

## 8. EDITING AND VIEWING COORDINATES, CREATING SCATTERGRAMS AND PRINCIPAL COMPONENTS ANALYSIS

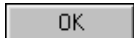
**Aim:** To introduce you to (i) how you can apply a geographical coordinate system to a rectified image of known geographical origin and pixel size (but stored as rows and columns only), (ii) how to plot pixel values in one band against those in another band to visualise how features separate spectrally, and (iii) how principal components analysis (PCA) can aid image analysis.

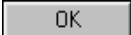
**Objectives:** By the end of this tutorial you will have learnt how to apply geographical coordinates to a (previously rectified) image imported without coordinates from another software package. You will also have explored the scattergram option, which allows pixel values in two bands to be displayed as a chart, and carried out PCA to enhance separation of different land cover types.

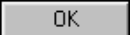
In the previous section on [geometric correction](#), methods for rectifying and resampling raw images were studied. Quite often you need to exchange images between different software packages or you may be sent binary “flat file” images without any coordinates attached, but which have already been geometrically corrected to a geographical coordinate system such as Universal Transverse Mercator (UTM) or latitude and longitude. In such cases, at least the coordinates of the top left of the image (the image “origin”) and pixel dimensions will normally be known. Armed with this information you can apply coordinates to the image and then save it as a CompuServe .gif or *Bilko* .dat file which will store the coordinate information. You will practice this using one band of a CASI (Compact Airborne Spectrographic Imager) digital image, which has been previously georeferenced to the local UTM grid system.

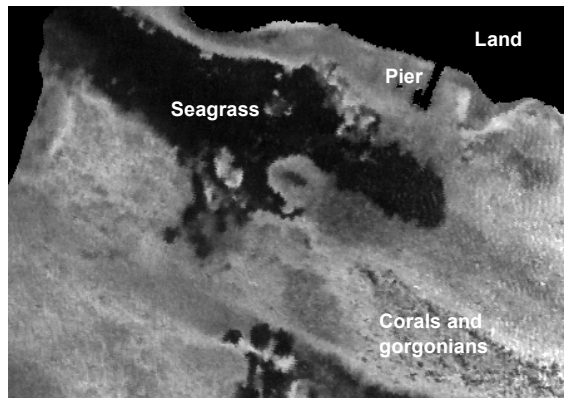
The single band image has been exported from ERDAS Imagine (a commercial remote sensing package) as a 16-bit unsigned integer binary flat file (.bin) extension (**Casi\_16-bit\_unsigned#2.bin**). Each pixel in this image is 1 × 1.1 m in size (so the spatial resolution is 1000× better than the AVHRR images you’ve been studying). The image is 452 pixels wide and made up of 317 scan lines.

*Activity:* Use **File, Open** to start the process of opening **Casi\_16-bit\_unsigned#2.bin**. *Bilko* needs more information from you to open this file and so an **Open As flat file** dialog box appears. There is no header, so leave the **Header length (in bytes):** entry as **0**. Put the correct values (see above) in the **Pixels per Line:** and **Number of Lines:** boxes. Now move to the **Pixel Format** drop-down menu box and select the correct pixel format. Since there is only one band the **Interleave Format** has to be Band Sequential (BSQ), which is the default. The program now has enough information to open the file, so click on



If the **Extract** dialog box appears, click on . In the **Redisplay Image**

dialog box, check the **Null Values(s):** checkbox so that 0, which is used to denote background areas outside the scanned image and masked land, is set as a null value. Note that the image data values range from a minimum pixel value of 626 to a maximum of 3988. Click on  to display the image with the default Linear stretch. You will now see the image which shows a seagrass bed off South Caicos Island in

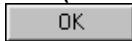


the West Indies. See above for a labelled picture to help you orientate yourself.

*Activity:* Click on dark and bright parts of the image such as the large seagrass bed and the sandy area around the coral and gorgonian (sea-fan) dominated seabed. Note the underlying pixel values displayed in the second to last panel of the Status Bar and the display values in the last panel. The underlying values were recorded on a scale of 0–4095. Click on the **View** menu and note that the **Coords** option is unavailable.

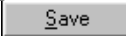
The image just has row and column coordinates at present but you are provided with the information that a GPS (global positioning system) position fix for the top left hand corner of the image gave the Easting 237430 and Northing 2378282 for UTM grid zone 19. You have also been told the pixel size after resampling ( $1.0 \times 1.1$  m).

*Activity:* Whilst the **Casi\_16-bit\_unsigned#2.bin** image is the current window, select **Edit, Coords** to enter the UTM coordinates. This will bring up the **Set Coordinates** dialog box. Enter the appropriate values (see above) in the **Easting (X):** and **Northing (Y):** boxes. The pixel size is 1.0 m wide by 1.1 m along-track. Enter these values in the **Pixel Size (metres)** boxes for **Width (DX):** and **Length (DY):** respectively and then click on



Now that you have entered the UTM coordinates of the top left of the image (which has been previously geometrically corrected) the *Bilko* program can calculate the UTM coordinates of any pixel. If you look at the Status Bar you will now find that rather than columns and rows, UTM eastings and northings are now displayed. If you click on **View, Coords** you can switch back to row and column coordinates. With **View, Coords** checked use **Edit, Go To...** function to put the cursor at UTM position 237760, 2378193.

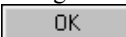
*Question 1:* Where on the image is the UTM coordinate 237760, 2378193?

*Activity:* To save the image with its coordinate information, you can save it in *Bilko* **.dat** format. Select **File, Save As** and in the **Save As** dialog box, select the file type as *Bilko* **.dat** (if not already automatically selected) and click on . Close the file and then re-open it. Note that now *Bilko* knows the file size and type and automatically displays the UTM coordinates. Close the file, which will not be needed again.

*Note:* All types of file can be stored in *Bilko* **.dat** format but standard CompuServe **.gif** format is recommended for 8-bit files. For export to other applications, **.bmp** (Windows bitmap) or **.bin** (flat binary) file formats are most likely to be useful.

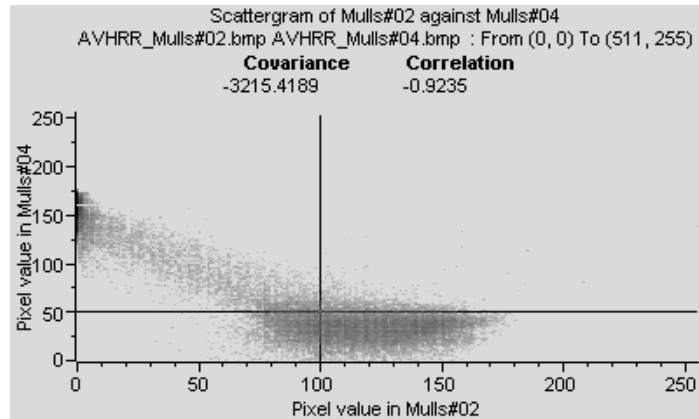
## Scattergram option

When you have two connected images, such as images of the same scene recorded in two different wavebands, it is often useful to be able to view how reflectance or DN values of pixels in one band relate to those of the same pixels in the other band. **SCATTER documents** allow you to do this. You will connect a band#2 and band #4 AVHRR image and see how pixel DNs are correlated between these two bands. Details of these two images are provided in earlier tutorials.

*Activity:* Open the two AVHRR images **AVHRR\_Mulls#02.bmp** and **AVHRR\_Mulls#04.bmp**. Use **Image, Connect** to connect the two images and use the **Selector** toolbar to make sure that **AVHRR\_Mulls#02.bmp** is designated image 1 (which will be plotted on x-axis) and **AVHRR\_Mulls#04.bmp** is designated image 2 (which will be plotted on y-axis). With the connected tiled images as the active window, click on **Edit, Select All** to select all of the two images. Then select **File, New** from the menu and **SCATTER Document** and click on  to display a scatter-gram. This is a plot of the DN values of each pixel in

**AVHRR\_Mulls#02**

(image 1, x-axis) against its DN value in **AVHRR\_Mulls#04** (image 2, y-axis). Note that there is an inverse relationship between pixel DN values in the two images. Thus pixels with a high DN in **AVHRR\_Mulls#04** have a low DN in **AVHRR\_Mulls#02**, whereas those with a



low DN in **AVHRR\_Mulls#04** tend to have a high DN in **AVHRR\_Mulls#02**. Note the cross-hair on the graph (positioned at a pixel value of 100 in Mulls#02 and 50 in Mulls#04 in the example above). The number of pixels with the combination of band #2 and band #4 values indicated by the cross-hair is shown in the third panel from the left of the Status Bar. [The stretched “display” value of this number is also shown in the fourth panel and can be useful when interpreting pixel densities]. Use the mouse to drag the cross-hair to coordinate 100, 50 on the scattergram.

*Question 2:* How many pixels have value of 100 in band #2 and 50 in band #4 of the **AVHRR\_Mulls** image?

*Question 3:* What feature on the image has pixels with a high DN in **AVHRR\_Mulls#04** but low DN in **AVHRR\_Mulls#02**? What feature on the image has pixels with a low DN in **AVHRR\_Mulls#04** but high DN in **AVHRR\_Mulls#02**? Why is there this inverse relationship?

*Activity:* Note that if you right-click on the scattergram, you have the option to **Redisplay** it with different stretches (just as for an image). You can also copy and paste palettes to it so that the density of pixels with different pairs of values is clearer. Open the palette **SST\_Pathfinder.pal** and copy and paste it to the scattergram. Note that with the default logarithmic stretch, pixel densities around 90–120 on the display scale are light-blue to cyan and those around 25–50 on the display scale are shades of mauve (assuming you have not changed the default logarithmic stretch). Close the scattergram, the connected tiled images, the palette and the **AVHRR\_Mulls#02.bmp** and **AVHRR\_Mulls#04.bmp** images.

**Principal Components Analysis**

Adjacent bands in multispectral images are often correlated, which implies redundancy in the data as some information is being repeated in different bands. Principal components analysis (PCA) defines the number of dimensions that are present in a data set and the principal axes of variability, and generates principal component images that encompass this variability. Thus in a six band Landsat Thematic Mapper (TM) image (omitting thermal band 6) of land cover you may be able to encompass over 95% of the variability of the data in the first 3 principal component (PC) images. A colour composite image made with these three PC images is thus likely to give you a much clearer picture of different land cover types than any combination of three of the original bands. A useful account of PCA is given in Mather (1999), which refers to the images you will use.

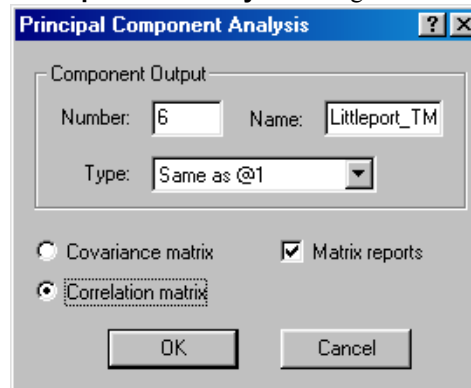
To demonstrate this and how to carry out PCA in *Bilko*, you will take a set of six bands from a Landsat TM image of the countryside around Littleport, near Ely in Cambridgeshire, UK. The main features in the image are the River Ouse running from south to north on the right-hand side of the image and the

parallel Old and New Bedford Rivers, running in a north-easterly direction on the left-hand side. The area is low-lying, flat, fertile Fenland where barley, wheat and sugar-beet are grown. A few clouds and their shadows can be seen.

*Activity:* Open the set **Littleport\_TM.set**, which will open the six bands as a stack. Use the <Tab> key or **Image, Animate** option (or button) to scroll through the six bands. Some images are a little poor in contrast, so use <Ctrl>+A to select all of the images and apply an automatic linear stretch. Scroll through the bands again and note the improvement. Now make a colour composite (you just need to select **Image, Composite** as the images are already connected). The false colour composite formed from bands #1, #2 and #3 is poor in contrast so apply an automatic linear contrast stretch to this too. This improves it markedly.

Your next task is to perform PCA on the six bands and then construct a false colour composite using the first 3 principal component images. This can be compared to the raw band composite already made or other composites you may care to make using different band combinations.

*Activity:* Minimize the composite and make sure the stacked set of images is the active window. Select **Image, PCA** to open the **Principal Components Analysis** dialog box. Leave the number of components set to the default of 6 and the type of output set to Same as @1 but select **Correlation matrix** for the matrix type. You will usually want the **Matrix reports**, which provide important information about the PCA, so leave this box checked. When the dialog box is set up correctly (see right), click on OK. Six principal component images will be produced, each accounting for progressively less of the variability in the data. Also two tables will be produced. The first of these shows the correlation matrix and has the first column headed **Correlation**. It shows the degree of correlation between all combinations of bands. The second table shows the principal component loadings for the six principal components of the Littleport TM image and has its first column headed **PCA**.



The principal component images will be labelled something like **Littleport\_TM pc1**, **Littleport\_TM pc2**, etc. You now want to make a false colour composite of the first 3 principal components.

*Activity:* Close the unwanted principal component images; that is, the pc4, pc5 and pc6 images. Then connect the pc1, pc2 and pc3 images as a stack. You can now directly make a colour composite (you just need to select **Image, Composite**). The false colour composite formed from principal component images pc1, pc2 and pc3 shows different land cover types clearly but its contrast can be improved. Apply an automatic linear contrast stretch. I hope that you agree that land cover types are now extremely clear! Compare this composite with the original one made from the three visible waveband TM images (and, if you wish, other combinations of raw bands).

This shows how PCA can be a useful method of data compression, concentrating most of the information in the six bands into 3 principal component bands in this example. To find out how much of the variability (variance) in the data has been compressed into the 3 principal component bands, you need to examine the PC loadings table and perform a few calculations on it. This can be done most easily in an *Excel* (or other) spreadsheet and forms part of a [mini-lesson \(Mini-lesson04\\_Principal\\_Components\\_Analysis\)](#), which you may wish to look at if you have reasonable spreadsheet skills. This analysis indicates that approximately 97% of the variance in the data is encompassed by the first 3 principal component images.

*Activity:* When you have finished, close all images and tables.

In earlier sections you have seen how to enhance images using stretches and palettes; the next section introduces you to how to use [\*\*filters to enhance features of interest in images\*\*](#).