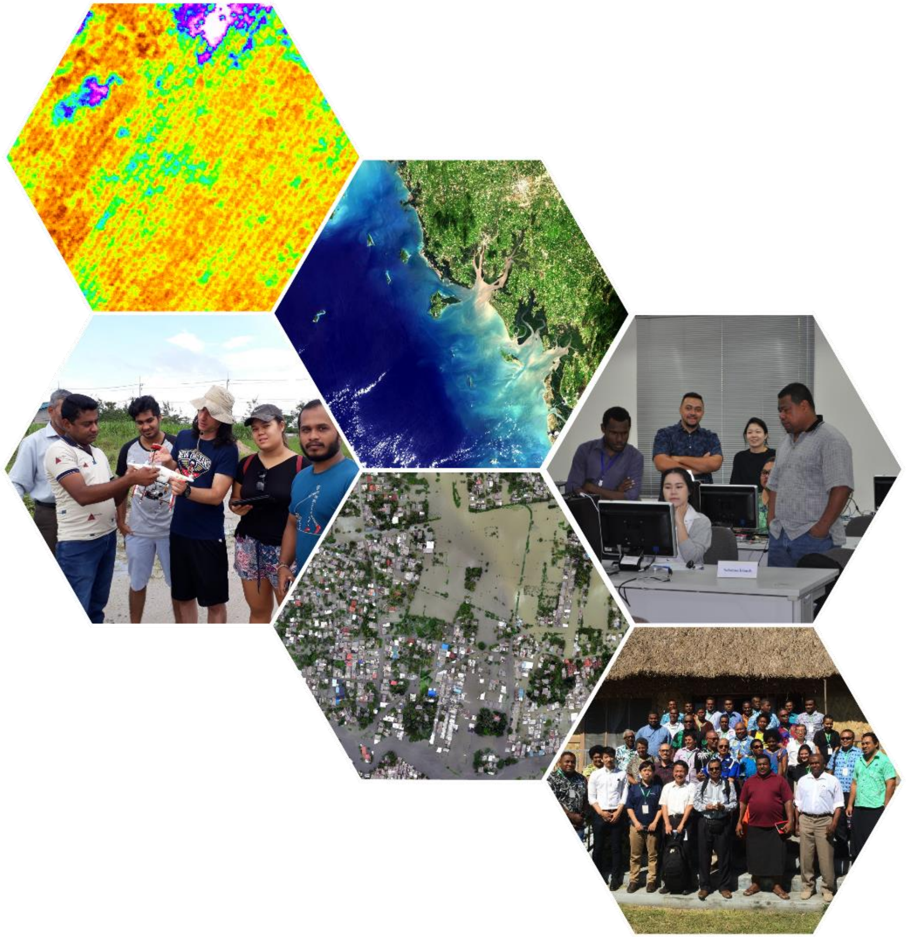
Forest Biomass Estimation

Practical Session



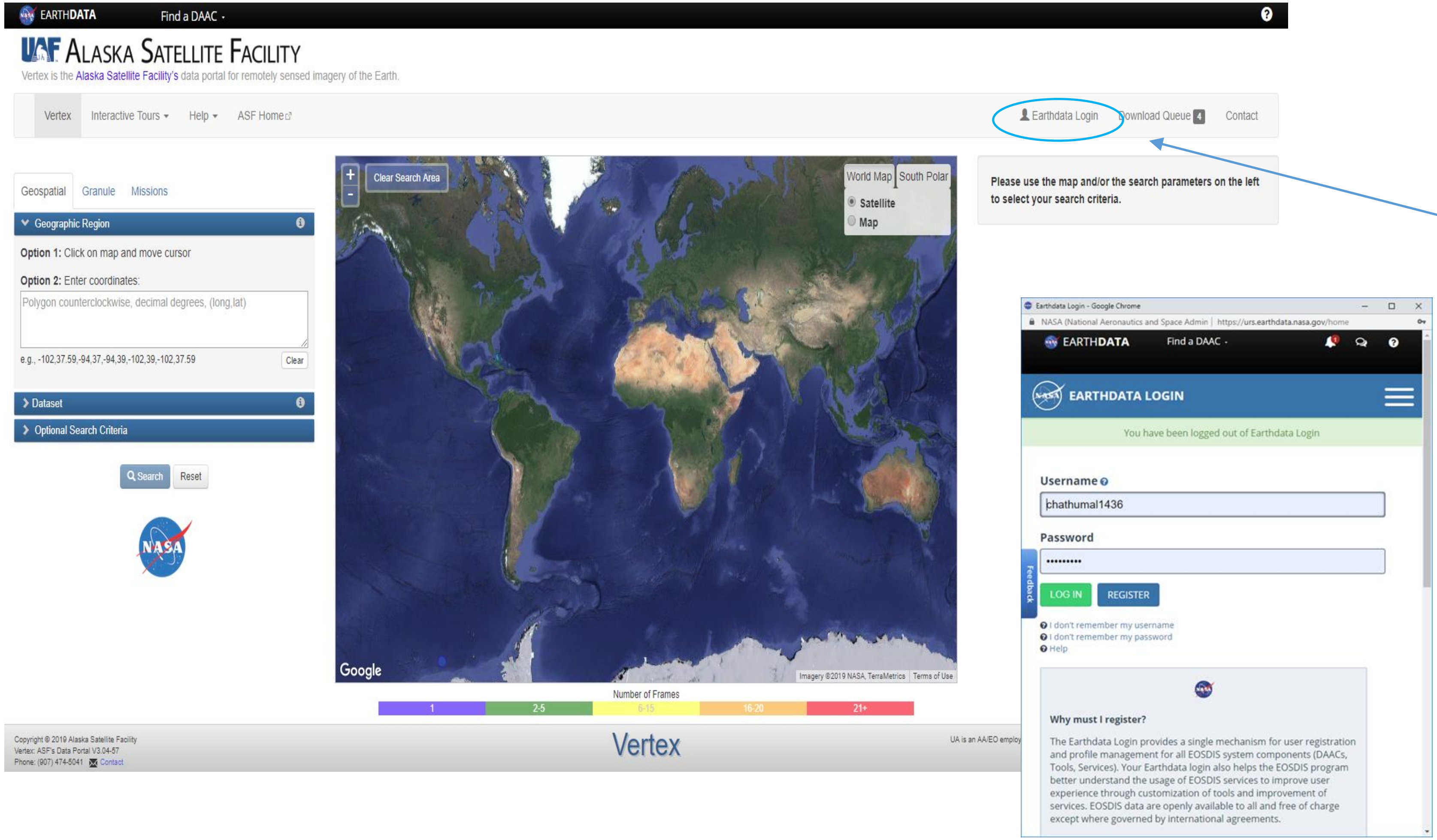
Chathumal Madhuranga

AIT Asian Institute of Technology

GIC

Downloading ALOS PALSAR Data

Alaska Satellite Facility vertex

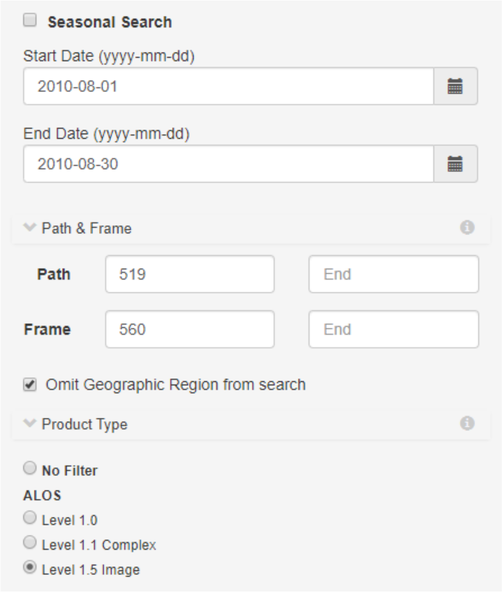


- ALOS PALSAR data can be downloaded from this site for 2006 – 2011
- First click on the earth data login and register.
- If you have already registered, log in and look for the ALOS PALSAR data.

- Provide the relevant information to the vertex and find the ALOS PALSAR image over Nepal.

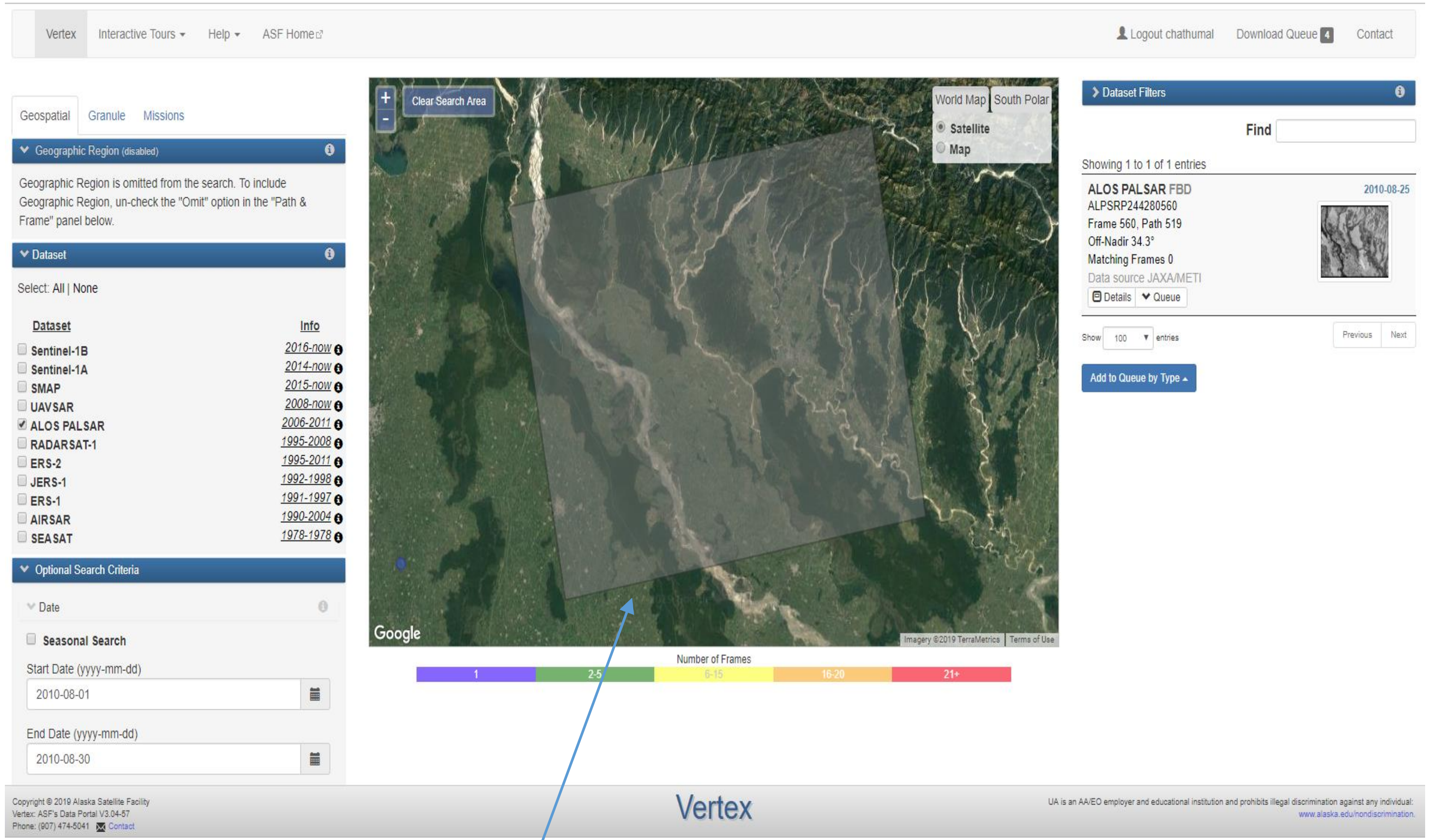
Specify the time frame

If you know the path and frame you can insert here



Specify the processing level of the images

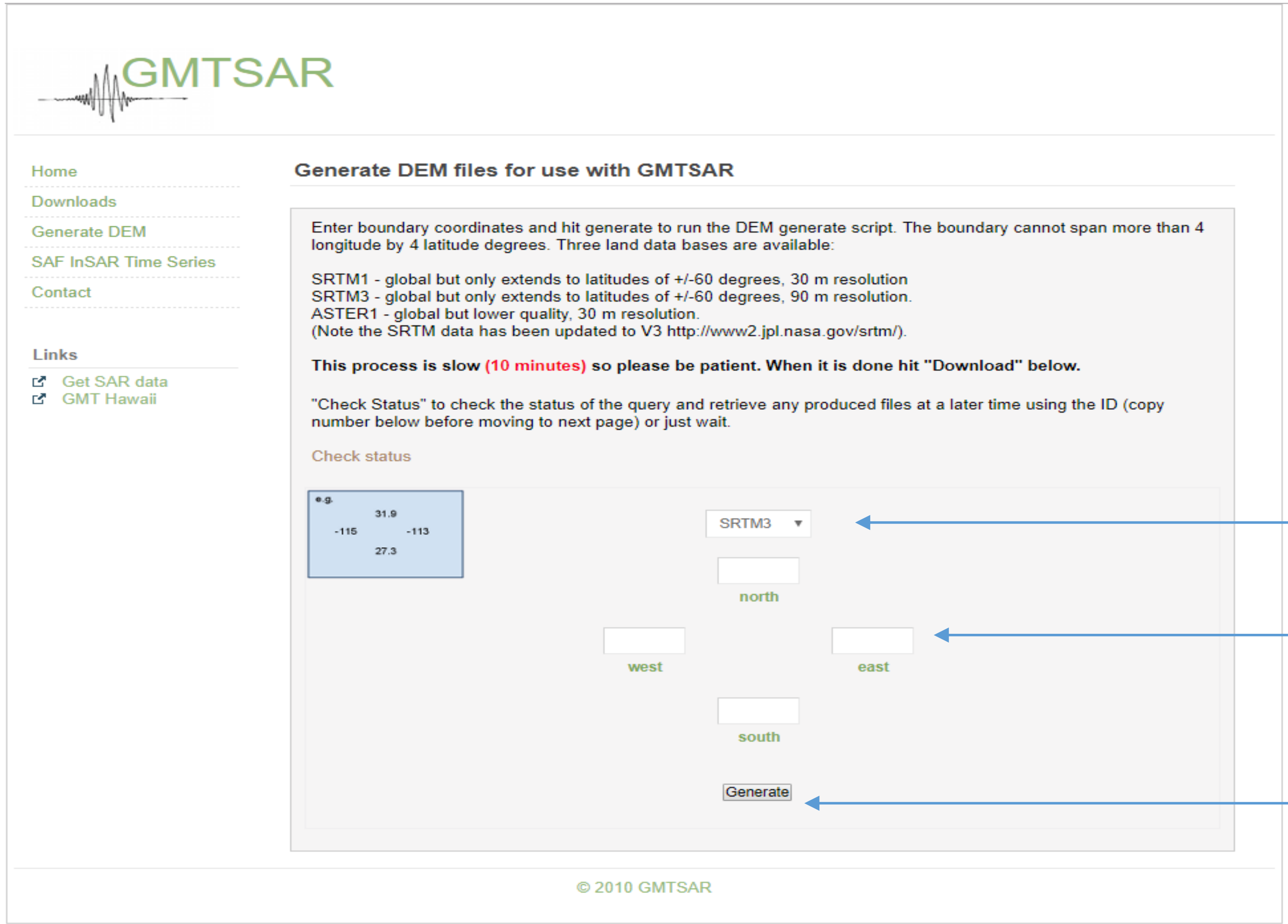
Tick the dataset you need



You can select the geographical region from here

Downloading SRTM DEM

- You can use GMTSAR website for download DEM



Type "Gmtsar dem" in web browser and go to GMTSAR Generate DEM site.

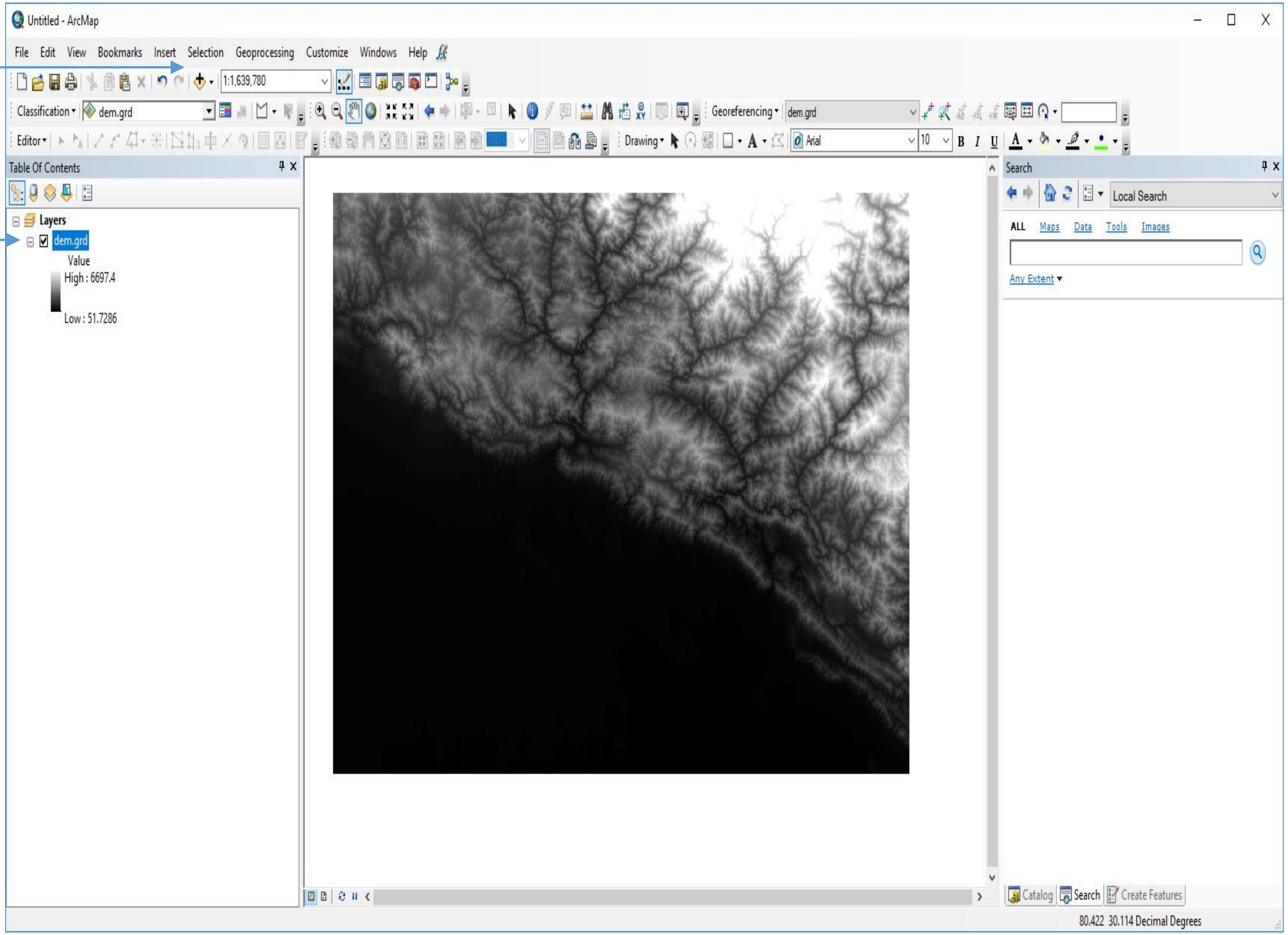
Select the dem resolution you need(SRTM3)

Specify the latitude and longitude decimal degree coordinates of the Area of interest

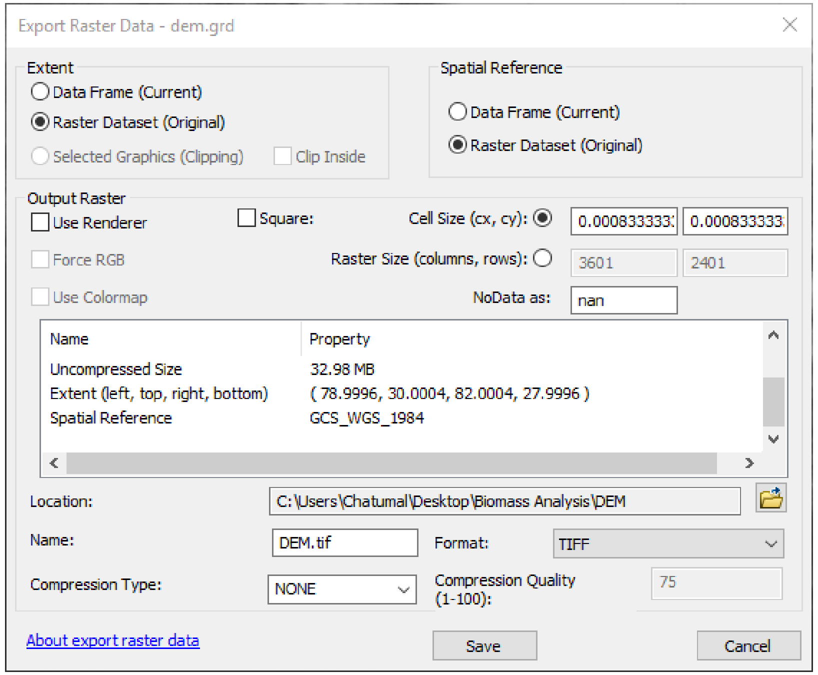
Click on generate. This will take little time then you can download the DEM

Convert DEM.grd to DEM.tif

Open the downloaded dem file(dem.grd) in Arcmap



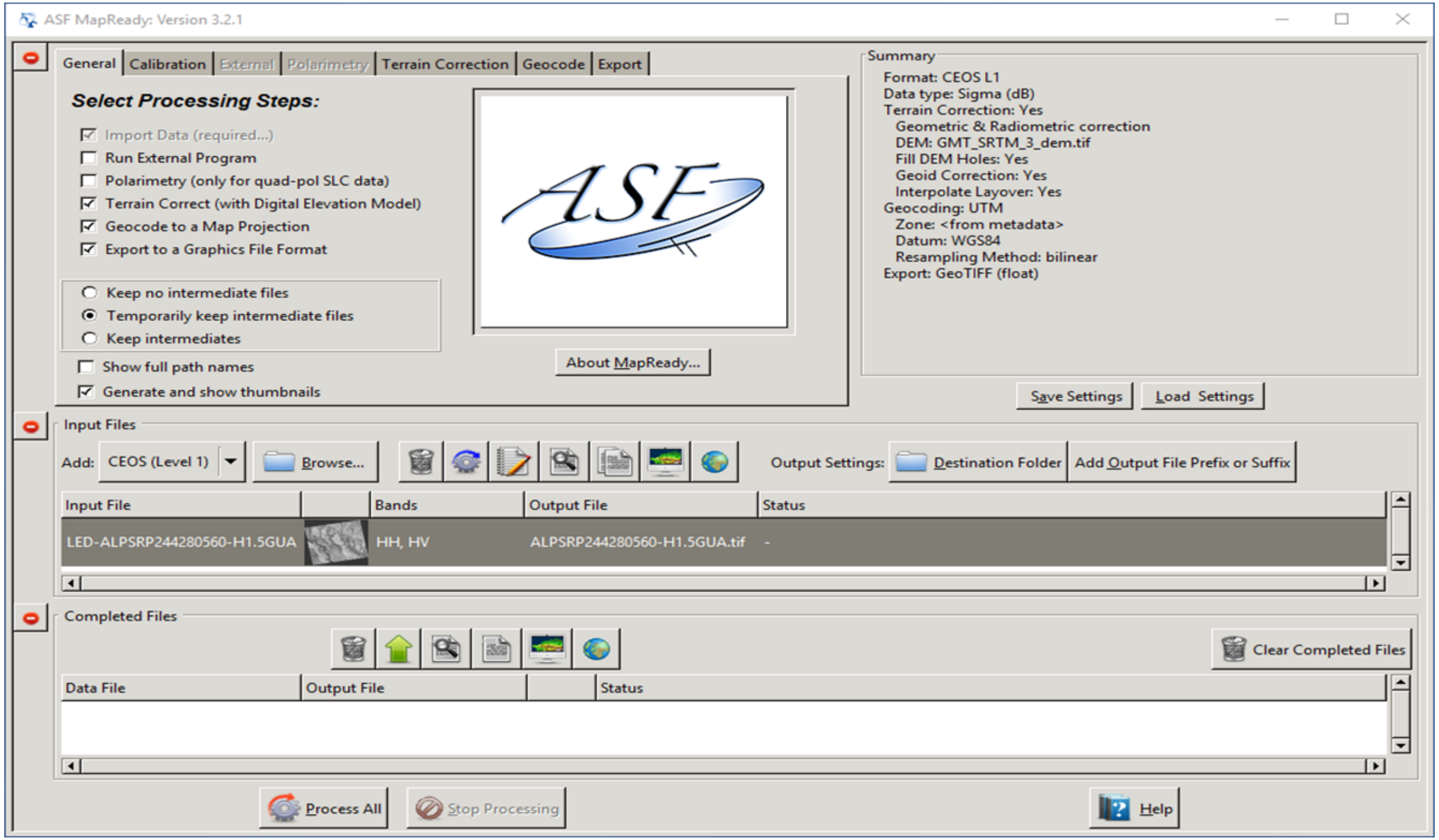
Right click on this layer and select Data >> export data. Then convert it as a tiff file in your desired folder



Pre process the ALOS PALSAR images

ASF Mapready

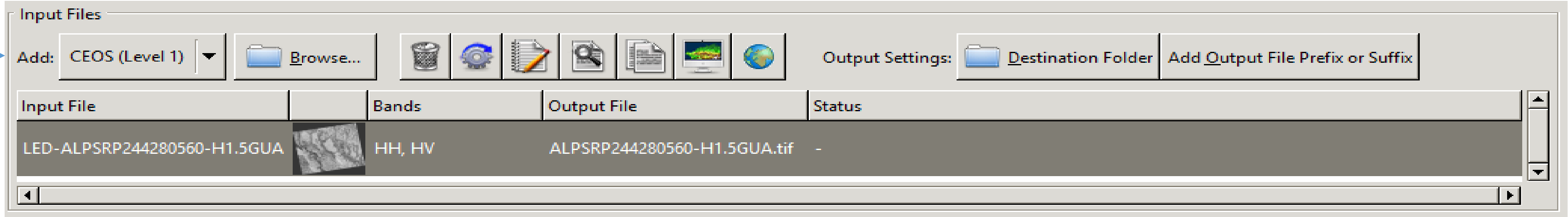
- Install the setup file for ASF Mapready software given in the software setup folder



Pre processing cont…..

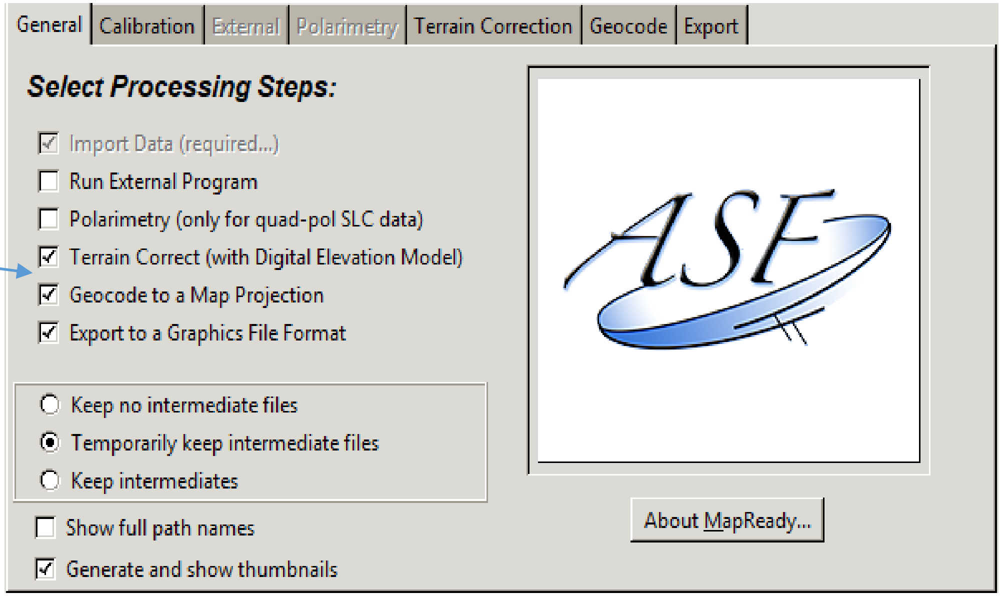
- Browse the CEOS formatted ALOS PALSAR level 1.5 Data

Select CEOS level 1 and browse the data folder

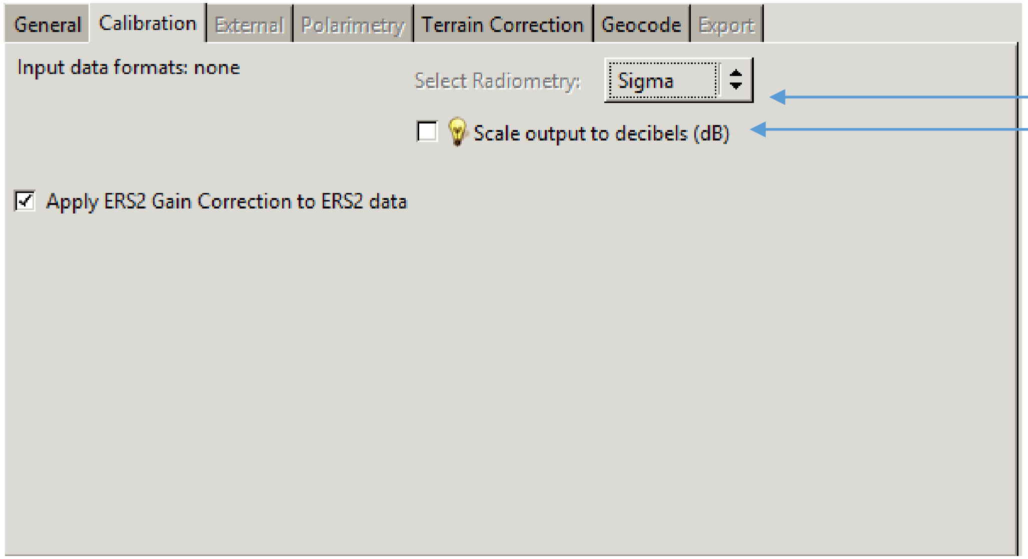


- Tic the following options and get the final sigma naught images. (not in dB values)

Tick these options



Pre processing cont…..

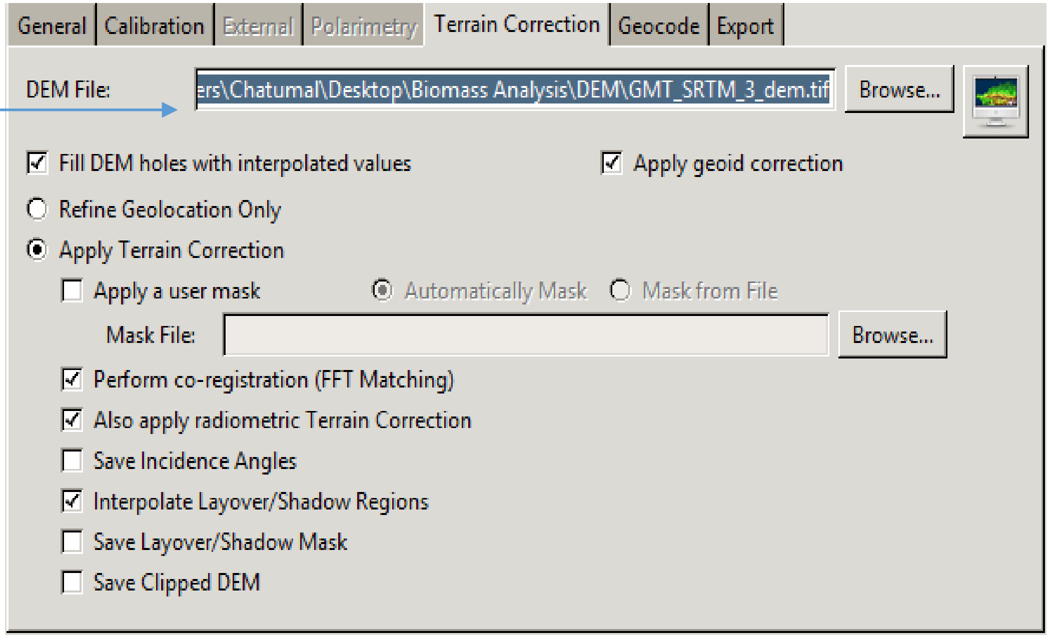


Choose Sigma

Do not tick this. Keep it as it is.

Browse your DEM file Here

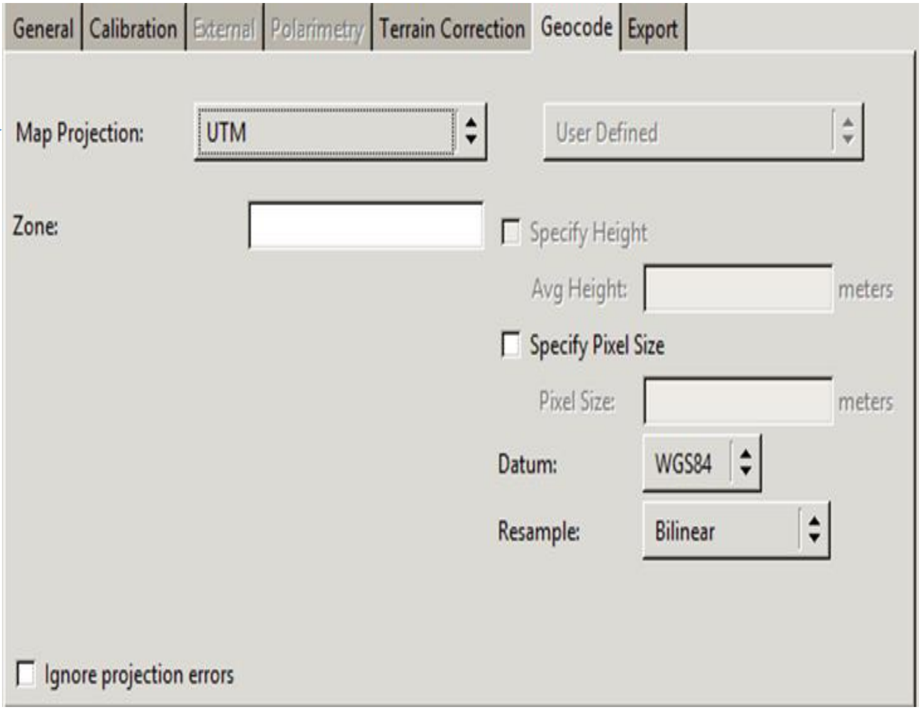
Select the all the options shown in this tab



Pre processing cont…..

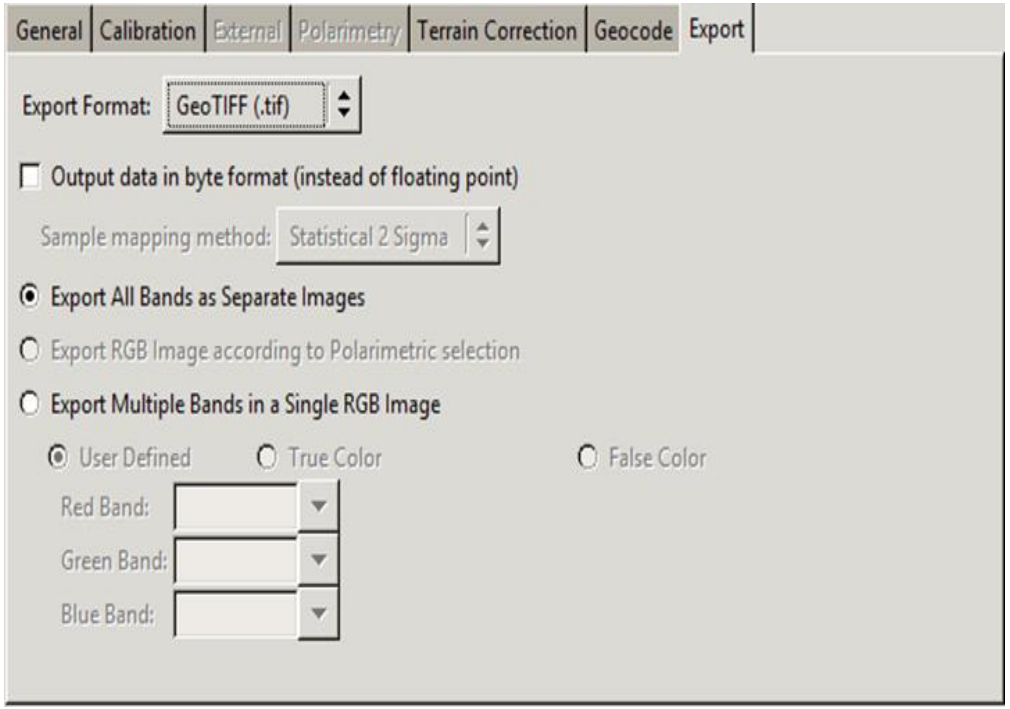
- Geocode the product in the UTM projection

Select UTM



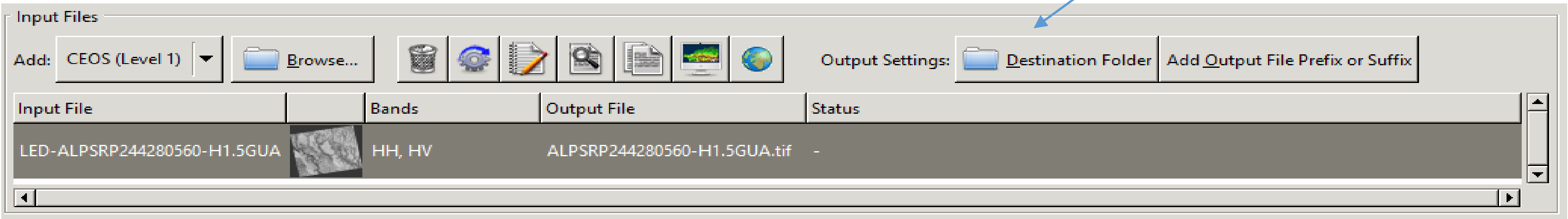
- Export the product as a Geo-tiff product.

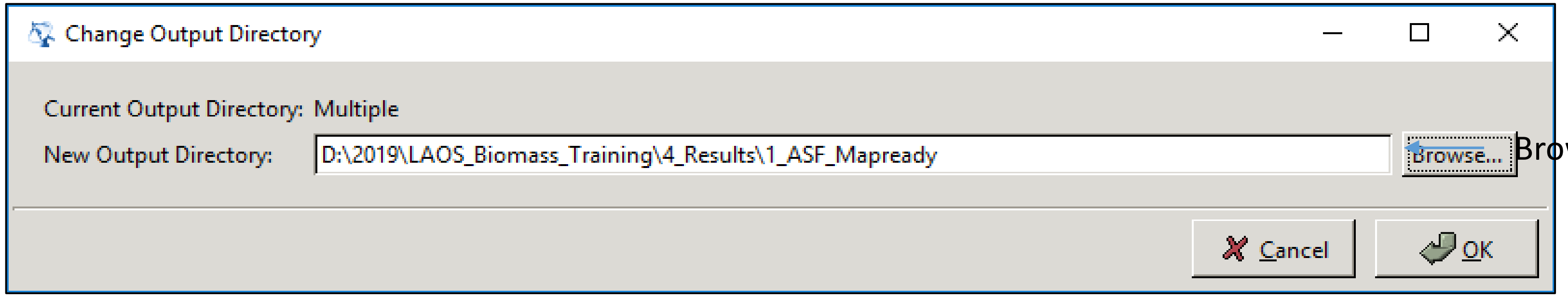
Select Export as Geotiff



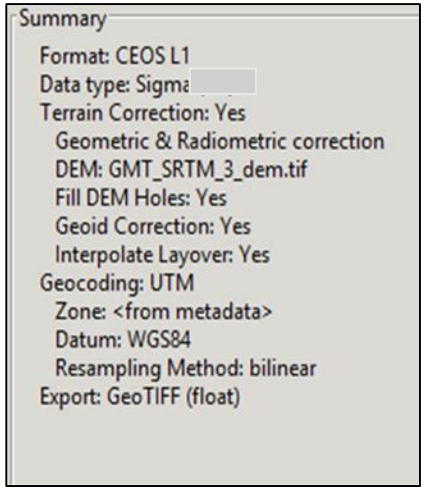
Pre processing cont…..

Select the output folder





Browse and press OK

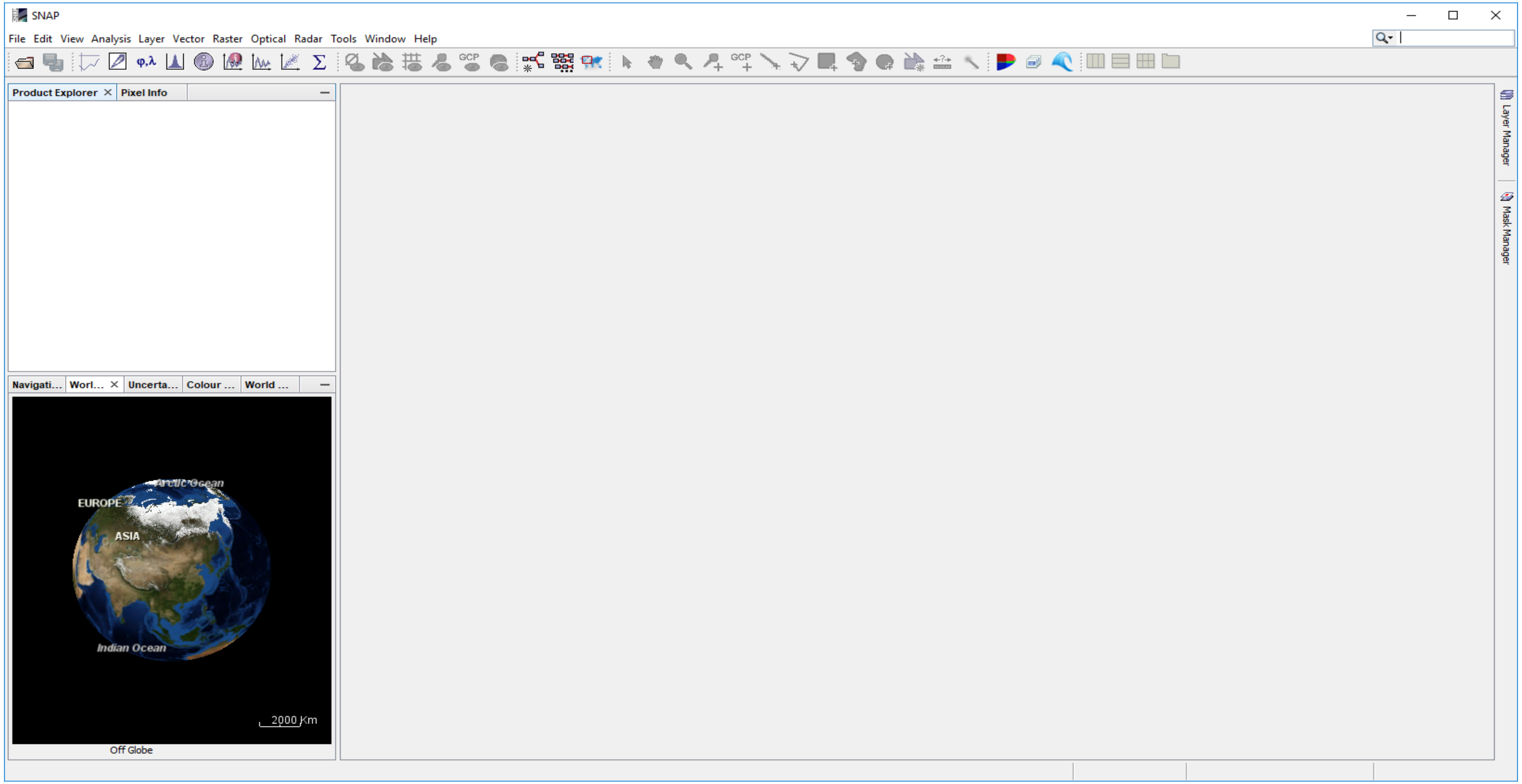


This will be the summary of your product

Speckle filtering

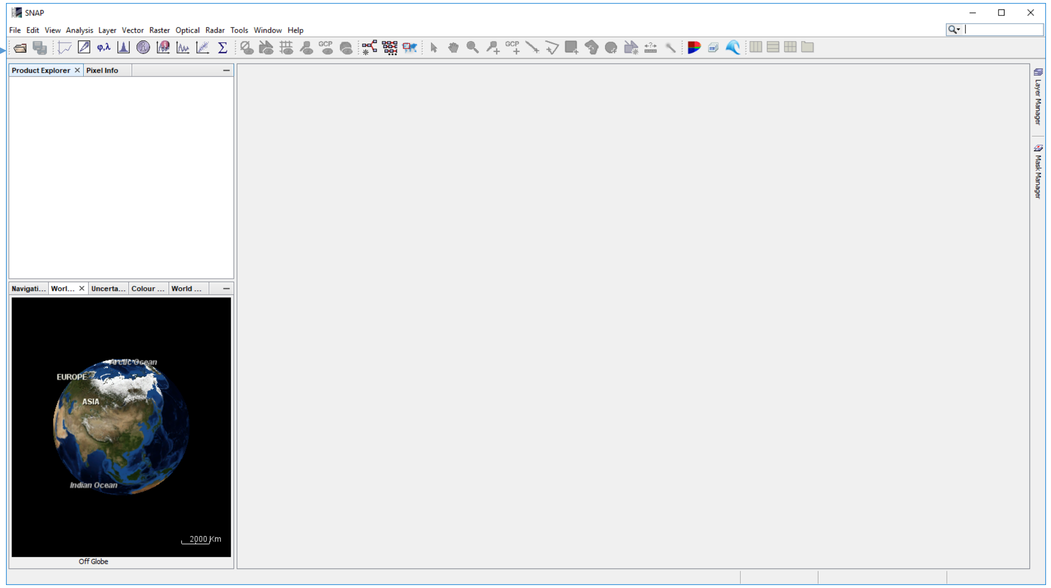
SNAP

- First install the SNAP software using the setup file in the software setup folder.

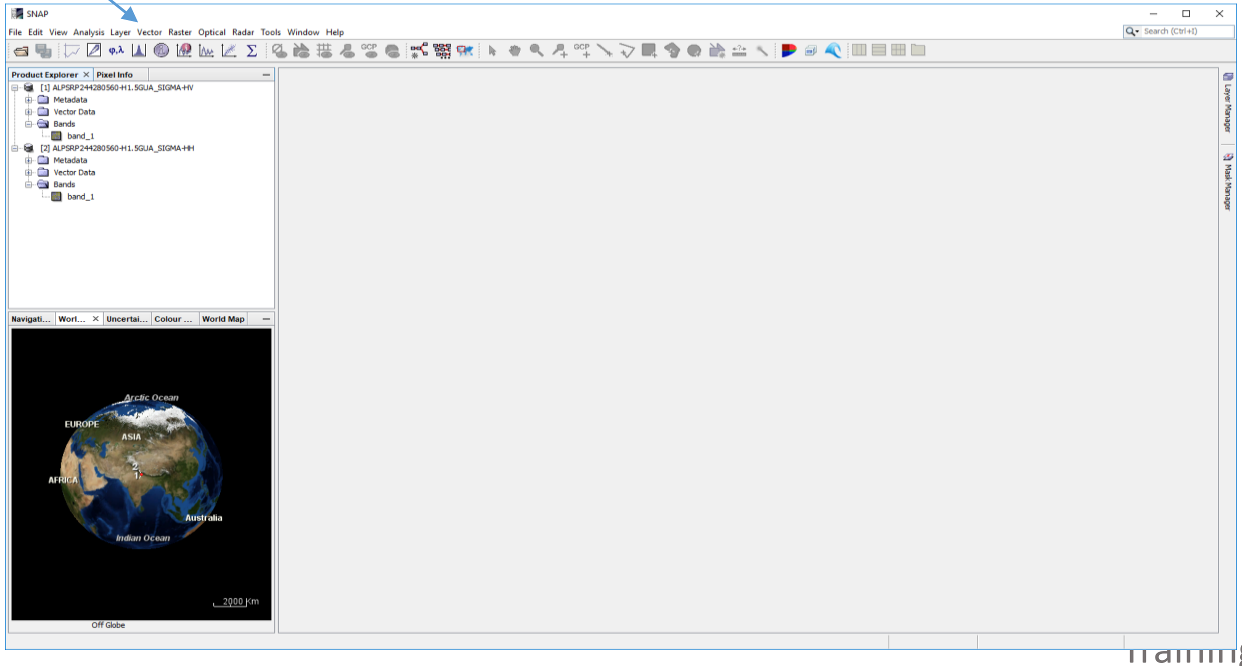


SNAP speckle filtering cont…..

Open the product you got from the ASF Mapready.
(ALPSRP244280560-H1.5GUA_SIGMA-HH.tif,
ALPSRP244280560-H1.5GUA_SIGMA-HV.tif)



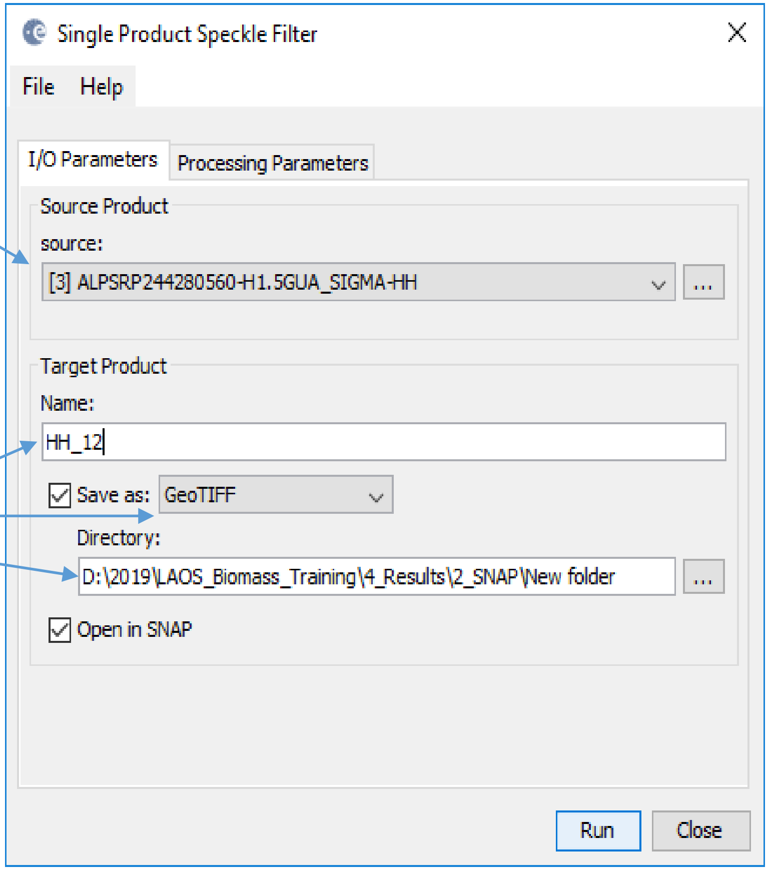
- 1)Click or radar tab.
- 2)Select single product speckle filtering



SNAP speckle filtering cont…..

1). Select the product you want to filter

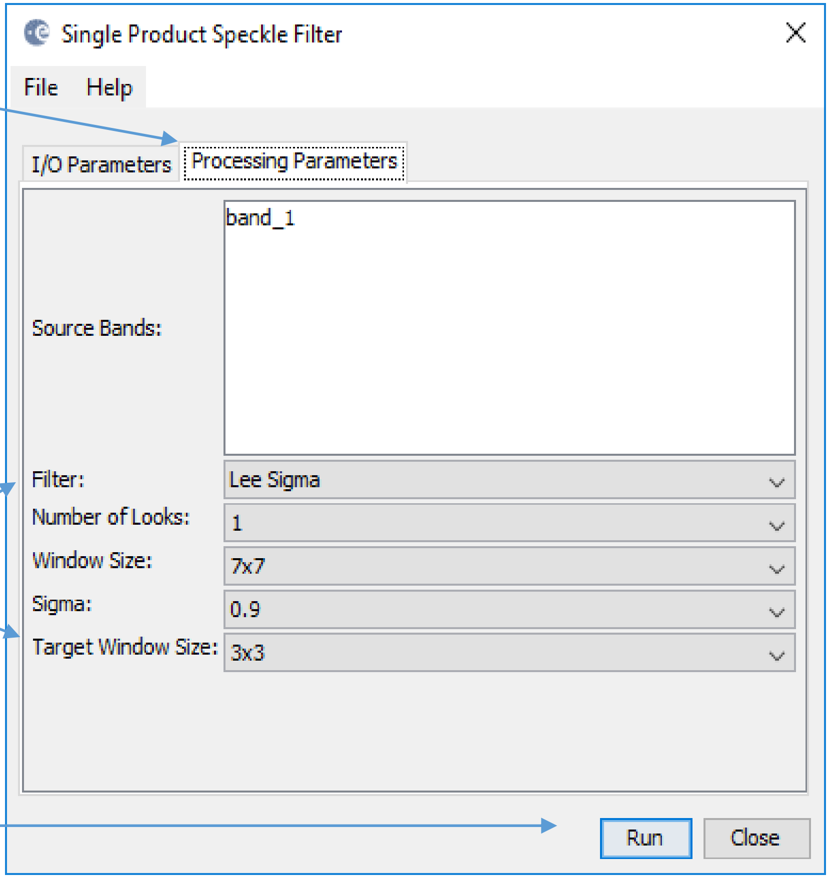
2).Give the output name, format and location



3). Select processing parameters

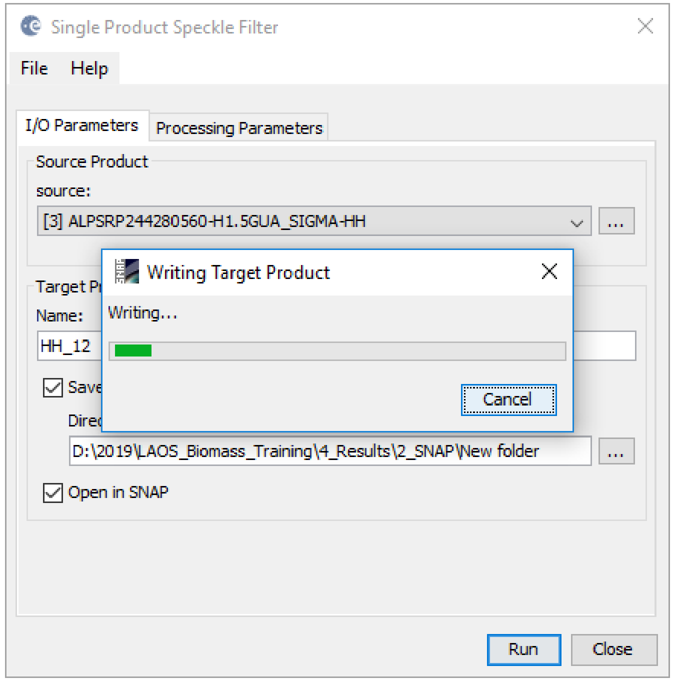
4). Select filter type and the target window size

5). Click Run



SNAP speckle filtering cont…..

Running the speckle filtering



Speckle Filtered Image Result

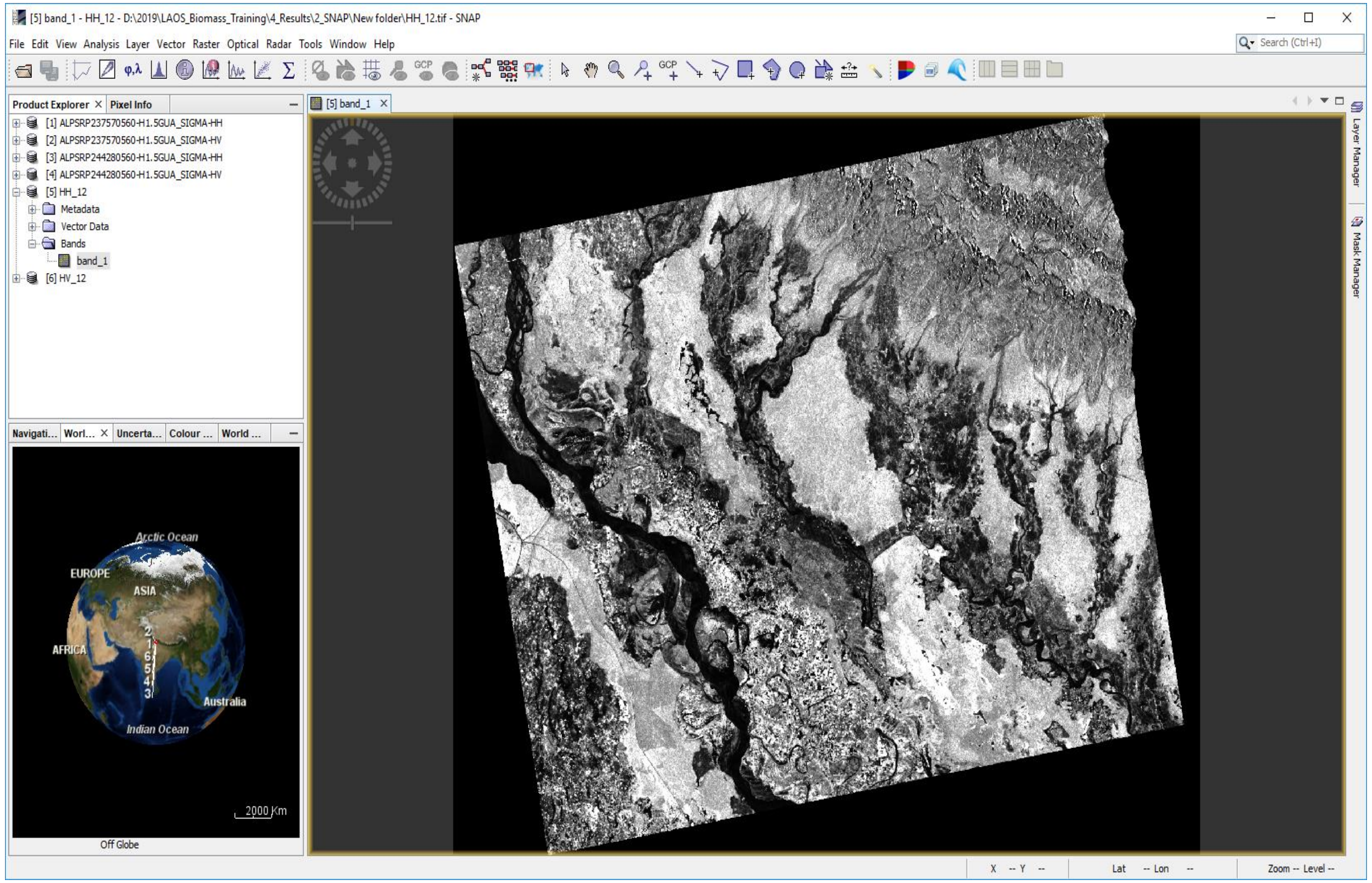


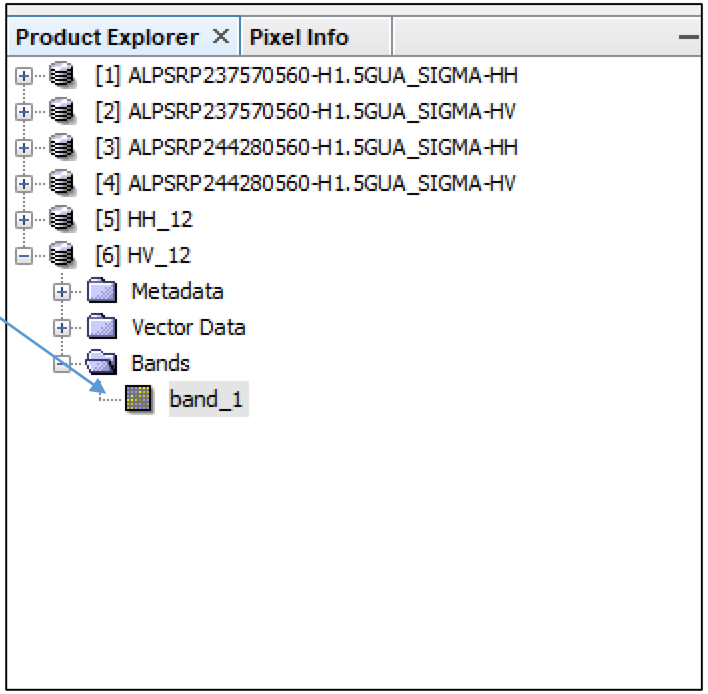
Image conversion to dB and band extraction

dB conversion

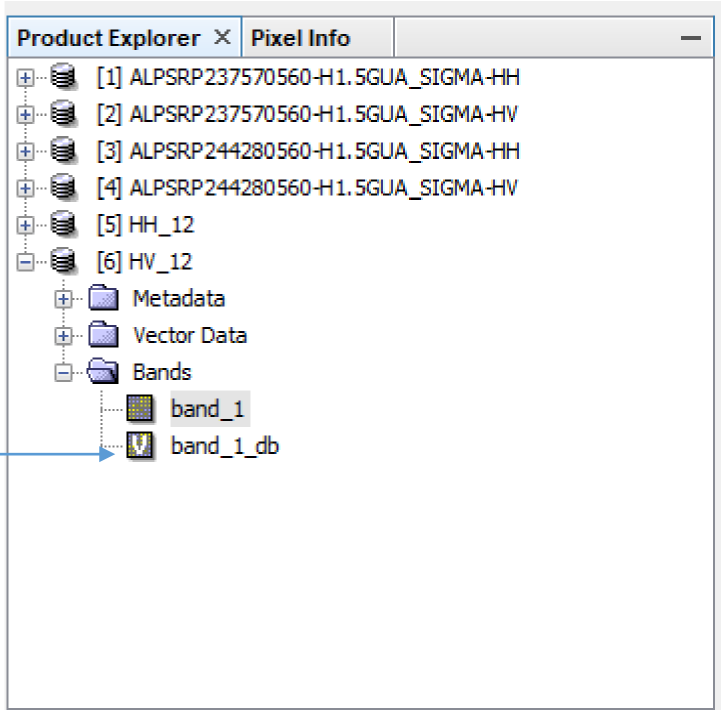
Now you can convert the speckle filtered image DN values to the **Sigma Naught dB** values.

1). Select the resulted band.

2). Right click on the band and click on **linear to/from dB**



Band in dB



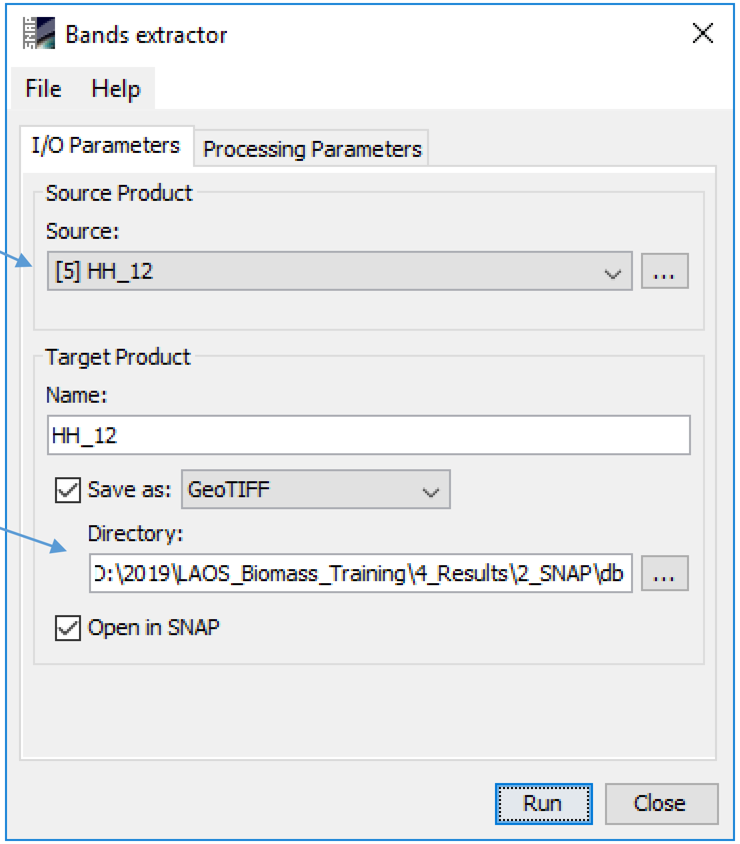
Band extraction

Now you can extract the speckle filtered dB band and export it into a geotiff file.

1). Goto the optical tab in SNAP and select Band Extractor

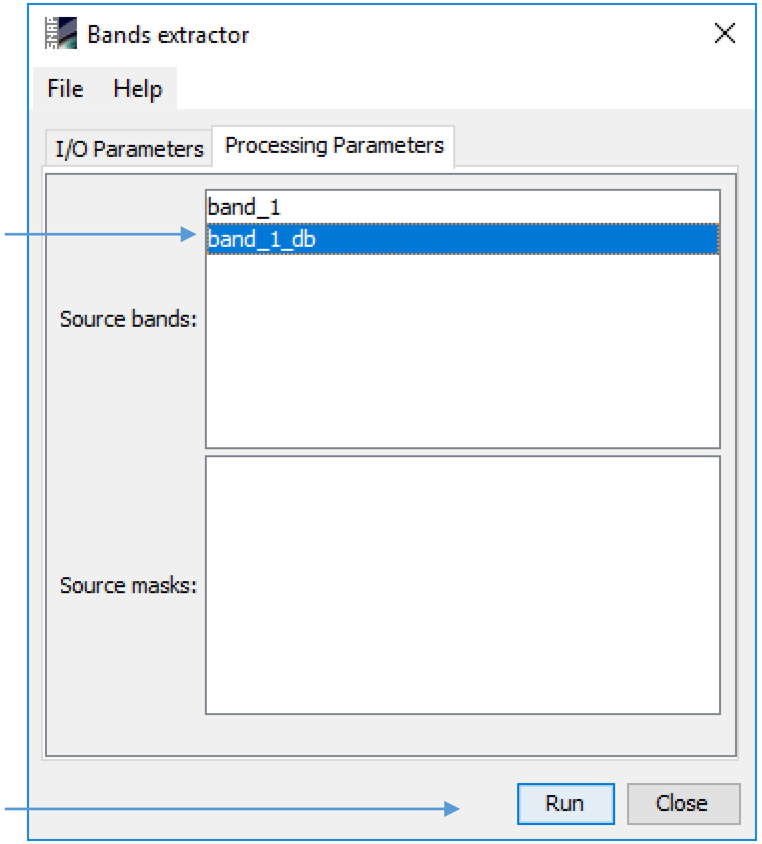
2). Select the filtered product

3).Give the output name, format and location



4).Select the band you want extract. (dB)

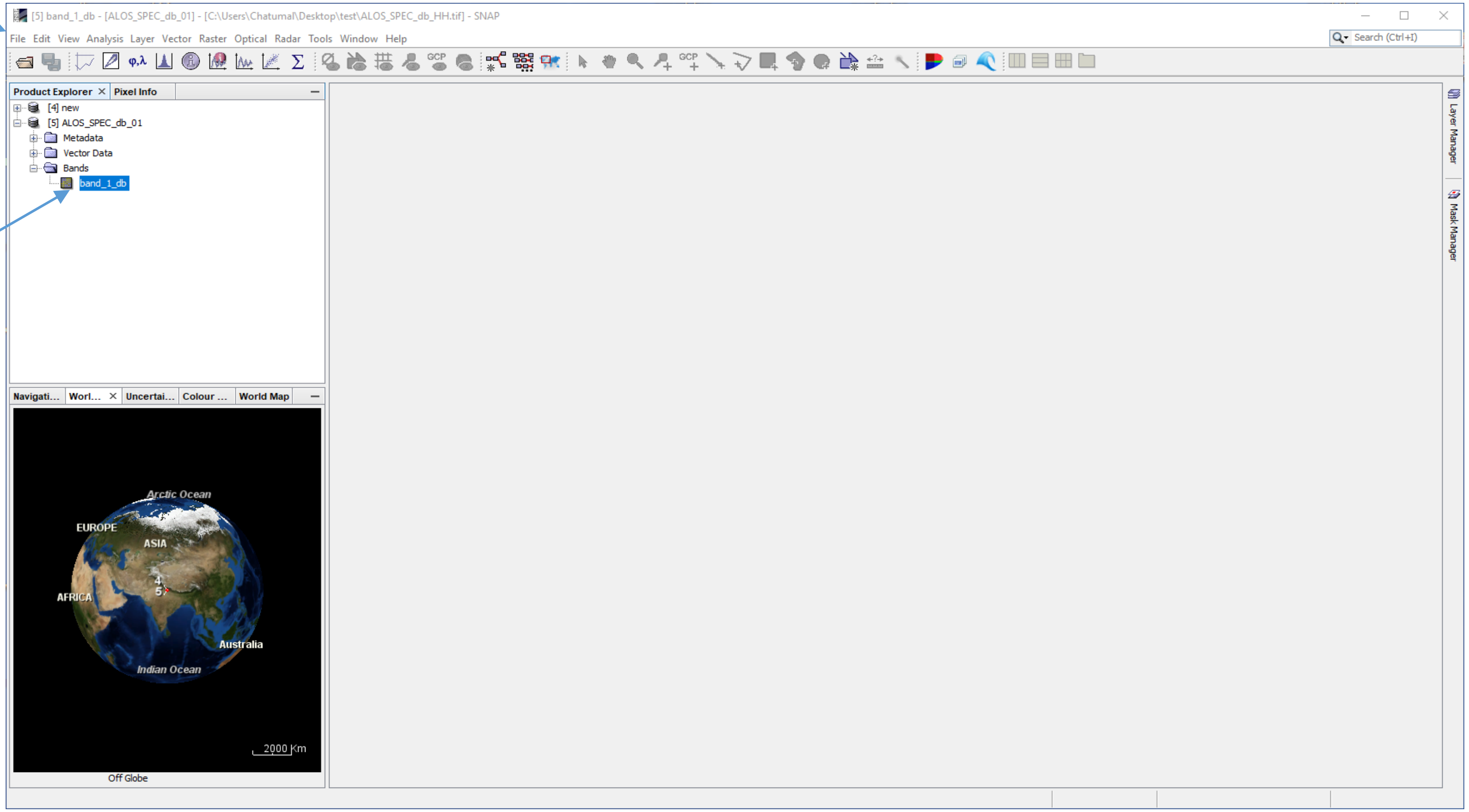
5).click Run



Export the final preprocessed image as a geotiff from SNAP

2).Click on the file tab.
File>> Export>>Geotiff

1). Select the extracted band from the product explorer tab



Biomass Estimation Analysis

1).Analysis of Lidar data using R

```
#install.packages("raster")
#install.packages("rgdal")
#install.packages("sp")
#install.packages("ggplot2")
#install.packages("mosaic")
#install.packages("mapview")

#setwd()

library(raster)
library(rgdal)
library(ggplot2)
library(mosaic)
library(mapview)

#importing rasters
dsm<- raster("D:\\2019\\LAOS_Biomass_Training\\2_Training_Data\\Lidar\\dsm_5m.tif")
dtm<- raster("D:\\2019\\LAOS_Biomass_Training\\2_Training_Data\\Lidar\\dtm_5m.tif")
NAvalue(dsm)=0
NAvalue(dtm)=0
```

Analysis of Lidar data using R cont….

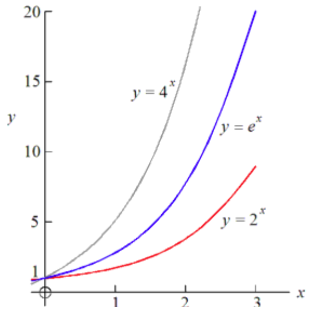
```
#Generation of canopy height raster
canopy_H = dsm - dtm

canopy_H[canopy_H < 0 ]   <- 0

writeRaster(canopy_H,"D:\\2019\\LAOS_Biomass_Training\\4_Results\\3_R_Lidar\\canopy_H.tif",
          format="GTiff",datatype='FLT4S', overwrite=TRUE)
plot(canopy_H)
#canopy_H


#CHM resample it to different resolutions (10m)
canopy_H10 = projectRaster(canopy_H,res=10,crs = crs(canopy_H), method='ngb')


writeRaster(canopy_H10,"D:\\2019\\LAOS_Biomass_Training\\4_Results\\3_R_Lidar\\canopy_H10.tif",
              format="GTiff", datatype='FLT4S',overwrite=TRUE)

#Importing the plot shape file
plots=readOGR("D:\\2019\\LAOS_Biomass_Training\\2_Training_Data\\Ground_Plots\\Plot.shp",stringsA
sFactors = FALSE)
```

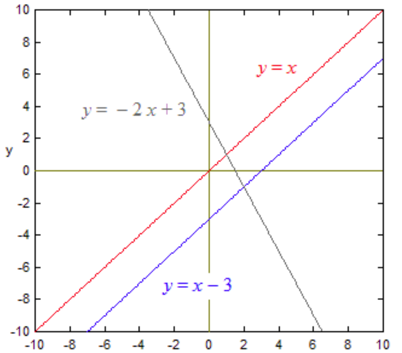
Analysis of Lidar data using R cont….

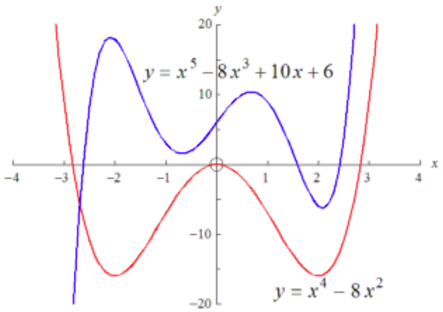
```
#Plotting the shape file
plot_map <- ggplot() + geom_polygon(data = plots, aes(x =
long, y = lat, group = group), colour = "black")
plot_map

#View it on a base map
mapviewOptions(basemaps = c("Esri.WorldImagery"))
mapview(plots, color = "red")

#Extract Mean Canopy Heght values from the canopy height layer
MCH = extract(canopy_H10, plots,fun = mean, na.rm = TRUE)
MCH=data.frame(MCH)

##Finding the linear correlation between MCH and AGB
#Combining the shape file data and the extracted data
write.table(plots@data,"D:\\2019\\LAOS_Biomass_Training\\4_Results\\3_R_Lidar\\Data_shp.csv",quote=F,row.na
mes=F,sep=",")
data_shp=read.csv("D:\\2019\\LAOS_Biomass_Training\\4_Results\\3_R_Lidar\\Data_shp.csv")
Final_Data=cbind(data_shp,MCH)

cor.test(Final_Data$AGB,Final_Data$MCH)
plot(Final_Data$MCH,Final_Data$AGB)
```

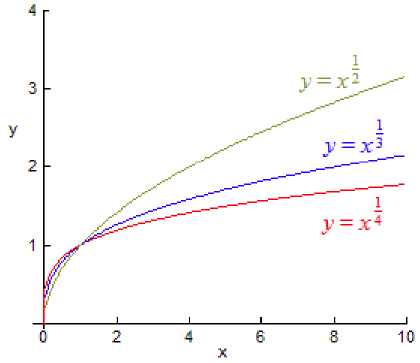
Curve types need for the analysis

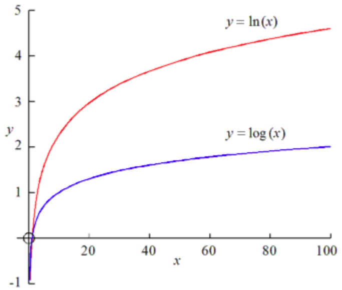
- Linear functions
- Exponential functions
- Logarithmic functions
- Polynomial functions
- Power functions





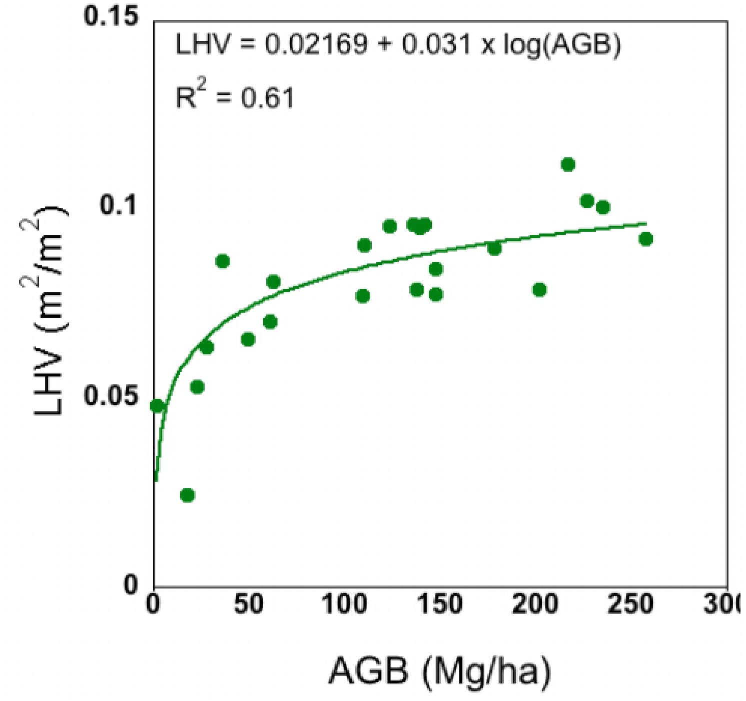


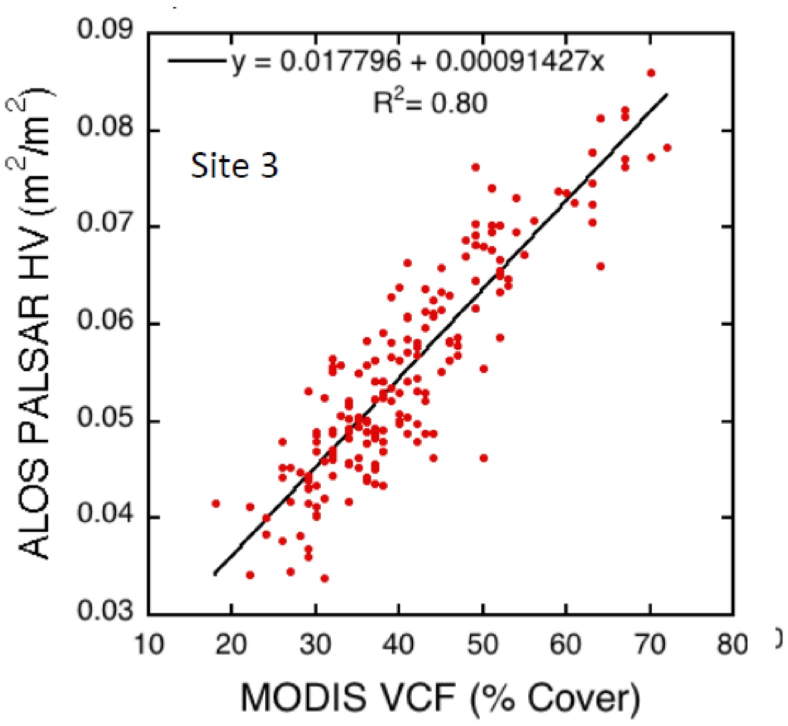


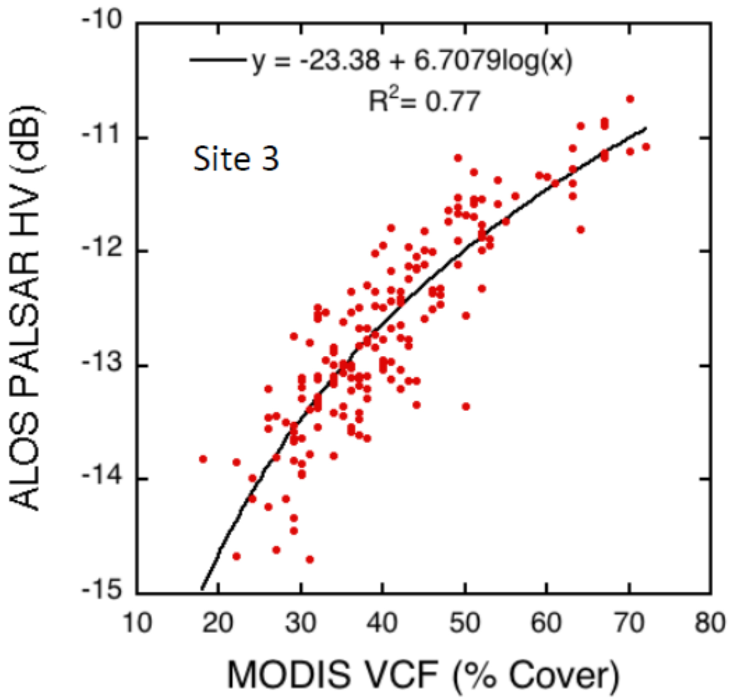


Curve fitting to the data set

100 m Resolution







Analysis of Lidar data using R cont….

```
#Estimation of goodness of fit
##Linear fitting
f1=fitModel(MCH~C+M*(AGB),data = Final_Data)
plotPoints(MCH~AGB,data = Final_Data)
plotFun(f1(AGB)~ AGB,add = TRUE, col = 'red')
y1=f1(Final_Data$AGB)
cor.test(y1~Final_Data$MCH)

##Curve fitting
f2=fitModel(MCH~A+B*log(AGB),data = Final_Data)
plotPoints(AGB~MCH,data = Final_Data)
plotFun(f2(AGB)~ AGB,add = TRUE, col='blue')
coef(f2)
y2=f2(Final_Data$AGB)
cor.test(y2~Final_Data$MCH)

#same fitting with "nls" command
#f22=nls(MCH~A+B*log(AGB),data = Final_Data)
#x<- predict(f22, newdata=Final_Data$AGB)
#cor.test(x~Final_Data$MCH)
```

Analysis of Lidar data using R cont….

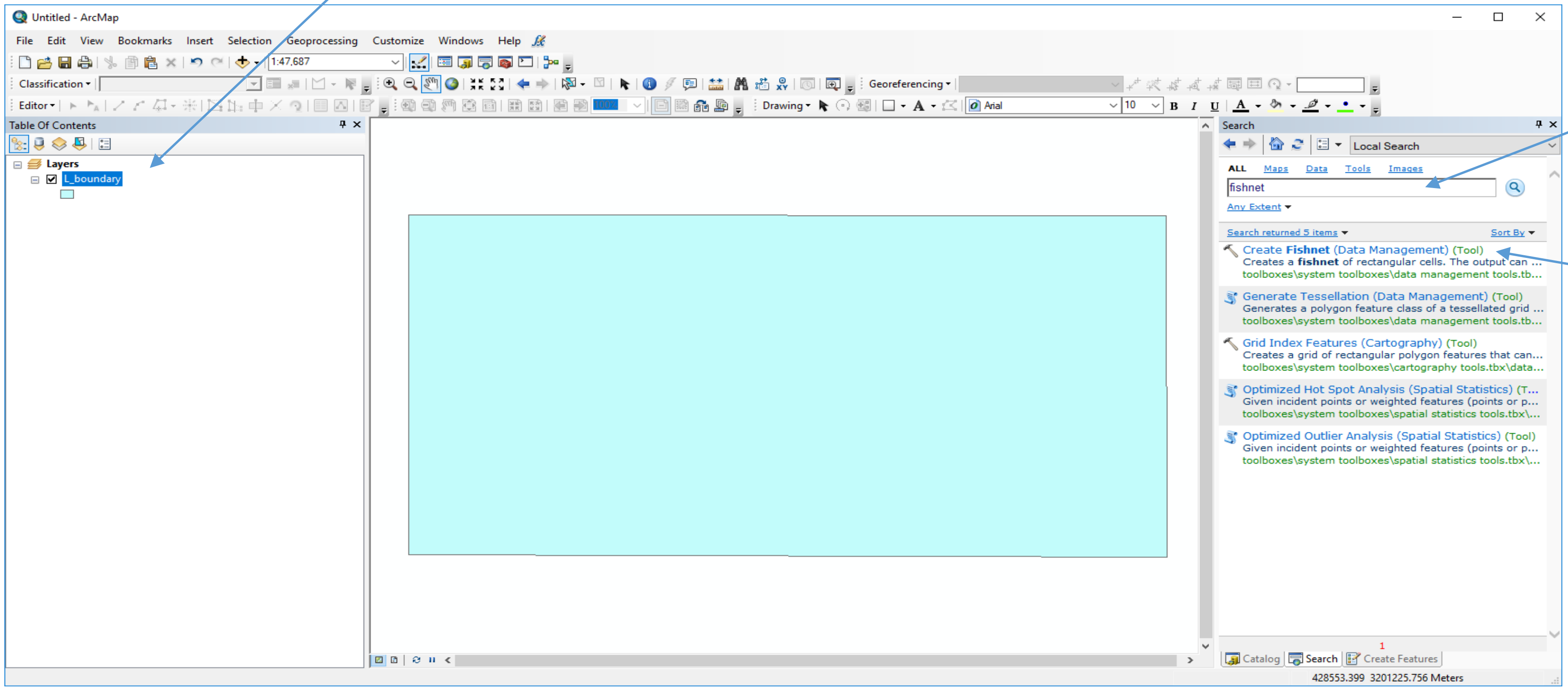
```
#A=-22.188819
#B= 7.151892
#MCH~A+B*log(AGB)
#AGB~exp((MCH-A)/B)


#Biomass map creation from the fitted model
bio=exp((canopy_H10+22.188819)/7.151892)
NAvalue(bio)=0
bio[bio>1000] <- 1000

writeRaster(bio,"D:\\2019\\LAOS_Biomass_Training\\4_Results\\3_R_Lidar\\LidarBiomass.tif" ,
          format="GTiff",datatype='FLT4S', overwrite=TRUE)

plot(bio)
```

Creating a grid network to densify Plots

Using fishnet tool in Arcmap

We can create a grid network as plots and extract the biomass values into these grid points.

1).Open Arcmap and insert the “L_boundary” shape file from the Training data



2).Search for "fishnet" in the search bar

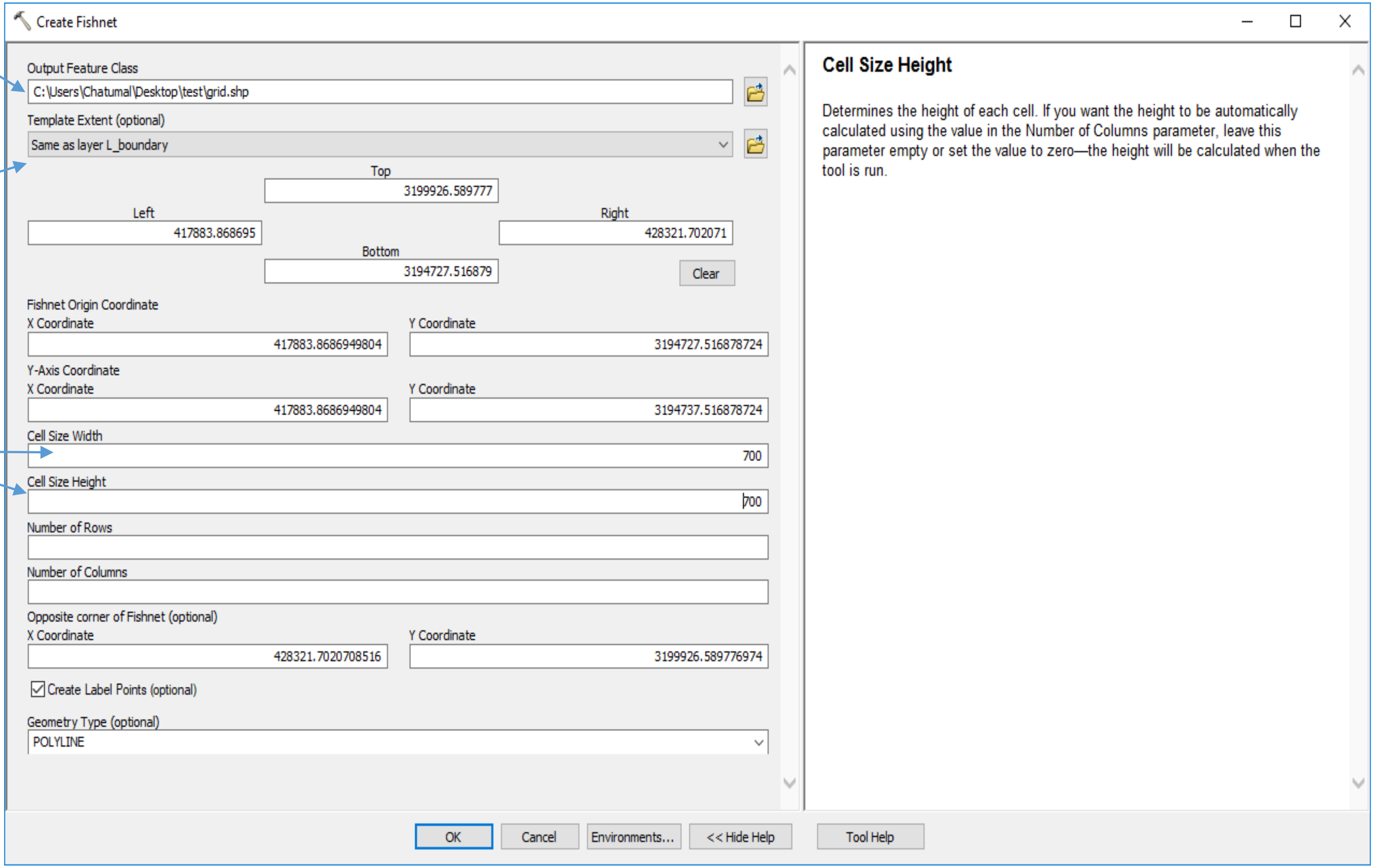
3).Click on **Create Fishnet (data management) tool**

Fishnet tool cont…..

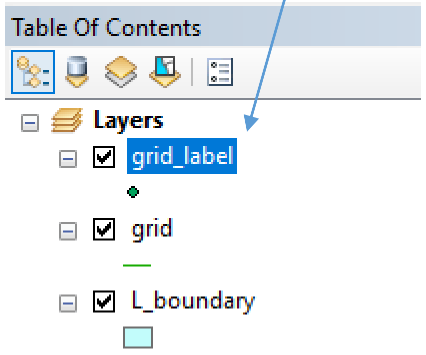
1).Give the output location

2).Select the L_boundary layer

3).Give the cell size
As 700*700 and click OK



This is the created grid point shape file. You can write click and export it to a new one if you need.



Biomass Estimation Analysis

2).Analysis of ALOS PALSAR data using R

```
#install.packages("raster")
#install.packages("rgdal")
#install.packages("sp")
#install.packages("ggplot2")
#install.packages("mosaic")
#install.packages("mapview")

#setwd()

library(raster)
library(rgdal)
library(ggplot2)
library(mosaic)
library(mapview

#Importing the speckle filtered ALOS PALSAR HH, HV data
HH_12=raster("D:\\2019\\LAOS_Biomass_Training\\4_Results\\2_SNAP\\dB_images\\HH_12.tif")
HV_12=raster("D:\\2019\\LAOS_Biomass_Training\\4_Results\\2_SNAP\\dB_images\\HV_12.tif")
```

```r
#Resampling HH_12 and HV_12 into different resolution (40m)
HV_40 =projectRaster(HV_12,res=40,crs = crs(HV_12), method='ngb')
writeRaster(HV_40, filename="D:\\2019\\LAOS_Biomass_Training\\4_Results\\4_R_ALOS\\HV_40.tif",
            format="GTiff", overwrite=TRUE)
names(HV_40)<-"HV_40"


HH_40 =projectRaster(HH_12,res=40,crs = crs(HH_12), method='ngb')
writeRaster(HH_40, filename="D:\\2019\\LAOS_Biomass_Training\\4_Results\\4_R_ALOS\\HH_40.tif",
            format="GTiff", overwrite=TRUE)
names(HH_40)<-"HH_40"

#Creating a list of rasters
raster_list= list(HH_12,HV_12,HH_40,HV_40)

#Importing the densified plot shapefile
dense_plots=readOGR("D:\\2019\\LAOS_Biomass_Training\\2_Training_Data\\Ground_Plots\\
dense_plot_700.shp",stringsAsFactors = FALSE)
```

```
##importing the lidar derived biomass raster
L_bio=raster("D:\\2019\\LAOS_Biomass_Training\\4_Results\\3_R_Lidar\\LidarBiomass.tif")
L_bio_40 =projectRaster(L_bio,res=40,crs = crs(L_bio), method='ngb')


##extracting AGB values for Denseplots from biomass map createed from Lidar
#Extract Mean Canopy Heght values from the canopy height layer
AGB = extract(L_bio_40, dense_plots,fun = mean, na.rm = TRUE)
AGB=data.frame(AGB)

#Combining the shape file data and the extracted data
write.table(dense_plots@data,"D:\\2019\\LAOS_Biomass_Training\\4_Results\\4_R_ALOS\\Data_shp.csv",
quote=F,row.names=F,sep=",")

data_shp=read.csv("D:\\2019\\LAOS_Biomass_Training\\4_Results\\4_R_ALOS\\Data_shp.csv")
Final_Data_ALOS=cbind(data_shp,AGB)
```

```
##Croping the rasters to the extent of Lidar_boundary
for (x in raster_list){
  crop <- crop(x, L_bio_40)
  writeRaster(crop, file.path("D:\\2019\\LAOS_Biomass_Training\\4_Results\\4_R_ALOS\\crop_rasters" ,
names(x)), bylayer=TRUE,
              format="GTiff", overwrite=TRUE)
}

#creating a cropped raster list
current.list <- list.files(path="D:\\2019\\LAOS_Biomass_Training\\4_Results\\4_R_ALOS\\crop_rasters",
                          pattern =".tif$", full.names=TRUE)

raster_list <- c()
for (x in current.list){
  y=raster(x)
  raster_list <- append(raster_list, y)
}
```

```
#extraction of backscatter values
#Extracting  the backscatter values for AGB plots

BS_values =list()
for (raster in raster_list){
  backscatter <- extract(raster,dense_plots,na.rm=TRUE)
  #Append value to the list
  BS<- as.data.frame(backscatter)
  BS_values<-append(BS_values,BS)
}

BS_values =data.frame(BS_values)

#Getting the name of Rasters
raster_names = list()
for (ras in raster_list){
  x=names(ras)
  raster_names=append(raster_names,x)
}
colnames(BS_values)= raster_names
```

```
#Combining the plot AGB values and the Raster backsactter values to a Dataframe.

AGB_final=cbind(AGB,BS_values)
View(AGB_final)
AGB_final=na.omit(AGB_final)
write.csv(AGB_final,"D:\\2019\\LAOS_Biomass_Training\\4_Results\\4_R_ALOS\\AGB_final.csv")


#Pearson correlation for identify the basic relationships of the AGB values and the Data.

cor.test(AGB_final$AGB,AGB_final$HV_40, method = "pearson")
cor.test(AGB_final$AGB,AGB_final$HV_12, method = "pearson")

plot(AGB_final$AGB,AGB_final$HV_40,xlim=c(20,25))
```

```
##Curve fitting
f2=fitModel(HV_40~A+B*log(AGB),data = AGB_final)
plotPoints(HV_40~AGB,data = AGB_final)
plotFun(f2(AGB)~ AGB,add = TRUE, col='blue')
coef(f2)
y2=f2(AGB_final$AGB)
cor.test(y2~AGB_final$HV_40)


#same fitting with "nls" command
#f22=nls(MCH~A+B*log(AGB),data = Final_Data)
#x<- predict(f22, newdata=Final_Data$AGB)
#cor.test(x~Final_Data$MCH)


#A=-21.052473
#B=1.746382
#BS~A+B*log(AGB)
#AGB~exp((BS-A)/B)
```

```
crop_HV40=raster("D:\\2019\\LAOS_Biomass_Training\\4_Results\\4_R_ALOS\\crop_rasters\\HV_40.tif")


#Biomass map creation from the fited model
bio=exp((crop_HV40 +21.052473)/1.746382)
NAvalue(bio)=0
bio[bio>1000] <- 1000

writeRaster(bio,"D:\\2019\\LAOS_Biomass_Training\\4_Results\\4_R_ALOS\\ALOSBiomass.tif" ,
          format="GTiff",datatype='FLT4S', overwrite=TRUE)

plot(bio)
```

Thank you!!!!