one and then 3 more. So, we run through the length, that means the first row 9 is chosen, then pixel number 2 and 3 is further column 2 and 3 are coded. Then

only 6 because the length here is just 1 pixel width. So, 6 - 6 then 8 - 10 and for all such rows and the coding is done.

So, for example, for 16 then there are 3 cells 3 pixels are in continuous so 16 and 9 to 11 likewise and rest is coded in run length code. So, it is another way of you know storing the shaded area or dark area and automatically the other area will also be coded that is the white area.

**(Refer Slide Time: 11:28)**

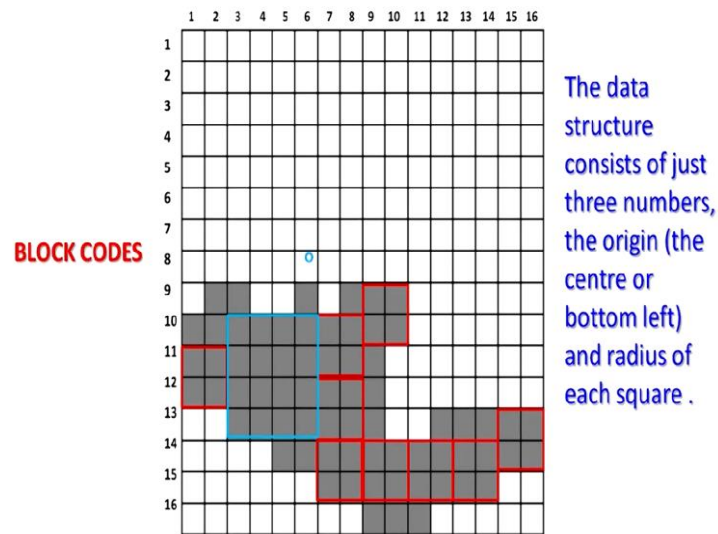


Now, there is another type of data compression technique which is called block codes, it is little complicated and one can say that, why to have a you know like 1 / 1, 2 / 2 or 4 / 4 by node 3 / 3, but when we go for these square units, then this is the either 1 square 2 square or 4 square 3 square and cannot be done in such examples. So, there is in block codes what we do that the because we are coding in each example, the dark area of this input image.

So, the first the area which is such is for the largest block and largest block in our example can be 4 / 4 is marked here in blue, then next is 2 / 2 that is marked and those blocks are marked by red. And then of course, the remaining will go they are in marked in black and they will be just 1 block that means the unit size. So, the in basically in the block codes and the idea which is we have just discussing case of run length code.

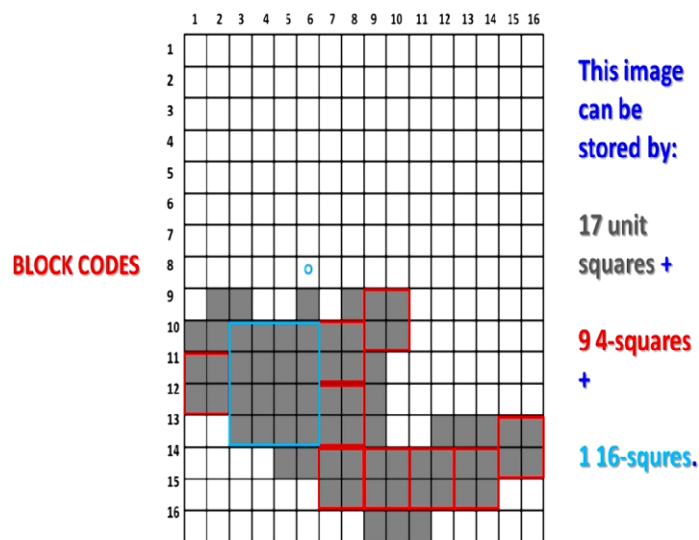
And can be extended to 2 dimensions by using square blocks is square blocks of the area which is the shaded area another example, to be mapped or to be stored. So, likewise if we code in block codes,

(Refer Slide Time: 13:04)



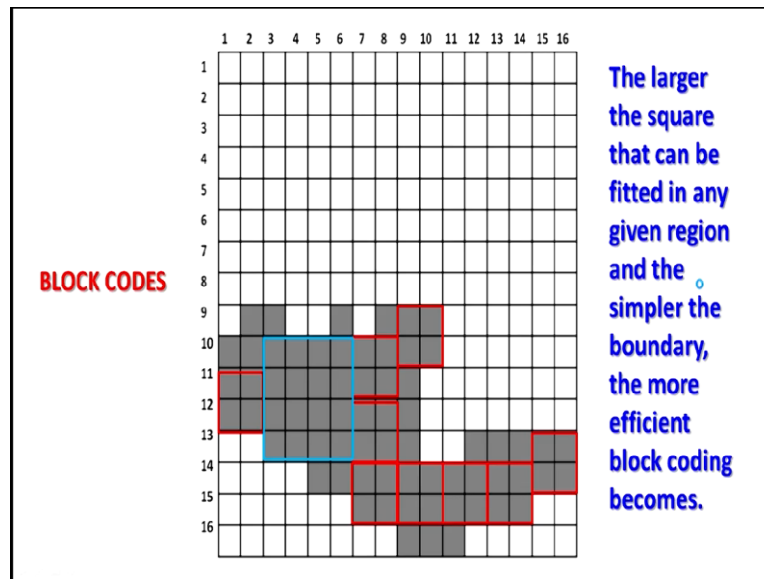
Then the data structure consists of just 3 numbers, that is the origin of the centre of the block and then radius of each square also is there.

(Refer Slide Time: 13:16)



So, here in there is only one square that means  $4 / 4$  pixels, there is only 1 square which is marked here in blue colour. And then plus we will have you know 9 squares of 4 pixels that means,  $2 / 2$  plus we will have 17 unit squares that means of the and we are reaching to the unit level and it be store the origin of each of these squares whether it is a  $4 / 4$  or  $2 / 2$  or 1 and the size of square. Then we can code the entire shaded part here. So, in that way we can do the block codes.

(Refer Slide Time: 14:05)



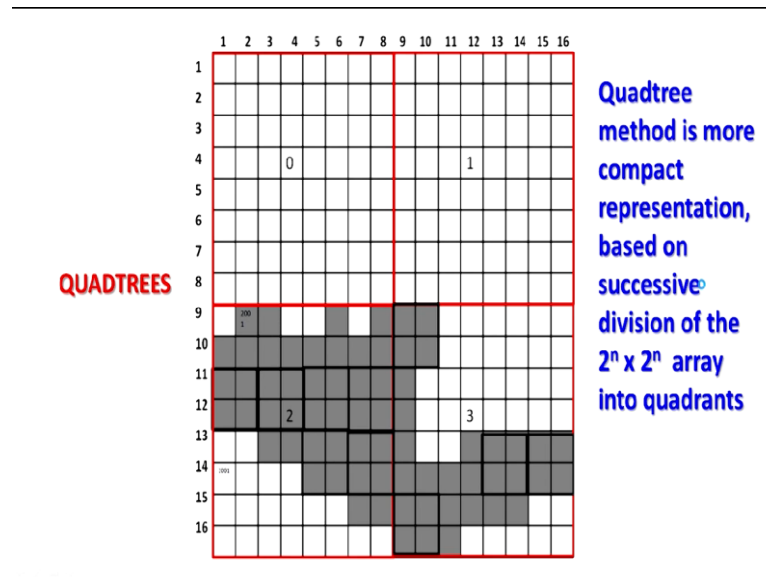
So, the larger the square and we get that means the homogeneity which is present. Now, we are seeing some more intricacies are coming in that and these data compression thing that if an image is having large areas, which are having the same pixel values that means, there is a large homogeneity is present you would achieve better compression even if you use any of the techniques which we are going so, 3 the third technique which we are discussing one more technique which we will discuss.

So, it will depend the efficiency of a compression will depend on the homogeneity of an image if a image is having heterogeneous means the pixel values are varying adjacent pixel values are varying having varying pixel values then you would not achieved high compression. So, for example, if one is having images of a desert area or a water body and there are not much movements or waves are there, then probably large areas will have the same pixel value and therefore, high compression can be achieved.

So, that is what here matters the homogeneity matters and heterogeneity will not provide high compression. The last technique which we are going to discuss is Quadrees is it is a more complex, but it provides a better compression and it has to this technique to has been implemented extensively in many software's for data compression.

**(Refer Slide Time: 15:50)**





So, in Quadtree what is done because the example which we have taken. The overall shape of the images is square. And as I said in the beginning the example which I have taken is the simplest one the overall shape of images is square and only it is a binary image that is black and white pixels are there. So let us see, first example of its simplest form and then we will see when an image is rectangular in shape, what will happen.

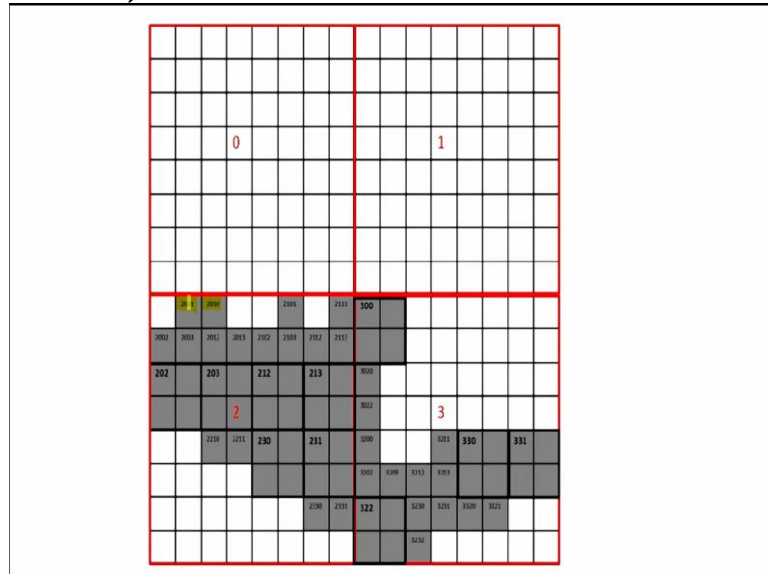
So, when image in a square in shape, first the entire image is divided in 4 quadrants and each quadrant and maybe assigned codes like 0 1 2 and 3. So for the northwest quadrant 0 for this entire area, 0 is assigned and for the other quadrants and 1 2 3 are assigned. Then searches made whether they it has achieved the homogeneity or not. So, in our example, quadrant 0 and 1 has achieved homogeneity that means no further divisions would we require for these 2 quadrants, whereas quadrant 2 and 3 would require further divisions.

So, let us now go for quadrant 2. So, that means further divisions. So, again 4 quadrants will be created like this, that quadrant 2 will be divided again in 4 sub quadrants, again the same way the search will be made. And if homogeneity has been achieved, then it will stop for the divisions if it has not, then it will go for further divisions, like in this case, if I take example of quadrant 2 0 because this is this will come 2 0 this will come 2 1 this will become 2 2 and this will become 2 3.

So, if I take example of 2 0, then again in homogeneity has not been achieved so far so 4 divisions will be done. Again the search is made. So now in quadrant 2 0 2 and 2 0 2 0 3 homogeneity has been achieved, but a quadrant 2 0 0 and 2 0 1 and the homogeneity has not

been achieved that means for the divisions will be required. So, divisions will be done till we reach to the unit level that is the pixel level. And likewise, the image is coded.

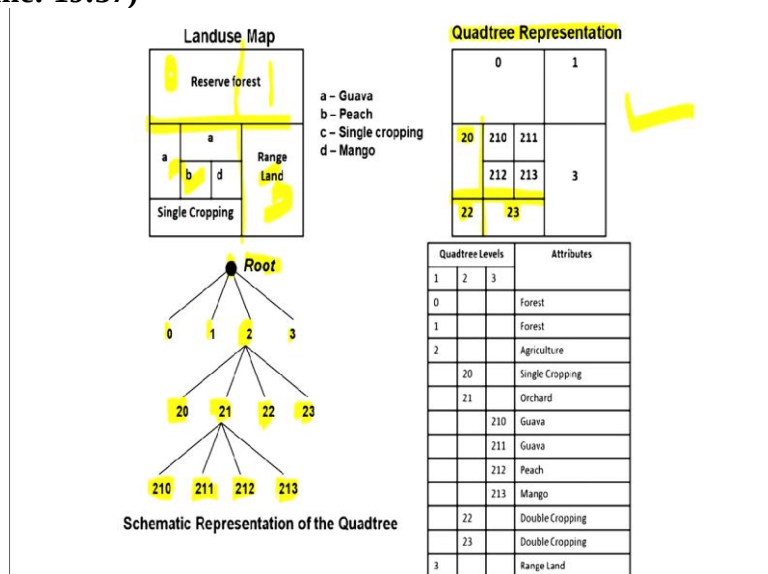
(Refer Slide Time: 18:39)



And so if we see the codes quadric quotes for this one and this will be something like this as you can see here, and that the example in this case only we have gone to the 4th level and each one has been assigned a code here because in quadrant 2 overall quadrant 2 then 2 you know 4 subdivisions were made the 0 so 2 0 then again subdivisions were made of quadrant were created. So again 0 and then finally be reached to 1 and so on, so forth.

So, in quadtree image is a square and shape, it is much easier 2 2 and 2 achieve compression through quadtree quite easy, but if image is not square in shape, if it is in rectangular and shape and then there might be a different scenario, so I am taking an example

(Refer Slide Time: 19:37)



And which is a and though it looks a square and say, but this is just a schematic. And so what happens if I divide in 4 quadrants, so my origin is here, then I do not have only one is square is left and that is in the same coding if I say 0 and 1 and 2 here and then 3 here then I am not having you know these quadrants as a square in shape, nonetheless extension on boundaries are done and before any compression is applied and then divisions will be made.

So, first a rectangular image would be made as a square in shape by adding few rows and columns on that one on the sides and then divisions could be made. So, likewise the quadtree representation of that Land use map which we are seeing on the left side will be done here. So, extension here on the drop side extension is also done on the right side then only quadrant means a square in shape would be achieved and also in the bottom side as well as you can see the origin here is this one.

So on all sides basically, maybe done, why it is called tree quadtree because this is the root. So, entire image starts from root 4 quadrants 0 1 2 3. And they suppose in this example, and the quadtree example, which we are seeing here that in quadrant 0 1 and 3 homogeneity has been achieved that means, no further divisions are required, whereas, for the and this region 2 quadrant 2 further divisions are required.

So, quadrant 2 further divisions are required 2 0 2 1 2 2 and 2 3 searches again made and it realized that a quadrant 2 1 of 2 basically 1 of 2 is required for the divisions because homogeneity has been not achieved, whereas in quadrant 2 0 2 2 2 3 homogeneity has been achieved therefore, no further division are required. So, 2 1 is required for the divisions there and buy this now at this stage we are reaching to the unit level.

So, that is why is it inverted tree or roots of it and tree which we are seeing here so, it is called quadtree and attribute information that means the pixel values can be stored like this which is shown in this table and though sounds a little complicated, but once it is coded, it provides a very high compression compared to other 3 techniques which we have discussed like chain codes, black codes, run lengthen codes, and so quadtree in that way is much better one can argue instead of quad can I have a octree yes.

One can definitely have but people have done that kind of competitive study. And by compressing the same image using quadtree and octree there is no significant improvement in

I mean that if I have compressed an image and whenever I want to decompress, I should go back or I should achieve the same image as I put for the compression and there is a compression or a file format also a compression technique which is called JPEG. JPEG is having problem it is began put in a destructive compression technique that means the image when it is saved as a JPEG then the image quality is compromised and there is a trade off that means you achieve very high compression.

**(Refer Slide Time: 24:12)**

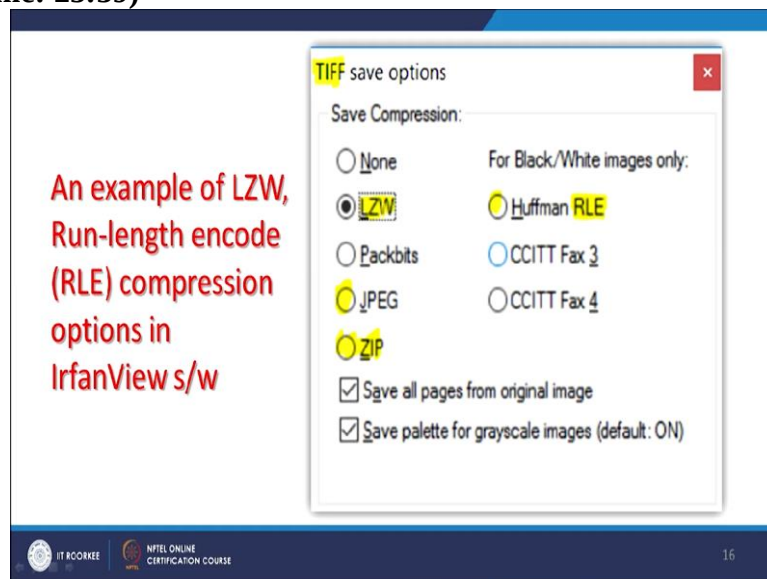
And LZW compression can be applied on TIFF images or other image formats, which is again common available in many software's, which was developed by 3 mathematicians Abraham Lempel, Jacob Ziv and Terry Welch and, you know, first character of their surnames have been given this name LZW, that is Lempel Ziv and Welch. So, this is called a LZW very popular compression technique.

And 2 commonly used file formats in LZW, which can be applied on GIF or TIFF image formats are used and LZW compression is suitable for compressing text files, because So, far

we have been discussing only image files, but text files can also be compressed we will not go for discuss on text file because the redundancy see the homogeneity is the major factor and to achieve better compression if because in text and redundancy part is very less and therefore.

In a type text and therefore, one cannot achieve very high compression, whereas, in case of satellite images sometimes and the redundancy part the homogeneity is much more and therefore, one can achieve high compression.

**(Refer Slide Time: 25:39)**

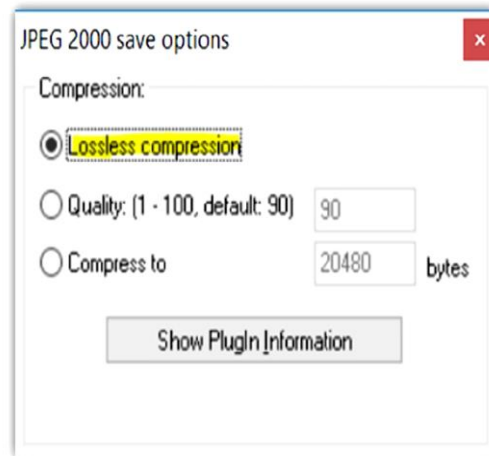


So, if you save a like I am taking an example, if you save a TIFF image, these are the compressions might be available. And the LZW compression is shown here. If you choose this compression, and then the different codes like RLE I was mentioning Run Length Encode. There are some other you know variant of these basic is a compression techniques are also available you can choose and JPEG is also given here.

And another one very popular which we use is ZIP which compress and basically tried to compress each and everything and or rather entire folder but, in sometimes only in the images, you can achieve better compression, but in case of just simple text files, we do not achieve much compression because redundancy is not there. So, RLE and LZW can go for TIFF files because we are trying to save it to TIFF file therefore, and these kind of options might be available depending on the software.

**(Refer Slide Time: 26:46)**

An example of  
compression  
options for  
JPEG-2000  
files in  
IrfanView s/w



If we go for JPEG as I have been mentioning that JPEG is a destructive compression technique. For example, if the original image was in the TIFF format I have compressed in JPEG format or saved as a JPEG format and I have deleted my original TIFF format TIFF file now by uncompressing what is saved is again TIFF and I will not achieve the same quality of image because JPEG is destructive.

But at the same time it provides a very high compression. So, though in options there is a like lossless compression. So, it is a compromise it will do it will provide some compression but not a very high compression, but at the same time there will not be any loss that means non-destructive and JPEG is also possible, quality you can control that how much and you want to compromise on the quality if you keep 100 % that means you are going for a lossless compression but if you give 50 % you are compromising on the quality.

Also in JPEG, in some software's and this is very interesting options are also available, sometimes we have to upload an image of it is restricted said that image has to be only one megabyte. So, we can also save as that image in JPEG and declaring here that I want only 1 megabyte or 2 megabyte. So, I can go for this option and I can constraint here that the image compression and accordingly the quality will be chosen of a JPEG. So JPEG is a, JPEG provides very high compression, but at the same time it compromises the quality.

So, if you are saving any image from TIFF to JPEG, please do not delete your TIFF image because you cannot normally in normal circumstances, you cannot go back to the original quality which was available through TIFF. Now, these LZW compression techniques and

which takes for example 12 bits of data and creates their own tables, which are called dictionary Or code books and then the compress the data and similarly.

(Refer Slide Time: 29:14)

- LZW compression algorithm takes each input sequence of bits of a given length (for example, 12 bits) and creates an entry in a table (sometimes called a "dictionary" or "codebook") for that particular bit pattern, consisting of the pattern itself and a shorter code.
- As input is read, any pattern that has been read before results in the substitution of the shorter code, effectively compressing the total amount of input to something smaller.

(Refer Slide Time: 29:21)

- Unlike earlier approaches, known as LZ77 and LZ78, the LZW algorithm does include the look-up table of codes as part of the compressed file.
- The decoding program that uncompresses the file is able to build the table itself by using the algorithm as it processes the encoded input.

There are different versions like in JPEG there are different versions of compression techniques latest is 2000 in JPEG. Similarly, in LZW some software's you may find and these options with these versions like this LZW of 77 78 or later LZW. So, they basically they look the compression is achieved through look up tables and having different approaches in different versions and these decoding programs or that is uncompressing programs will be accordingly should also be chosen.

(Refer Slide Time: 30:00)



JPEG (Joint Photographic Experts Group) is an ISO group of experts that develops and maintains standards for a suite of compression algorithms for computer image files.

- JPEG (usually pronounced JAY-pehg) is also a term for any graphic image file produced by using a JPEG standard.
- A JPEG file is created by choosing from a range of compression qualities (actually, from one of a suite of compression algorithms).
- When one creates JPEG or convert an image from another format to a JPEG, question is asked to specify the quality of image you want.

Now, just say JPEG is a basically ISO group and which is stands for joint photographic experts group and this a JPEG is really usually as we know that is pronounced as JPEG. JPEG is choosing the range of compression is there and when JPEG is there, we convert an image from another format and the question is asked to specify the quality you want. So, compromises there,

**(Refer Slide Time: 30:29)**

- Since the highest quality results in the largest file, you can make a trade-off between image quality and file size. Formally, the JPEG file format is specified in ISO standard 10918. The JPEG scheme includes 29 distinct coding processes although a JPEG implementer may not use them all.
- Together with the Graphic Interchange Format (GIF) and Portable Network Graphics (PNG) file formats, the JPEG is one of the image file formats supported on the World Wide Web, usually with the file suffix of ".jpg". You can create a progressive JPEG that is similar to an interlaced GIF.

One has to be very careful. So, this we have already discussed there is another format which is also equally popular which is called the PNG format apart from your GIF format, which are also available and which are also used on internet is basically the GIF format or PNG format. And these are non destructive, but obviously, GIF saves many majors in one single file having a different thing

**(Refer Slide Time: 31:02)**

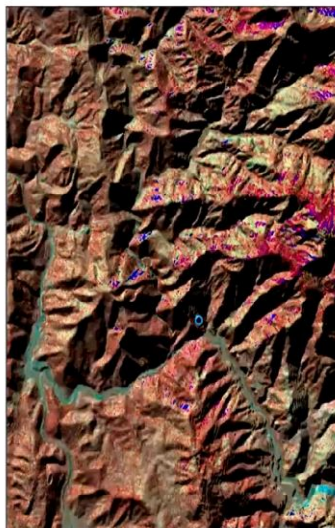


- The GIF (Graphics Interchange Format) is one of the two most common file formats for graphic images on the Web. The other is the JPEG.
- The GIF has become a de facto standard form of image.
- The GIF uses the 2D raster data type and is encoded in binary.
- There are two versions of the format, 87a and GIF89a.
- Version 89a (July, 1989) allows for the possibility of an animated GIF, which is a short sequence of images within a single GIF file.

IT ACORRE NTTEL ONLINE CERTIFICATION COURSE

For example, I can show you and this GIF, GIF is also popular for 2D dataset. So, the example let me show you.

**(Refer Slide Time: 31:12)**



And this animation also I have shown in some other presentations. So, this is in the GIF format, there are 8 scenes and which have been put in one single file a GIF file has been created and these are the this format is very popular for animations. One more example of GIF is given here, we are you know, many scenes have been more than 10 scenes have been put in a time sequence and each frame time gap has also been set and accordingly one can create a GIF.

So, many say 10 scenes can be put in one single file and therefore, one can see that this is also a type of compression technique. But using this format and one can create beautiful

animations, which are very popular on internet. So, GIF there are also versions of GIF are available and which is a graphic interchange format or interlaced for presentations.

(Refer Slide Time: 32:21)

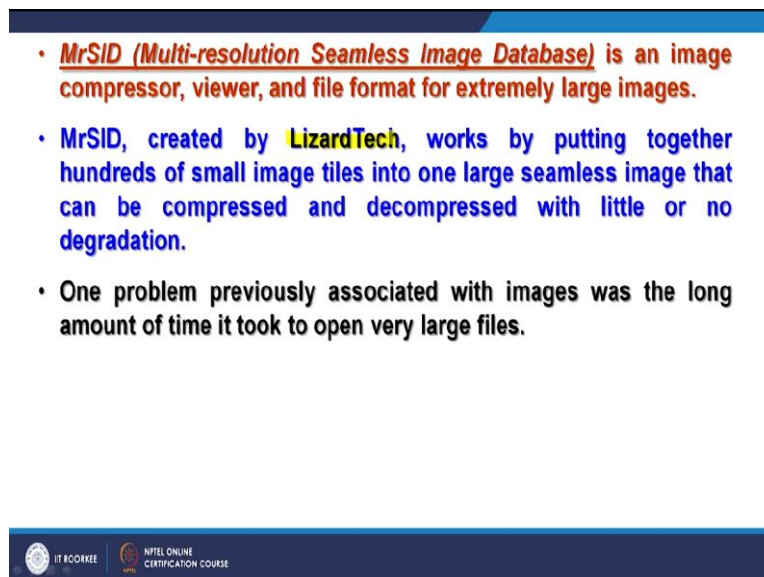


- A GIF89a can also be specified for interlaced GIF presentation.
- A patent-free replacement for the GIF, the PNG format, has been developed by an Internet committee and major browsers support it or soon will.

IT 800KKEE NPTL ONLINE CERTIFICATION COURSE

Also GIF with along with the PNG format has been developed especially for internet and these are very popular on there.

(Refer Slide Time: 32:29)



- MrSID (Multi-resolution Seamless Image Database) is an image compressor, viewer, and file format for extremely large images.
- MrSID, created by LizardTech, works by putting together hundreds of small image tiles into one large seamless image that can be compressed and decompressed with little or no degradation.
- One problem previously associated with images was the long amount of time it took to open very large files.

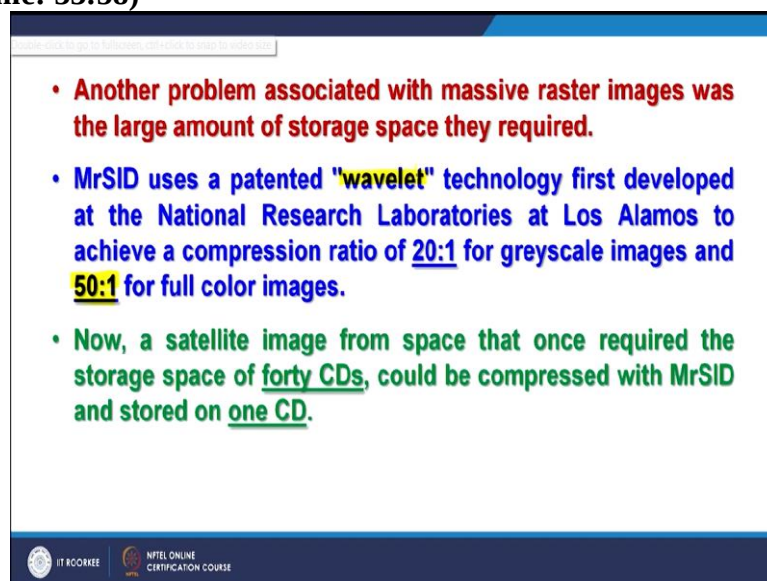
IT 800KKEE NPTL ONLINE CERTIFICATION COURSE

Now, 2 more points which I want to discuss. And before I close this discussion, one is MrSID a multi resolution seamless image dataset or database. Basically this is also a compression technique. And this is a this technique has been developed by a company which is called Lizard Tech and the details about and the intricacies or inside of this compressions are copyright protected.

And therefore, only the advantages or the benefits which we get and can I will be discussing and because this is the technique which provides the highest compression so far and this is basically non destructive image compression technique MrSID and the images high resolution images which you see on say Google Earth or other platforms, they in the background MrSID data compression tools or techniques are being used there.

It provides a very high compression even it can provide compression up to 50 times that means if I am having an image of 50 MB using MrSID compression tool, I can reduce to 1 MB and that too without destruction.

(Refer Slide Time: 33:58)



Multiple choice to go on following, click on to pay to add more

- Another problem associated with massive raster images was the large amount of storage space they required.
- MrSID uses a patented "wavelet" technology first developed at the National Research Laboratories at Los Alamos to achieve a compression ratio of 20:1 for greyscale images and 50:1 for full color images.
- Now, a satellite image from space that once required the storage space of forty CDs, could be compressed with MrSID and stored on one CD.

IIT Bombay NPTEL ONLINE CERTIFICATION COURSE

So, 50 times full colour image can be reduced by 50 times. So this ratio of compression is very, very high. And this technique uses the wavelet concept of wavelet transform and it has been developed by National Research Laboratories but it is copyright protected and they are like 40 CDs data can be stored in just one CD if the compress using MrSID software like ERDAS supports any file of 500 MB or less can be saved as MrSID.

But if I am image file is more than 500 MB then I have to pay for to that company to save or compress my image file. So free is available up to only 500 MB. The last in this discussion is about the profits. This is a different concept. Though be one can argue that this is not really a compression technique about nonetheless, it provides a faster display and promise means that the base will have a high spatial resolution images.

And (35:16) say you would have a coarser resolution representation that means, when I am displaying a large area, I would be seen at a relatively coarser resolution, when I go for zooming I continuously and going in you know inside of that Pyramid going towards the bottom going towards the higher spatial resolution and why which I can improve the display performance of my system.

So, Pyramid is a very popular it has been implemented in like popular software like ArcGIS or ERDAS and first time when you display try to display an image of a larger size, the system will ask, do you want to create Pyramid So, if you go for the yes option and then only 2 small files are creating a new system having the first before and before point having the same need only the extensions would be different and Pyramid will be created in few second time.

Next time when you will display the image which for which a Pyramid have already been created, then the display is going to be very fast. So, it is basically is can speed up the display of raster images by retrieving

(Refer Slide Time: 36:33)

## Pyramids

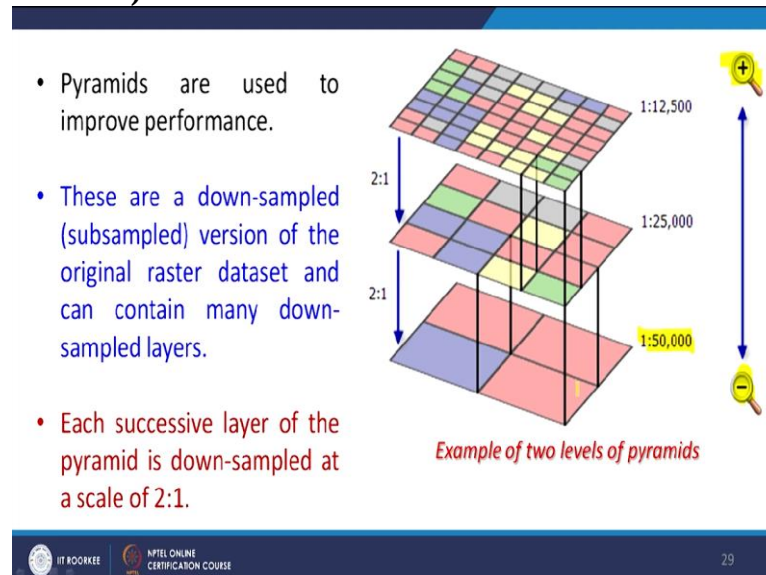
- Pyramids are reduced resolution representations of dataset and are used to improve performance
- Pyramids can speed up display of raster data by retrieving only the data at a specified resolution that is required for the display



Data only at a specified resolution and that is required for the display. So larger the area coarser the resolution will display. But when we go for the zooming, that means I am asking to display a smaller area, but then higher spatial resolution. So Pyramid works in that way. At the courses label. The resolution is a quick quickly drawn because it using coarse resolution that means say Pyramid at the top of the Pyramid.

When we I am zooming, then I am going towards the final space of resolution or towards the bottom of the Pyramid. And that way, and the my display becomes this figure on the right side depicts that thing, that

**(Refer Slide Time: 37:20)**



This scale order that the resolution and you know, just 4 cells for pixels are representing the area. But when I zoom in, like here it is so in zooming, then I am going for finer and finer spatial resolution when I zoom out and then I go for coarser and coarser spatial resolution. And that means it improves the display performance of our system by just creating Pyramid only once the first time when you display and any file on your system.

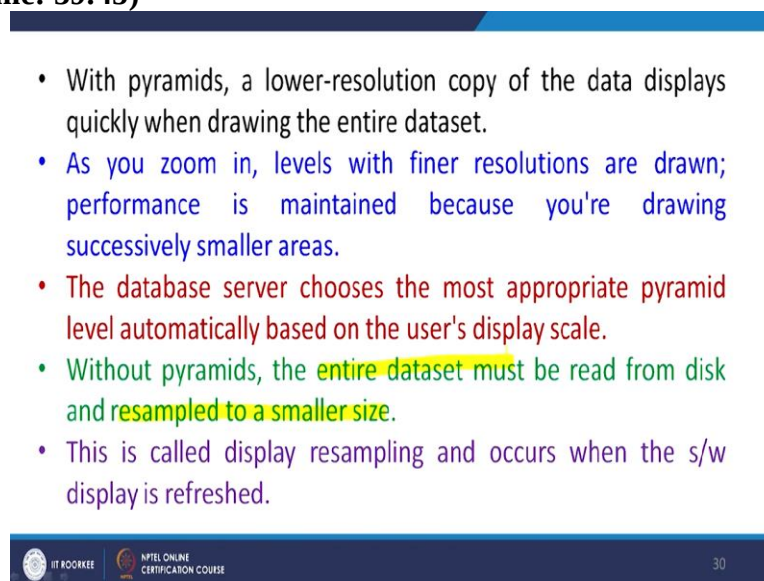
And then it will ask, do you want to create Pyramid and once you have said yes, then next time it will not ask. And your display will be faster. So, in that way and one can say that it provides a compression kind of thing, but no harm to original image, see the we are talking of higher spatial resolution that means the quality we are talking and if we have to compromise the quality then no use of going for higher spatial resolution.

For example, nowadays anyone with mobile cameras we are going for higher and higher megapixels camera, but the same time people are storing their images from high megapixel camera in JPEG format and therefore, and when they acquire the image originally they have destroyed the quality of image. So, better it is if you are using high spatial resolution cameras, it is better first to store image in TIFF and later on.

If it is required you can convert to or save as a JPEG and because that is why we are going for higher spatial resolution and camera or higher spatial resolution satellite images. So satellite images spatially. Should not initially stored in JPEG format because JPEG is a destructive kind of image compression, the original file if it has been deleted, then the same quality by decompressing JPEG or save as JPEG as TIFF, you will not archive.

So, this one has to keep in mind many techniques which I have discussed are all non destructive except the JPEG. So, Pyramids are basically is that I was saying and provides a fine resolution at the zooming and it maintains a very good

**(Refer Slide Time: 39:45)**



- With pyramids, a lower-resolution copy of the data displays quickly when drawing the entire dataset.
- As you zoom in, levels with finer resolutions are drawn; performance is maintained because you're drawing successively smaller areas.
- The database server chooses the most appropriate pyramid level automatically based on the user's display scale.
- Without pyramids, the entire dataset must be read from disk and resampled to a smaller size.
- This is called display resampling and occurs when the s/w display is refreshed.

Efficiency in that display and that basically makes our you know system more efficient and basically it resemble any smaller size the entire dataset at coarser size in the Pyramid and that is the it helps the display very quickly. So, this Pyramids details are here

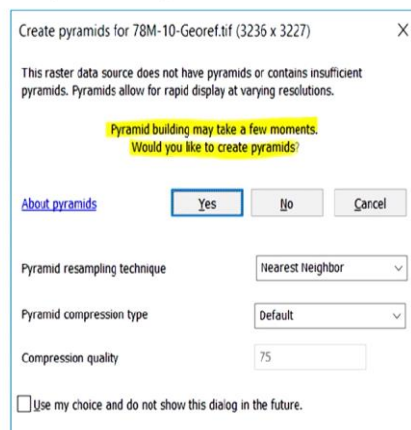
**(Refer Slide Time: 40:12)**



- Pyramids only need to be built once per raster dataset; after that, they are accessed each time the raster dataset is viewed.
- The larger the raster dataset, the longer it takes to create the set of pyramids.
- However, this also means you will save more time in the long run.

(Refer Slide Time: 40:16)

### An example of Pyramid structure in ArcGIS



And larger the image it is always go better to go for and Pyramids example here, when you first time displayed this example is from ArcGIS when you go for display it will ask their Pyramid make a few moments, but would you like to create Pyramid, because this raster data source does not have permits. So, first is checks whether Pyramid have already been created or not.

If is yes then it will not ask but if it no it will ask. So, once the Pyramid has been created, then of course there are some options will also be there so better. If we want to retain the quality of image then always choose nearest neighbour, which is generally in the default and compression and other things you can change and once it has been created, then you get The advantage of faster display on your system.

So, this brings to the end of our discussion we started with the 4 basic data compression techniques. Then we also discuss the LZW which is associated with different zip. We also discuss a MrSID, which provides very high compression, completely non destructive data compression technique, but which is a copyright protected. So, if you are having large datasets, then you have to pay and finally, we have discussed about the Pyramid.

It helps us to display large images of higher spatial resolution at a very high speed. But if I have to cover if I have to display the entire image of high spatial resolution, then Pyramid will help me to display very quickly. So, this brings to the end of this discussion. Thank you very much.