

Water Body Extent Thought Experiment

Rebecca Stubbs

Prompt

Describe how you would find and track the extent of a water body within a 1 year period. Explain the data, method, and platform you would use.

Answer

There are undoubtedly many ways to solve this problem— but to keep things tractable, I’ve focused on a solution that would hopefully minimize both the effort and budget required. I would most likely use remote sensing to find and monitor a waterbody’s extent. This strategy would work best for a larger water body in an area with infrequent cloud-cover.

Landsat-8 looks like a good candidate for a data source— it has a 30m resolution, takes images every 16 days, and is also freely available. The scenes can be downloaded via Amazon Web Services with reasonable efficiency. Landsat is particularly attractive as a data source due to the presence of a corresponding data product for each scene— the quality assessment band— that has bit-encoded information about whether each pixel is clear, cloud-covered, cloud-shadowed, terrain occluded, snow, or water.

Step 1: Where is the Water in the Scene?

The first step is determining which pixels are water in the downloaded scenes. There are a few options, escalating in sophistication, to determine where water is within the imaged scene we are interested in:

- Use the quality assessment band provided by Landsat. It seems that the classification algorithm that is used to generate these layers can work better for some areas than others— however, it could serve as a first pass solution. There is a free ArcGIS toolbox¹ that can be used to extract this information, and a function in an R package (RStools::classifyQA) that could accomplish this task as well.
- Calculate an index, such as the NDWI (Normalized Difference Water Index) using the spectral image bands to determine water presence/absence for each pixel, and use a threshold or cutoff point to determine pixels that are likely to be water.
- Use a combination of different methods, including masks and classification algorithms, as proposed in this paper from 2016: An effective modified water extraction method for Landsat-8 OLI imagery of mountainous plateau regions. A method like the one described in this paper (requiring multiple indices and algorithms to be applied) could be the most accurate, but also might be the most time-consuming and would require the most up-front effort.
- If year-to-year changes are the focus of the study, finding clear images representative of each year taken in the same calendar month might suffice; if seasonality (but not a finer resolution) is important, it could be possible to pick the most cloudless images taken within each season.
- If a finer resolution of measurement in time is required, the course of action would most likely be to process all images taken by Landsat 8 in the time period of interest, and then exclude cloud-covered pixels to generate a time series with missing observations due to cloud cover.

¹<https://github.com/USGS-EROS/landsat-qa-arcgis-toolbox>

For areas that freeze during the winter, or become covered in snow and ice, this would become very challenging—a different classification method would almost certainly be appropriate, and might require a different strategy altogether.

Step 2: Focusing on the area of interest

The rough extent of many water bodies can be defined using freely-available hydrography vector layers. After finding the geometry (for example, a shapefile from a county’s GIS website) for the water body of interest, its polygon shape (buffered to include peripheral terrain) could serve as a template to clip the raster images to an area of focus.

If an index, such as NDWI, is being calculated, or the quality assessment pixel layer is being used, cropping the raster images early in the data processing pipeline would save processing time and space for resulting saved files. However, if you are using an image classification algorithm, it might be worth waiting on this step, depending on how much the surrounding terrain influences the designation for each pixel.

Step 3: Compare algorithm outputs for each time period to track changes in extent

Once a raster layer of binary ‘water’ and ‘not-water’ pixels for each time period of interest has been generated, there are a variety of ways the extent of the water body could be monitored:

- Generating polygon or line vector information on the boundary between land and water— using each pixel centroid as a vertex, you could create the vector-version of the water body’s boundaries for each time period.
- Track, for each pixel, what times during the year each area was classified as “water” or “not water”. This could get even more sophisticated if a classification process was run to recognize “damp soil”, which is information that can potentially be gleaned from short-wave infrared bands.

In terms of platform, I could foresee myself using either R or ArcGIS for this task— it depends on the extent of automation I am looking for. I would prefer to code in R if a high level of automation makes sense (for example, if many months worth of imagery data needed to be processed), but ArcGIS also has a wonderful suite of point-and-click tools that might get the job done quickly if I were only processing a handful of images.

A few other ideas I had during this thought experiment that seemed worth mentioning:

- It looks like the USGS is on the brink of generating a data product using Landsat-8 called “Dynamic Surface Water Extent”², which could neatly solve this problem (however, since it doesn’t exist yet, there’s no way to know)
- For a water body that is very small, that needs to be measured at very specific times, or that is covered in clouds most of the time, commercial or UAV imagery might be worth considering. Project budget would undoubtedly be a factor before considering the purchase of imagery or data products.
- As a sanity-check for how well this method is working, I would consider overlaying the water boundary layers generated from this analysis with images from a publicly-available, very high resolution source (like Google Maps). It is possible to determine when images on Google Earth were taken, and the closest, cloud-free corresponding Landsat image time (and the resulting water/no-water layer) could be used for a comparison.
- For many repeated measurements over the course of a year, using SAR (synthetic aperture radar), rather than the visual spectrum, might be a good choice; some researchers out of Canada have developed methods³ of water classification in advance of the launch of Canada’s Radarsat constellation (supposedly flying sometime in 2018). The Sentinel satellites (flying now) currently contain SAR observations.

²https://remotesensing.usgs.gov/ecv/SWE_overview.php

³<http://www.mdpi.com/2072-4292/8/4/285/pdf>

- Depending on available time, resources, and how often the process is needed, another strategy could be to import the image into a platform like ArcGIS, and have a human trace images, building a vector data set representing the water body extent— after all, the human mind is an awesome classification tool.
- In conjunction with the right (high-resolution, timely) bathymetry data, a water gauge of some kind could measure water volume for a year. After taking 1 set of lake boundaries (using a classification method above, for instance) where the lake was at a certain depth could potentially be used to extrapolate where the water line would be at different water levels. This solution might work for lakes that become ice-covered, but still are subject to changes in water level beneath the frozen surface.

Playing around with a test case

Out of curiosity, I tried out the NDWI method for water classification for an area in the mountains just outside Boulder, CO, focusing on an area to the east of Rocky Mountain National Park— Lake Granby.

First, I found a relevant scene using <https://earthexplorer.usgs.gov/>, and then found the place on amazon web services where Landsat scenes for this area were stored: <https://landsatonaaws.com/L8/034/032>. I chose a scene from August of 2017, due to its low cloud-cover content (only 9%).

NB: This document is as a .pdf rendering of an R Markdown, which allows for the integration of documentation, code, and outputs. What follows is all of the code required to process the data, and generate the outputs for this exercise.

```

# Waterbody extent sandbox; RStubbs 02/2018
# Test case Lake Granby, CO
# Input: Landsat 8 images, Grand County Lakes .shp
# Output: Classified water/not-water raster layer

rm(list=ls())
library(raster)
library(MapSuite) # My package, has some functions to turn a raster into a data.table

# For the NDWI, we actually only care about the green and nir band,
# which are B3 and B5, respectively, but to get a sense of the scene, we can load in red,
# and blue as well.

img_prefix<-paste0("/Users/stubbsrw/Documents/git_code/stubbs_repo/",
                   "fe_problems/water_body_extent/landsat_8/",
                   "LC08_L1TP_034032_20170827_20170914_01_T1/",
                   "LC08_L1TP_034032_20170827_20170914_01_T1_")

# Load in bands
red<-raster::raster(paste0(img_prefix,"B4",".TIF"))
green<-raster::raster(paste0(img_prefix,"B3",".TIF"))
blue<-raster::raster(paste0(img_prefix,"B2",".TIF"))
nir<-raster::raster(paste0(img_prefix,"B5",".TIF"))

# Make a raster-brick, crop to roughly the Grandby area
granby_visual_spectrum<-crop(brick(red,green,blue),
                             extent(419996, 437500, 4441045, 4456990))

# Plot April 2017 Scene
plotRGB(granby_visual_spectrum) # use native plot function for R

```



This scene contains some cloud cover, but is fairly clear overall where we need it to be (over Lake Granby). Let's see how the NDWI, by McFeeters 1996, as seen here on the USGS⁴ website does— positive values indicate surface water for this index. A paper published in 2015⁵ has a variety of different calculation methods.

Calculating the NDWI index based on the Green and NIR bands

```
ndwi<-((green-nir)/(green+nir)) # McFeeters NDWI, 1996; positive values=Water
# Let's zoom in to the area around Lake Granby:
granby_ndwi<-crop(ndwi,extent(419996, 437500, 4441045, 4456990))
```

There is a handy function in my R package that converts from a raster object to a data.table with x/y coordinates of the points, which should make mapping this on a more sensible scale (for example, with all values above 0 being blue), more straightforward:

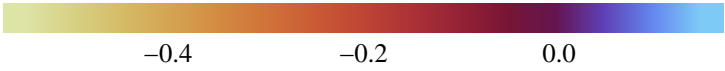
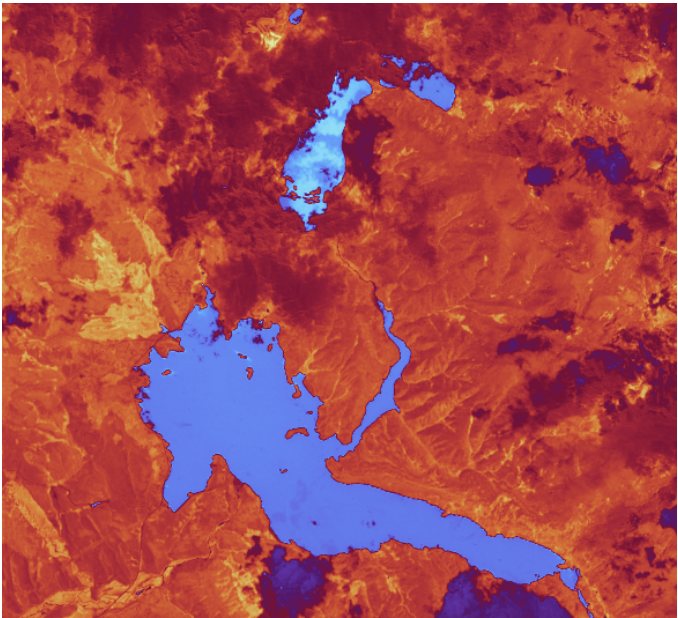
```
raster_table<-MapSuite::PixelsToTables(rast=granby_ndwi,var_name='ndwi')
MapSuite::RasterMap(raster_table,id="id",xcol="x",ycol="y",variable="ndwi",
  map_colors=rev(wpal("diverging_orange_purple_blue")),
  map_diverging_centrvalue = 0,
  map_title = "Normalized Difference Water Index",
  map_subtitle = "Lake Granby Area, CO")
```

⁴https://deltas.usgs.gov/fm/data/data_ndwi.aspx

⁵<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4507615/#sec2-sensors-15-13763title>

Normalized Difference Water Index

Lake Granby Area, CO



It would probably depend on the water body as to whether this method would be considered a “success” or not. At this point, comparing pixels within the lake’s polygon boundary (or a buffered distance around the lake’s polygon boundary) for scenes spanning a year would describe waterbody extent changes over time. Creating a binary classification (water, or not) might further clarify how well this process worked.

```
# Let's create a binary, with 0 as non-water, and 1 as water:
raster_table[,factor_water:=ifelse(ndwi>=0,"Surface Water","Not Water")]

# Load in lakes of Grand County, select Lake Granby from SPDF
gcl<-MapSuite::ShptoSPDF(shp_dir=paste0("/Users/stubbsrw/Documents/git_code/",
                                         "stubbs_repo/fe_problems/water_body_extent/grand_county_lakes/"),
                        shp_layer="lakes" , id_field="Acres")

## OGR data source with driver: ESRI Shapefile
## Source: "/Users/stubbsrw/Documents/git_code/stubbs_repo/fe_problems/water_body_extent/grand_county_l
## with 152 features
## It has 5 fields
## [1] "Note: There is a new column in the @data slot called polygon_order. Remember that the attributes

# Select Lake Granby
granby <- gcl[gcl@data$polygon_order==142,]
# Re-project to same info as raster
granby <- spTransform(granby, CRS(proj4string(granby_ndwi)))

# Make map with red Granby outline
MapSuite::RasterMap(raster_table,id="id",xcol="x",ycol="y",variable="factor_water",
                    map_colors=rev(wpal("diverging_orange_purple_blue")),
                    map_diverging_centervalue = 0,
                    map_title = "Water/Not Water Classification achieved through NDWI",
                    map_subtitle = "Lake Granby Area, CO",
                    outline = granby, outline_color="red")
```

Water/Not Water Classification achieved through NDWI

Lake Granby Area, CO



■ Not Water ■ Surface Water

Shown another way, here is the surface water detected by this method, this time classified as “water” or “not water”, with the polygon boundary of Lake Granby (as procured via Grand County’s Digital Data Sets Portal)⁶ overlayed in red.

⁶<http://co.grand.co.us/170/Digital-Data-Sets>