

TRUE SENSOR CAPTURED RADIANCE AND SURFACE REFLECTANCE FROM LANDSAT-8 MULTISPECTRAL IMAGE SETS

Dr. C. Rajabhushanam¹, Dr. A. R. Arunachalam²

^{1,2}Doctorate Professor, Department of CSE, Bharath Institute of Science & Technology, BIHER, Selaiyur, Chennai,

¹rajabhushanamc.cse@bharathuniv.ac.in

²hodcse@bharathuniv.ac.in

Abstract

Proper validation of multispectral image sets that are produced by Satellite Sensors, in this era of big-data computing, is an important aspect in signal processing of natural semantic scenes. In this research article we explore fundamental issues that are cropped up in radiance correction and surface reflectance measurements. We calibrate the raw Landsat-8 imagery digital numbers by using standard functional oriented modular programs in R software, for true sensor radiance and reflectance values. This important preprocessing step is essential for activities that use multispectral images for natural resources monitoring and change-detection studies. It is noted that the procedures for preprocessing can be applied in machine learning algorithms for scientific studies.

Objective

Traditionally, Landsat imagery from repositories such as National Remote Sensing Center, had systematic variance measures that varied from true calibrated values of the sensor. The pre-processing step of feature object retrieval was done to reduce noise effects, such as speckle, striping, salt-pepper, and irregularities in scan lines (aperiodic). To produce true sensor calibrated observed value, i.e., dampening of Quantization bit errors, the practice was to perform a viable model fit to the raw imagery, thereby reducing polarization effects.

This research work focuses on functional programming interfaces to implement processing measures that reduce abnormalities in sensor frame grabber techniques. We make sure that conversion of atmospheric radiance value is converted to true pixel reflectance DN. Thereby, improving scene capture, scene extraction, and scene modeling activities.

Specifically, generic functions that are stored as collection objects, methods that operate on inputs, model fitting, statistical calibration, and feature invariant transformation, are tested on Landsat-8 satellite imagery.

Related Literature Survey:

Landgrebe [1] describes a conceptual model that is conceived as the scene model, sensor model and the processing model. In each of the non-linear models studied, a fundamental view of the structure is in, storage and extraction of fixed variables (such as quantization, signal attenuation, and data range). The extraction and analysis of scene and sensor variables, is the accuracy of supervised feature methods and their parameter values to create a normalized image set. For example, Landgrebe [1], implicitly emphasizes that there is no precursor variable for data, available in the database to simulate non-lambertian forms of ground cover types. We must first simulate an environment in which scene invariant measure can be extracted by varying input parameters, such as, radiometric correction, atmospheric correction, and scene pixel normalized digital number. As the scene reflectance values do not change within band interleaving, the sensor model does not produce any variations in the results. The variations in the results are most likely from the atmospheric and geometric effects.

There are two kinds of radiometric calibration that may be used in the preprocessing steps: relative and absolute calibration. Relative method makes input data to have a sequential measurement to the next radiance value,

whereas, absolute calibration makes the preprocessed data conform to an external standard, such as measures and measurements.

Data Sources:

Input imagery from [AWS – Landsat](#) is the primary data ingest site. While selecting a specific Path number and Row number for a defined scene, we can create a spatio-temporal fixed view of the imaging dataset. So, we will obtain a sample 3-band Landsat-8 imageset located at Chennai (India), to produce true sensor ground truth.

Using machine learning software such as R, we will input a working algorithm for the following tasks -

to input a pseudo invariant sensing model

to input a histogram matching technique that equalizes scene differences temporally

to input a sensor radiometric correction method

to input a modifiable radiometric control set

By using C programming language, a generic collection object that contains – Vector, Spatial range, Time, Biophysical measurements (NDVI, LAR, PAR, Correlation, Variance, and Semivariogram), will be computed along with input feature imageset.

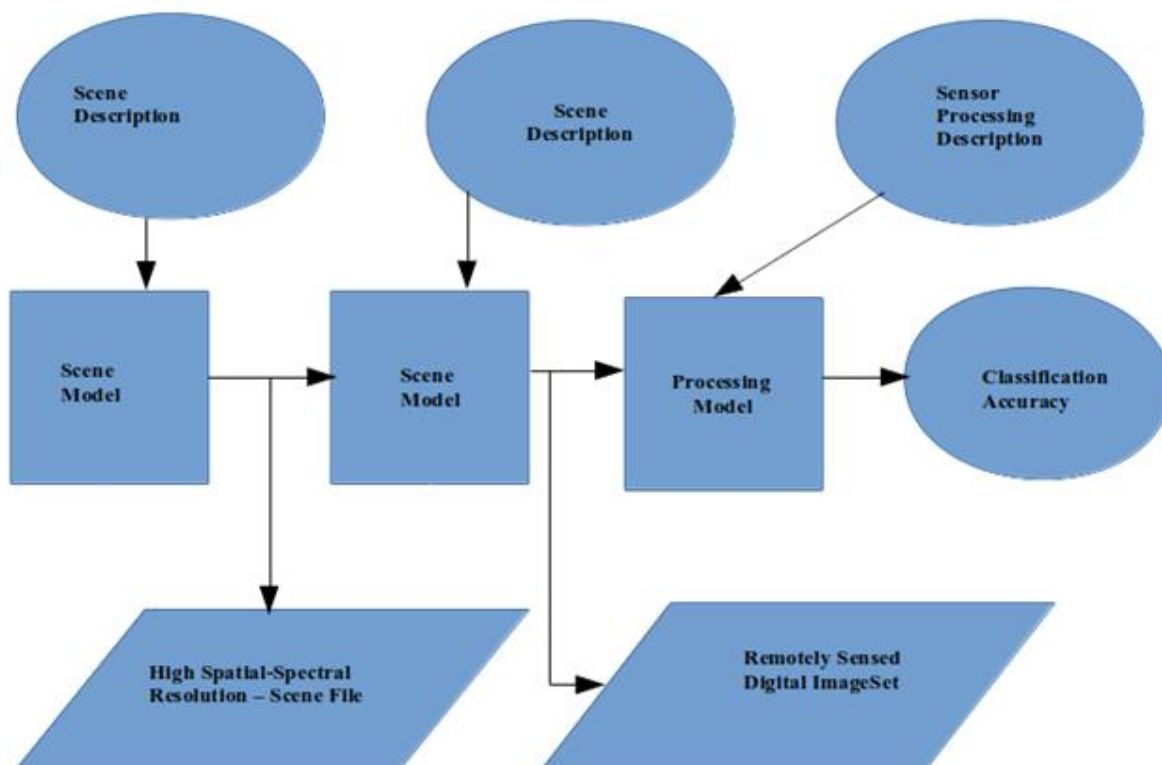


Figure 1: Scene-Sensor-Processing model for Landsat-8 Imagery

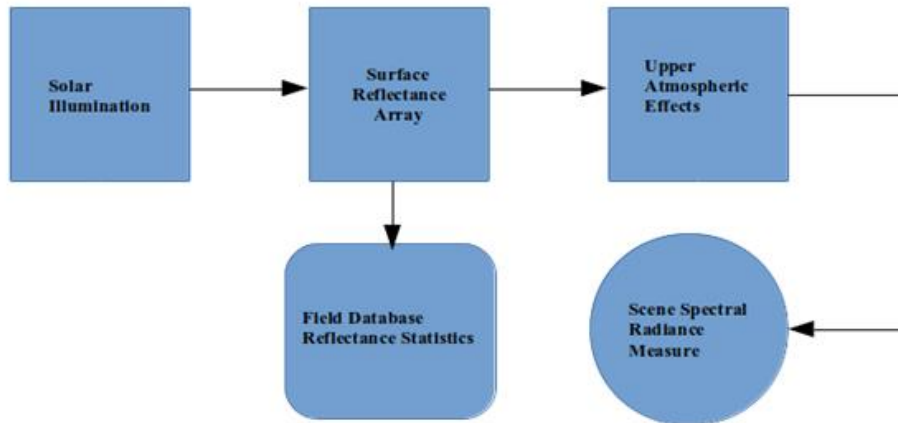


Figure 2: Sensor-Scene model of surface reflectance measure for Landsat-8 Imagery

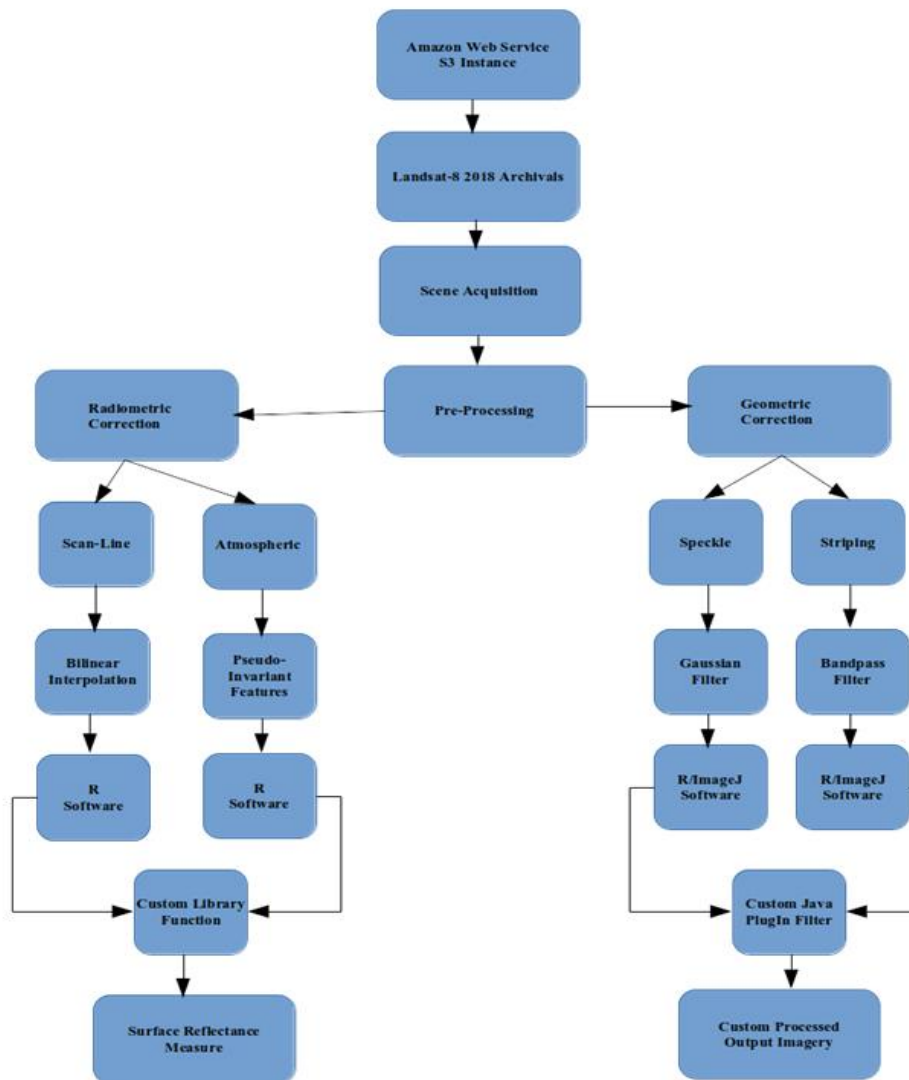


Figure 3: Research Methodology for producing true calibrated sensor

From Figure 3, it is evident that the pre-processing step involves two primary operations, namely Radiometric effects calibration and Geometric anomalies correction. Both of the operative methods can form a custom library R function that will inherit from the Rasterobject instance in R.

Once we have created a virtual function that works with the two dichotomies, we can assimilate the output result objects that are modeled as a workflow routine in R. We find that functional object-oriented structured programming in R, enables us to implement the stated objectives, of producing true sensor surface reflectance and intensity imagery from surface radiance.

Results:

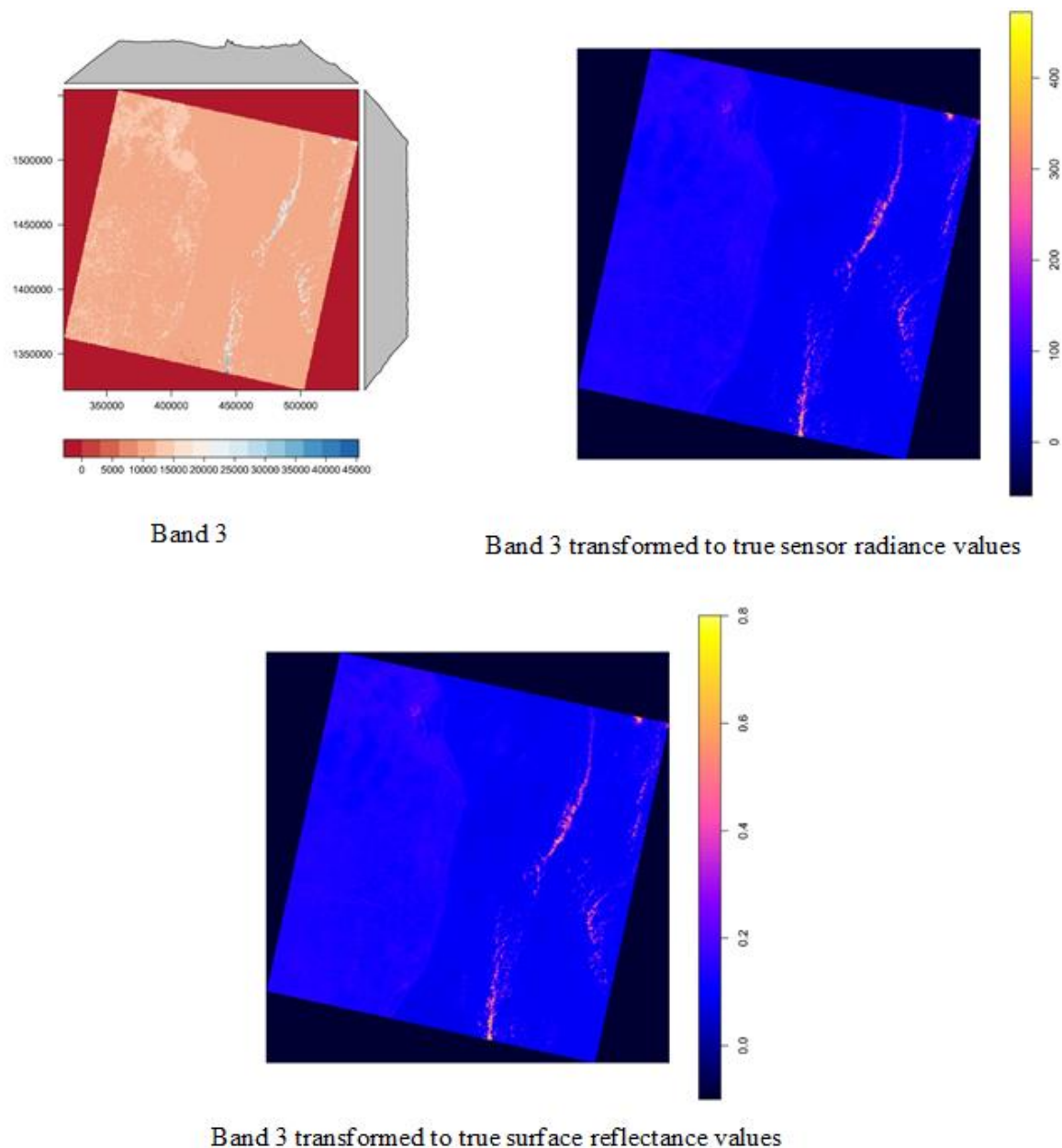
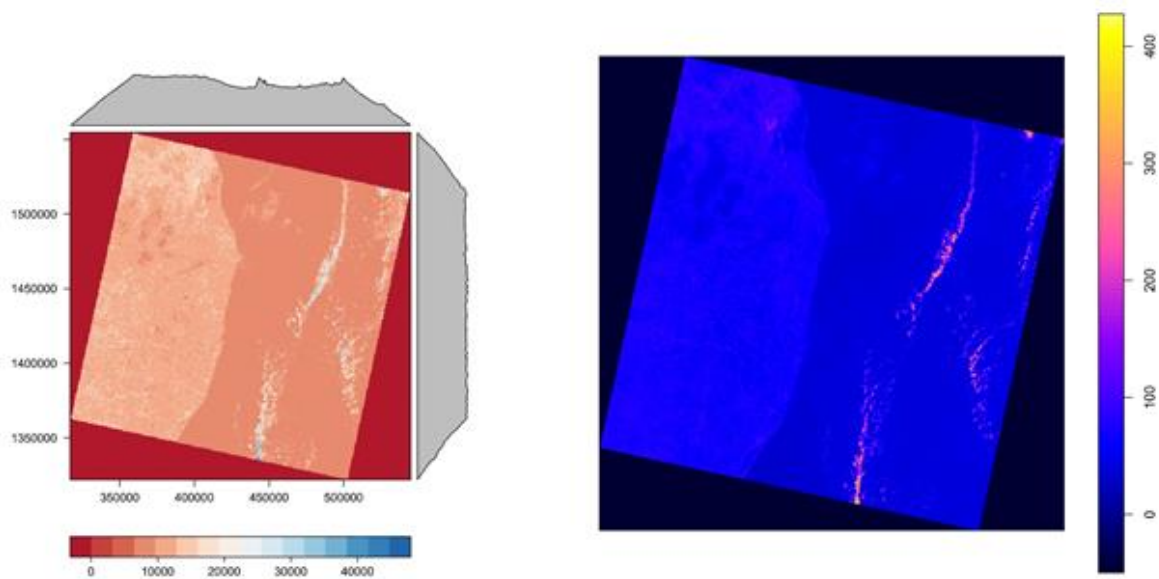
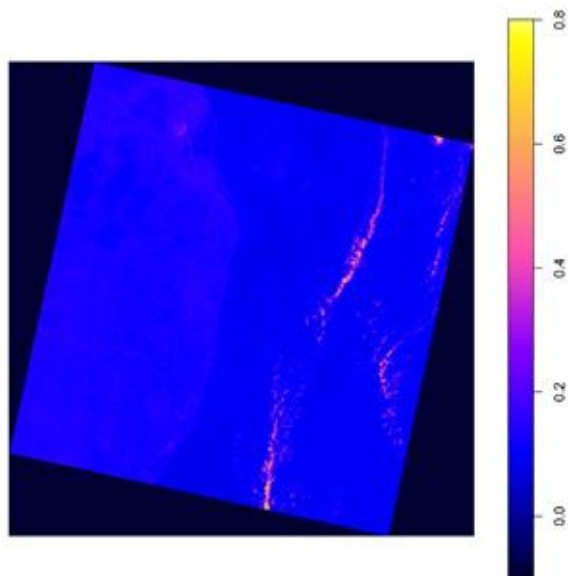


Figure 4: LC08_L1TP_142051_20170410_20170414_01_T1_B3.TIF



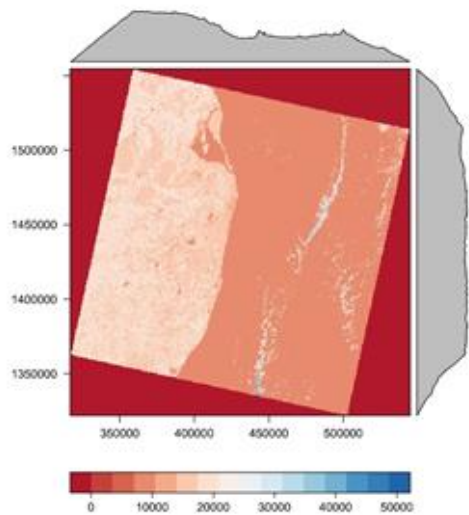
Band 4

Band 4 transformed to true sensor radiance values

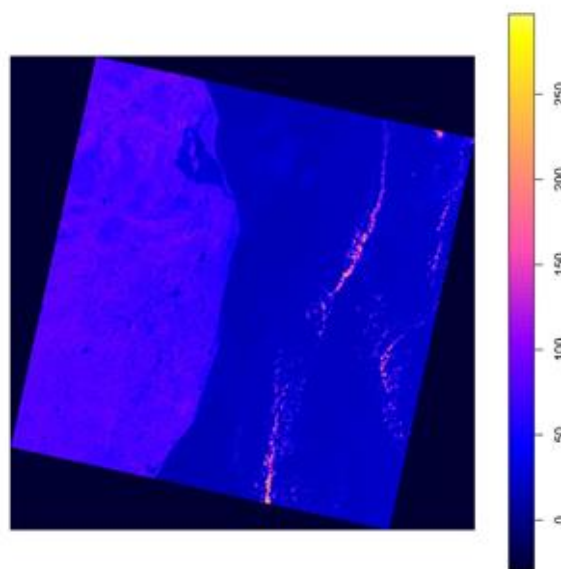


Band 4 transformed to true surface reflectance values

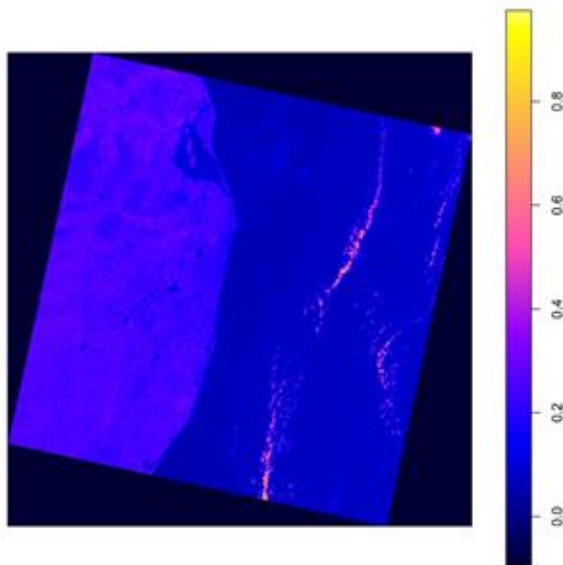
Figure 5: LC08_L1TP_142051_20170410_20170414_01_T1_B4.TIF



Band 5

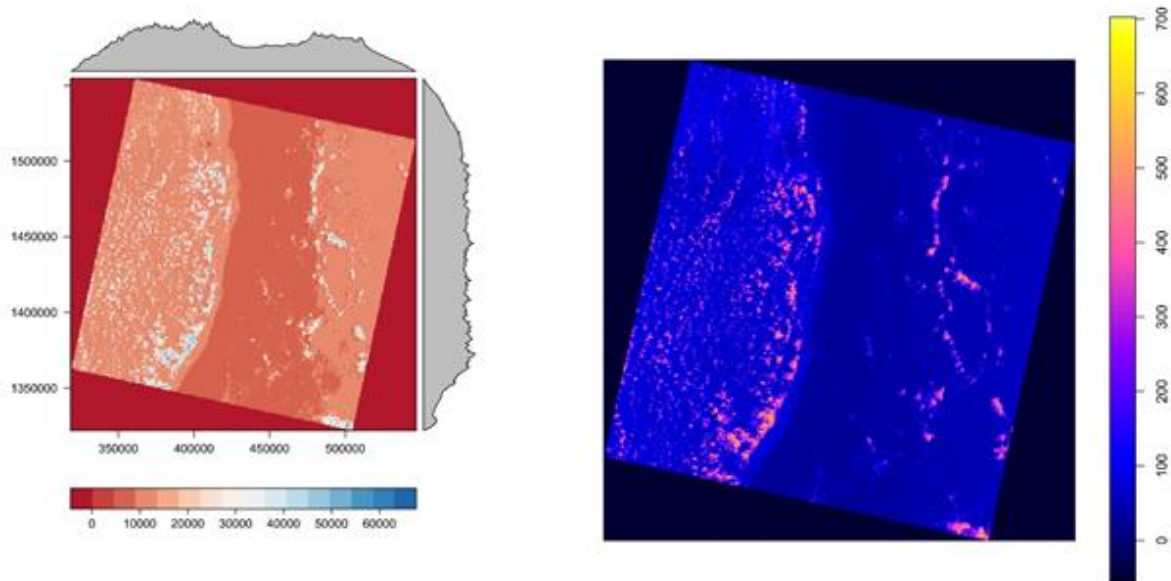


Band 5 transformed to true sensor radiance values



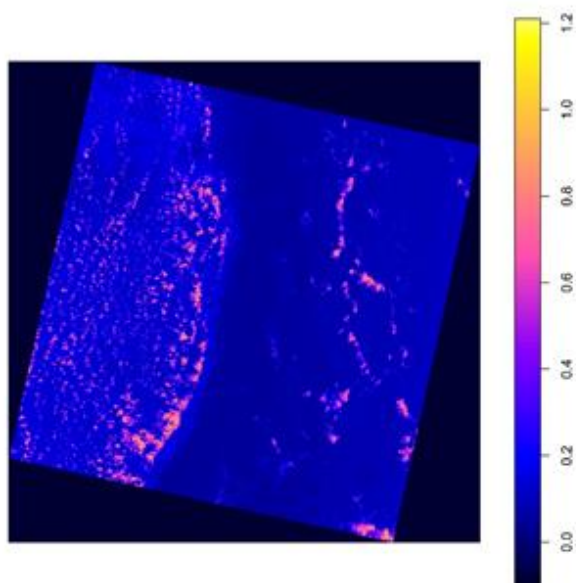
Band 5 transformed to true surface reflectance values

Figure 6: LC08_L1TP_142051_20170410_20170414_01_T1_B5.TIF



Band 3

Band 3 transformed to true sensor radiance values



Band 3 transformed to true surface reflectance values

Figure 7: LC08_L1TP_142051_20170512_20170525_01_T1_B3.TIF

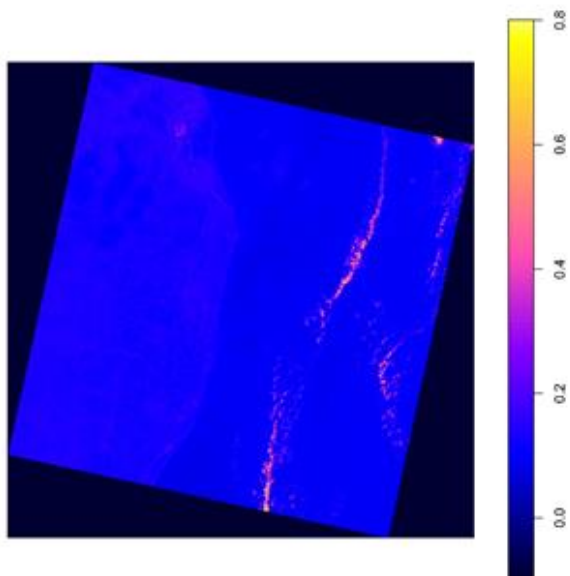
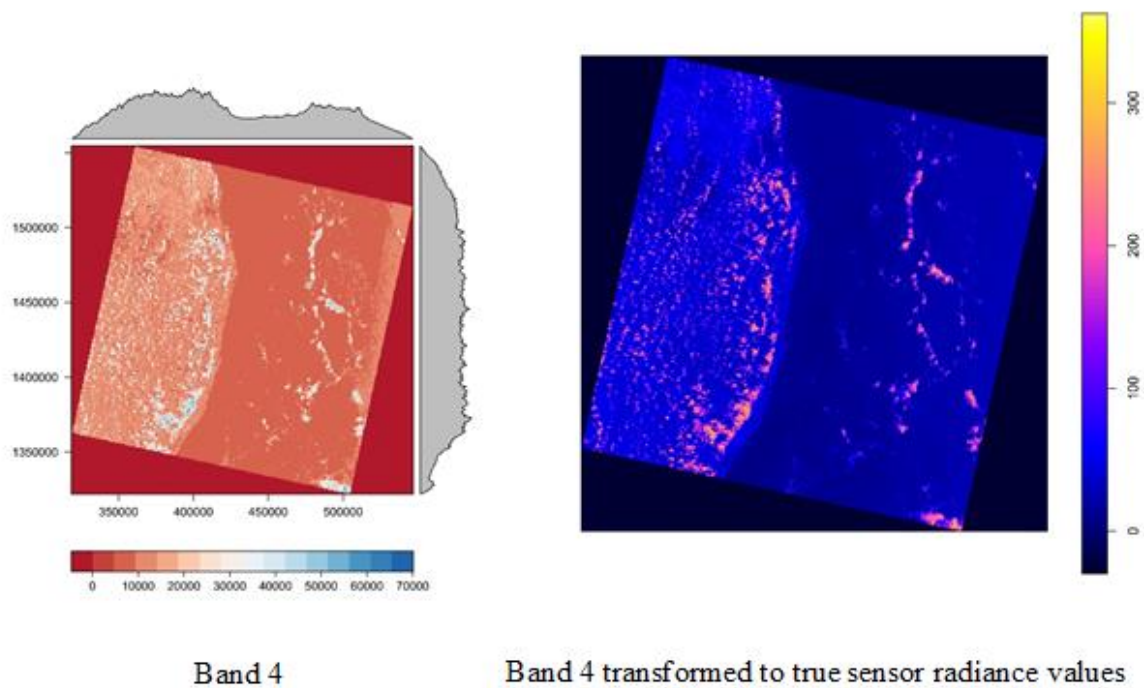
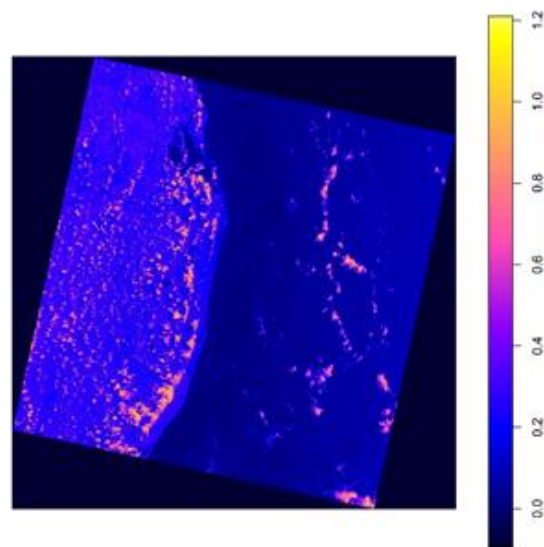
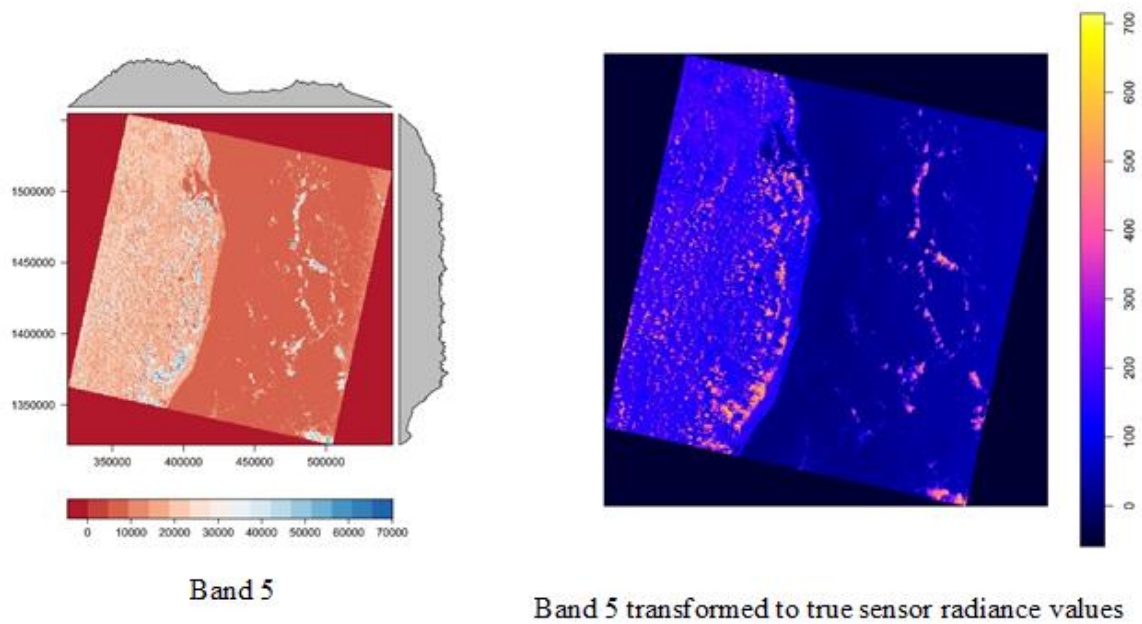


Figure 8: LC08_L1TP_142051_20170512_20170525_01_T1_B4.TIF



Band 5 transformed to true surface reflectance values

Figure 9: LC08_L1TP_142051_20170512_20170525_01_T1_B5.TIF

Table 1: Source code in R for calibration of Radiance and Reflectance from Sensor

```
library(raster)
```

```
library(landsat8)
```

```
library(sp)
```

```
library(rgdal)
```

```
library(rasterVis)
```

```
setwd("/home/rajcheruk/Documents/Pilot-Project/BigData-Landsat8")
```

```
sink(file = "/home/rajcheruk/Documents/Pilot-Project/BigData-Landsat8/Radiance_run.log", append =TRUE,  
type = c("output", "message"), split = FALSE)
```

```
pdf("/home/rajcheruk/Documents/Pilot-Project/BigData-Landsat8/PixelRadiance.pdf", useDingbats = FALSE)
```

```
Image1 <-raster("/home/rajcheruk/Documents/Pilot-Project/BigData-  
Landsat8/LC08_L1TP_142051_20170410_20170414_01_T1_B3.TIF", layer=1, values="TRUE")
```

```
#Image1
```

```
r1grid <- as(Image1, 'SpatialGridDataFrame')
```

```
r1rad <- radconv(r1grid, 1.1805E-02, -59.02273)
```

```
r1ref <- reflconv(r1grid, 2.0000E-05, -0.100000)
```

```
plot(r1rad)
```

```
levelplot(Image1, par.settings=RdBuTheme, contour=FALSE)
```

```
plot(r1ref)
```

```
Image2 <-raster("/home/rajcheruk/Documents/Pilot-Project/BigData-  
Landsat8/LC08_L1TP_142051_20170410_20170414_01_T1_B4.TIF", layer=1, values="TRUE")
```

```
#Image2
```

```
r2grid <- as(Image2, 'SpatialGridDataFrame')
```

```
r2rad <- radconv(r2grid, 9.9543E-03, -49.77129)
```

```
r2ref <- reflconv(r1grid, 2.0000E-05, -0.100000)
```

```
plot(r2rad)
```

```
levelplot(Image2, par.settings=RdBuTheme, contour=FALSE)
```

```
plot(r2ref)
```

```
Image3 <- raster("/home/rajcheruk/Documents/Pilot-Project/BigData-Landsat8/LC08_L1TP_142051_20170410_20170414_01_T1_B5.TIF", layer=1, values="TRUE")
```

```
#Image3
```

```
r3grid <- as(Image3, 'SpatialGridDataFrame')
```

```
r3rad <- radconv(r3grid, 6.0915E-03, -30.45755)
```

```
r3ref <- reflconv(r3grid, 2.0000E-05, -0.100000)
```

```
plot(r3rad)
```

```
levelplot(Image3, par.settings=RdBuTheme, contour=FALSE)
```

```
plot(r3ref)
```

```
Image4 <- raster("/home/rajcheruk/Documents/Pilot-Project/BigData-Landsat8/LC08_L1TP_142051_20170512_20170525_01_T1_B3.TIF", layer=1, values="TRUE")
```

```
#Image4
```

```
r4grid <- as(Image4, 'SpatialGridDataFrame')
```

```
r4rad <- radconv(r4grid, 1.1610E-02, -58.05023)
```

```
r4ref <- reflconv(r4grid, 2.0000E-05, -0.100000)
```

```
plot(r4rad)
```

```
levelplot(Image4, par.settings=RdBuTheme, contour=FALSE)
```

```
plot(r4ref)
```

```
Image5 <- raster("/home/rajcheruk/Documents/Pilot-Project/BigData-Landsat8/LC08_L1TP_142051_20170512_20170525_01_T1_B4.TIF", layer=1, values="TRUE")
```

```
#Image5
```

```
r5grid <- as(Image5, 'SpatialGridDataFrame')
```

```
r5rad <- radconv(r5grid, 5.9911E-03, -29.95571)
```

```
r5ref <- reflconv(r1grid, 2.0000E-05, -0.100000)
```

```
plot(r5rad)
```

```
levelplot(Image5, par.settings=RdBuTheme, contour=FALSE)
```

```
plot(r5ref)
```

```
Image6 <- raster("/home/rajcheruk/Documents/Pilot-Project/BigData-
Landsat8/LC08_L1TP_142051_20170512_20170525_01_T1_B5.TIF", layer=1, values="TRUE")
```

```
#Image6
```

```
r6grid <- as(Image6, 'SpatialGridDataFrame')
```

```
r6rad <- radconv(r6grid, 1.1805E-02, -59.02273)
```

```
r6ref <- reflconv(r6grid, 2.0000E-05, -0.100000)
```

```
plot(r6rad)
```

```
levelplot(Image6, par.settings=RdBuTheme, contour=FALSE)
```

```
plot(r6ref)
```

Table 2: Run-Time log of R script execution

R version 3.4.2 (2017-09-28) -- "Short Summer"

Copyright (C) 2017 The R Foundation for Statistical Computing

Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.

You are welcome to redistribute it under certain conditions.

Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.

Type 'contributors()' for more information and

'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or

'help.start()' for an HTML browser interface to help.

Type 'q()' to quit R.

```
> library(raster)
```

Loading required package: sp

> library(landsat8)

> library(sp)

> library(rgdal)

rgdal: version: 1.2-16, (SVN revision 701)

Geospatial Data Abstraction Library extensions to R successfully loaded

Loaded GDAL runtime: GDAL 2.2.1, released 2017/06/23

Path to GDAL shared files: /usr/share/gdal/2.2

GDAL binary built with GEOS: TRUE

Loaded PROJ.4 runtime: Rel. 4.9.3, 15 August 2016, [PJ_VERSION: 493]

Path to PROJ.4 shared files: (autodetected)

Linking to sp version: 1.2-7

> library(rasterVis)

Loading required package: lattice

Loading required package: latticeExtra

Loading required package: RColorBrewer

>

> setwd("/home/rajcheruk/Documents/Pilot-Project/BigData-Landsat8")

> sink(file = "/home/rajcheruk/Documents/Pilot-Project/BigData-Landsat8/Radiance_run.log", append = TRUE,
 type = c("output", "message"), split = FALSE)

>

> pdf("/home/rajcheruk/Documents/Pilot-Project/BigData-Landsat8/PixelRadiance.pdf", useDingbats =
 FALSE)

>

> Image1 <- raster("/home/rajcheruk/Documents/Pilot-Project/BigData-Landsat8/LC08_L1TP_142051_20170410_20170414_01_T1_B3.TIF", layer=1, values="TRUE")

> #Image1

> r1grid <- as(Image1, 'SpatialGridDataFrame')

> r1rad <- radconv(r1grid, 1.1805E-02, -59.02273)

> r1ref <- reflconv(r1grid, 2.0000E-05, -0.100000)

```
> plot(r1rad)

> levelplot(Image1, par.settings=RdBuTheme, contour=FALSE)

> plot(r1ref)

>

> Image2 <- raster("/home/rajcheruk/Documents/Pilot-Project/BigData-
Landsat8/LC08_L1TP_142051_20170410_20170414_01_T1_B4.TIF", layer=1, values="TRUE")

> #Image2

> r2grid <- as(Image2, 'SpatialGridDataFrame')

> r2rad <- radconv(r2grid, 9.9543E-03, -49.77129)

> r2ref <- reflconv(r1grid, 2.0000E-05, -0.100000)

> plot(r2rad)

> levelplot(Image2, par.settings=RdBuTheme, contour=FALSE)

> plot(r2ref)

>

> Image3 <- raster("/home/rajcheruk/Documents/Pilot-Project/BigData-
Landsat8/LC08_L1TP_142051_20170410_20170414_01_T1_B5.TIF", layer=1, values="TRUE")

> #Image3

> r3grid <- as(Image3, 'SpatialGridDataFrame')

> r3rad <- radconv(r3grid, 6.0915E-03, -30.45755)

> r3ref <- reflconv(r3grid, 2.0000E-05, -0.100000)

> plot(r3rad)

> levelplot(Image3, par.settings=RdBuTheme, contour=FALSE)

> plot(r3ref)

>

> Image4 <- raster("/home/rajcheruk/Documents/Pilot-Project/BigData-
Landsat8/LC08_L1TP_142051_20170512_20170525_01_T1_B3.TIF", layer=1, values="TRUE")

> #Image4

> r4grid <- as(Image4, 'SpatialGridDataFrame')

> r4rad <- radconv(r4grid, 1.1610E-02, -58.05023)

> r4ref <- reflconv(r4grid, 2.0000E-05, -0.100000)
```

```

> plot(r4rad)

> levelplot(Image4, par.settings=RdBuTheme, contour=FALSE)

> plot(r4ref)

>

> Image5 <- raster("/home/rajcheruk/Documents/Pilot-Project/BigData-
Landsat8/LC08_L1TP_142051_20170512_20170525_01_T1_B4.TIF", layer=1, values="TRUE")

> #Image5

> r5grid <- as(Image5, 'SpatialGridDataFrame')

> r5rad <- radconv(r5grid, 5.9911E-03, -29.95571)

> r5ref <- reflconv(r1grid, 2.0000E-05,-0.100000)

> plot(r5rad)

> levelplot(Image5, par.settings=RdBuTheme, contour=FALSE)

> plot(r5ref)

>

> Image6 <- raster("/home/rajcheruk/Documents/Pilot-Project/BigData-
Landsat8/LC08_L1TP_142051_20170512_20170525_01_T1_B5.TIF", layer=1, values="TRUE")

> #Image6

> r6grid <- as(Image6, 'SpatialGridDataFrame')

> r6rad <- radconv(r6grid, 1.1805E-02, -59.02273)

> r6ref <- reflconv(r6grid, 2.0000E-05,-0.100000)

> plot(r6rad)

> levelplot(Image6, par.settings=RdBuTheme, contour=FALSE)

> plot(r6ref)

> proc.time()

user system elapsed
282.865 62.082 662.973

```

References

1. Signal Theory Methods in Multispectral Remote Sensing, by David A. Landgrebe. Wiley Series in Remote Sensing, Wiley Interscience, 2003.

2. Innovations in Remote Sensing and Photogrammetry, by Simon Jones and Karin Reinke. Springer publications, 2009.
3. Parameter tradeoffs for imaging spectroscopy systems, by J. P. Kerekes and D. Landgrebe, IEEE Transactions on Geoscience and Remote Sensing, Vol. 29, No. 1, pp 57-65, January 1991.
4. Radiometric and Topographic correction of satellite imagery, by Sarah Goslee. Package "Landsat" for R software, version 1.0.8, 2012.
5. Learning R for Geospatial Analysis, by Michael Dorman, Packt Publishing, 2014.