Matching sensor overlap

Jose Eduardo Meireles

The problem

Spectra may have overlap regions between two sensors and the reflectance or radiance values can have abrupt ‘jumps’ in those regions. In those cases, the regions between sensors need to be matched (spliced).

Unspliced spectra collected with a 3-sensor instrument (e.g. an SVC) may look like this:

```r
# Path to raw (unmatched) spectra
path_raw = system.file("extdata/svc_raw_and_overlap_matched_serbin/SVC_Files",
                       package = "spectrolab")

# Read spectra as reflectance and radiance
reflect_raw  = read_spectra(path   = path_raw,
                            format = "SIG",
                            type   = "target_reflectance")
```

```
## Duplicated band values are not allowed!
## Bands updated as follows:
##  band_position original_value updated_value
##            514          975.6      975.6011
##            524         1013.9     1013.9011
```

```
## c(514, 524)c(975.6, 1013.9)c(975.6011, 1013.9011)
```

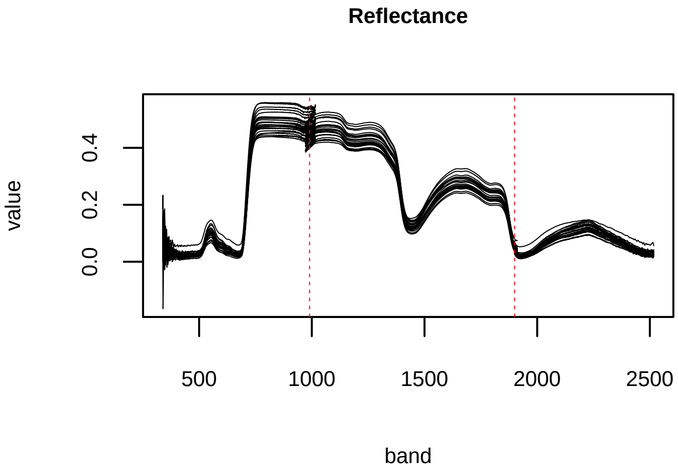
```r
radiance_raw = read_spectra(path   = path_raw,
                            format = "SIG",
                            type   = "target_radiance")
```

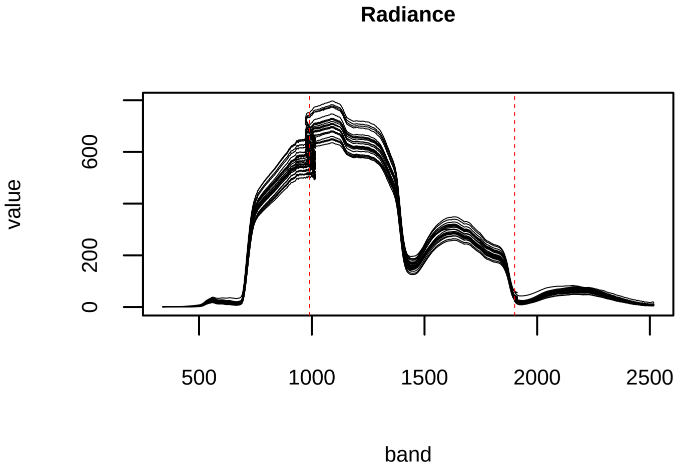
```
## Duplicated band values are not allowed!
## Bands updated as follows:
##  band_position original_value updated_value
##            514          975.6      975.6011
##            524         1013.9     1013.9011
```

```
## c(514, 524)c(975.6, 1013.9)c(975.6011, 1013.9011)
```

```r
# Sensor overlaps marked with vertical dashed lines
lwd = 0.5
cex = 0.7
par(mfrow = c(2, 1))
```

```
plot(reflect_raw, main = "Reflectance",
     lwd = lwd, cex.main = cex, cex.lab = cex, cex.axis = cex)
abline(v = c(990, 1900), col = "red", lty = 2, lwd = lwd)

plot(radiance_raw, main = "Radiance",
     lwd = lwd, cex.main = cex, cex.lab = cex, cex.axis = cex)
abline(v = c(990, 1900), col = "red", lty = 2, lwd = lwd)
```

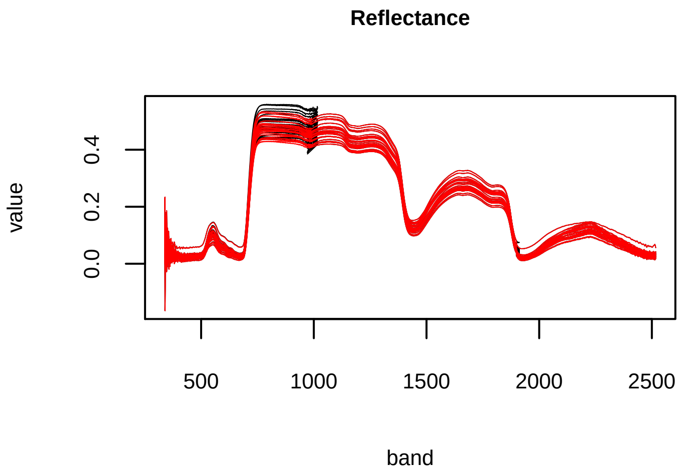


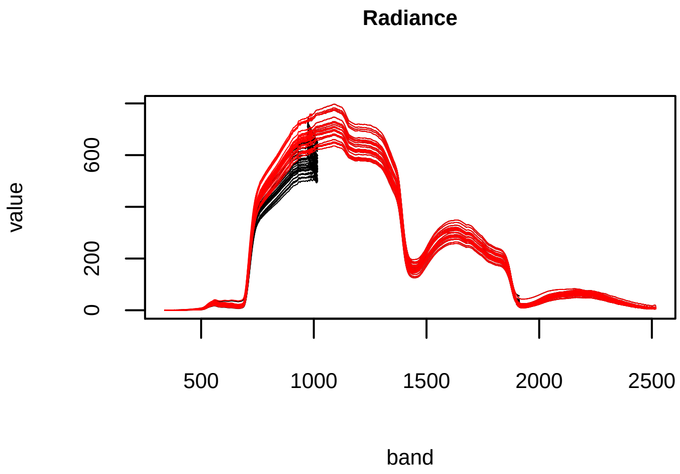


The solution

Use the function `match_sensors` to splice the sensor overlap regions as shown below. You must pass the boundary between sensors using the `splice_at` argument. It is **critical** that you get those bands right and every instrument (even from the same vendor) is different. You can use `plot_interactive` zoom into a particular spectral region and decide what the `splice_at` values should be.

```
# Boundaries between sensors
splice_bands = c(990, 1900)

# Match the reflectance and radiance data
reflect_matched = match_sensors(x               = reflect_raw,
                                splice_at       = splice_bands,
                                interpolate_wvl = c(5, 1))

radiance_matched = match_sensors(x               = radiance_raw,
                                 splice_at       = splice_bands,
                                 interpolate_wvl = c(5, 1))

lwd = 0.5
cex = 0.7
par(mfrow = c(2, 1))


plot(reflect_raw, main = "Reflectance",
     lwd = lwd, cex.main = cex, cex.lab = cex, cex.axis = cex)

plot(reflect_matched, col = "red", add = TRUE,
     lwd = lwd, cex.main = cex, cex.lab = cex, cex.axis = cex)


plot(radiance_raw, main = "Radiance",
     lwd = lwd, cex.main = cex, cex.lab = cex, cex.axis = cex)

plot(radiance_matched, col = "red", add = TRUE,
     lwd = lwd, cex.main = cex, cex.lab = cex, cex.axis = cex)
```

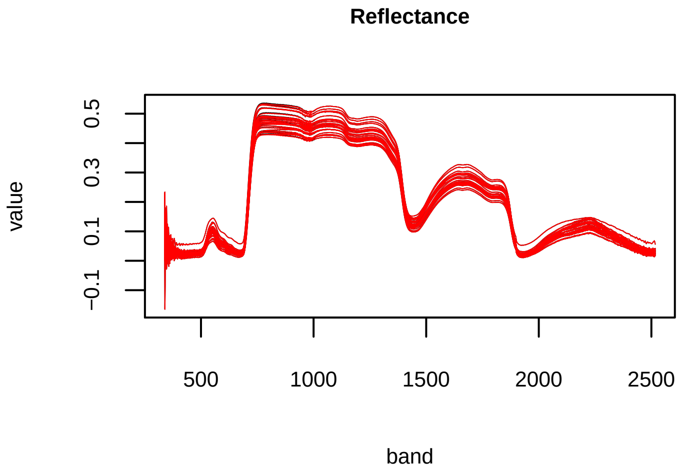


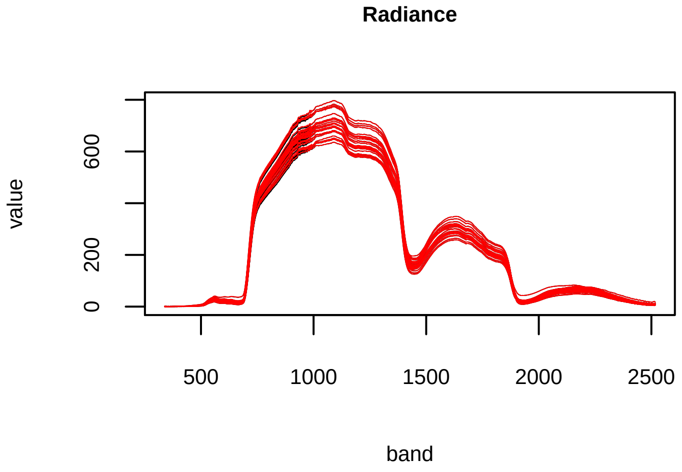


And we can check the results from spectrolab's `match_sensors` against SVC's proprietary matching algorithm.

```
path_moc = system.file("extdata/svc_raw_and_overlap_matched_serbin/SVC_Files_MOC",
                       package = "spectrolab")
reflect_moc   = read_spectra(path   = path_moc,
                             format = "SIG",
                             type   = "target_reflectance")

radiance_moc = read_spectra(path   = path_moc,
                             format = "SIG",
                             type   = "target_radiance")



lwd = 0.5
cex = 0.7
par(mfrow = c(2, 1))
```

```
plot(reflect_moc, main = "Reflectance", col = "black",
     lwd = lwd, cex.main = cex, cex.lab = cex, cex.axis = cex)

plot(reflect_matched, col = "red", add = TRUE,
     lwd = lwd, cex.main = cex, cex.lab = cex, cex.axis = cex)


plot(radiance_moc, main = "Radiance", col = "black",
     lwd = lwd, cex.main = cex, cex.lab = cex, cex.axis = cex)

plot(radiance_matched, col = "red", add = TRUE,
     lwd = lwd, cex.main = cex, cex.lab = cex, cex.axis = cex)
```


Reflectance
value
band


Radiance
value
band