

Google Earth Engine: Ένα εργαλείο ανοικτού κώδικα για την ανίχνευση και παρακολούθηση του φυσικού περιβάλλοντος

**Τσιούκας Β., Guerra, F. ,
Balletti, C., Vernier, P.**

Τμήμα Αγρονόμων και Τοπογραφών Μηχανικών, ΑΠΘ
Università IUAV di Venezia, Laboratorio di Cartografia e GIS

Η χαρτογραφία σε έναν κόσμο που αλλάζει

- Παρακολούθηση μεταβολών μέσω δορυφορικών εικόνων
- Σύγχρονα εργαλεία (λογισμικά)
- Σύγχρονα αλλά και παλιά-ιστορικά δεδομένα που βρίσκονται online (δορυφορικές εικόνες)
- Σύγχρονες τεχνικές-αλγόριθμοι (αυτοματοποιημένοι)
- Πλατφόρμες (Windows-Unix-MacOs, online-offline)

Google Earth Engine

- About: *Earth Engine is a platform for planetary-scale environmental data analysis*
- Έχει πρόσβαση οποιοσδήποτε αρκεί να έχει λογαριασμό στην google (xxx@gmail.com) και να αποκτήσεις κωδικούς στο σύστημα ανάπτυξης κώδικα

Περιβάλλον ανάπτυξης εφαρμογών ανοικτού κώδικα

- Το Google Earth Engine αποτελείται από ένα περιβάλλον προγραμματισμού με μεγάλο αριθμό συναρτήσεων βιβλιοθηκών επεξεργασίας και γεοδομένων όπου κάθε χρήστης μπορεί να εμπλουτίσει με δικές του συναρτήσεις-υποπρογράμματα
- JavaScript ή python
- Λειτουργεί σε διαδικτυακό επίπεδο με **όχι** απαραίτητα υψηλές απαιτήσεις ισχυρού επεξεργαστή και μεγάλης μνήμης (λόγω cloud processing and sharing)

Τίποτα δεν διατηρείται τοπικά!

Κατάλογος δεδομένων

- Πολυφασματικές εικόνες Landsat (4,5,6,7 και 8) και Sentinel 2
- Radar (Sentinel 1)
- Ψηφιακά μοντέλα εδάφους (SRTM ανάλυσης grid size 90m ή lidar πχ Ολλανδίας grid size 0.5m)
- Υπερφασματικές εικόνες Hyperion (EO-1)
- Ατμοσφαιρικά (μόλυνση) και κλιματολογικά δεδομένα
- Θεματικοί χάρτες (πχ. Επικινδυνότητας)

Παράδειγμα Λίμνη Κορώνειας

- 1987 (e6)
- 1991 (e5)
- 2003 (e4)
- 2006 (e3)
- 2010 (e1)
- 2011 (e0)



Παράδειγμα της λίμνης Κορώνειας

- Αναζήτηση κατάλληλων διαχρονικών εικόνων από τον κατάλογο ([μέσω EarthExplorer](#))
- Εφαρμογή κατάλληλου δείκτη (κανονικοποιημένος δείκτης νερού NDWI)
- Δημιουργία μάσκας
- Εκτίμηση επιφάνειας σε μέγεθος ανάλυσης ψηφίδας (30x30m)

$$NDWI = \frac{(X_{nir} - X_{swir})}{(X_{nir} + X_{swir})}$$

Παράδειγμα κώδικα

```
1 var e0=ee.Image('LT5_L1T_TOA/LT51840322011266MOR00'); // ανοιγμα εικόνας
2 Map.addLayer(e0,{'bands': ['B4', 'B3', 'B2']},'2010'); // προσθήκη σε google maps
3 var ndwi = image.normalizedDifference(['B3', 'B5']); // υπολογισμός δείκτη νερού
4 var ndwiViz = {min: 0.3, max: 1, palette: ['00FFFF', '0000FF']}; // εμφάνιση τιμών δείκτη
5 var ndwiMasked = ndwi.updateMask(ndwi.gte(0.4)); // επιλογή ψηφίδων για δείκτη νερού > 0.4
6 var hotspots = ndwi.gt(0.3); // ορισμός ομοιγενών περιοχών με δείκτη νερού > 0.4
7 hotspots = hotspots.updateMask(hotspots); // δημιουργία μάσκας
8 var patchsize = hotspots.connectedPixelCount(256, false); // δημιουργία segment
9 Map.addLayer(patchsize, {}, '1987 lake'); // προσθήκη layer που υπολογίστηκε ως λίμνη
10 var patchid = hotspots.connectedComponents(ee.Kernel.plus(1), 256);
11 //print(e6)
12 var roi = ee.Geometry.Point([23.15,40.68]).buffer(6000); // επιλογή περιοχής γύρω από το κέντρο της λίμνης
13 var lake=patchid.clip(roi) // υπολογισμός ψηφίδων που ανήκουν στην τάξη του νερού
14 var test=lake.reduceRegion(ee.Reducer.sum())
15 var sq=test.get('nd')
16 print(sq) //εμφάνιση του συνολικού αριθμού ψηφίδων
17 var date = ee.Date(image.get('system:time_start'));
18 print('Timestamp: ', date); // εμφάνιση ημερομηνίας λήψης της εικόνας
19
20 Map.setCenter(23.15,40.68,12,9)
```


Παραγωγή video

- Χρήση όλου του αρχείου των εικόνων Landsat (από το 1991 μέχρι το 2011)
- Εικόνες χωρίς σύννεφα (κάλυψη <10%)
- 3 γραμμές κώδικα



```
1 * (**** start of imports. if esites, may not auto-convert in the playground. ****)
2
3 var collection = ee.imagecollection('LANDSAT/LT1_L1T_TOA')
4 // San Francisco Bay.
5 .filter(ee.Filter.eq('IRS_PATH', 113))
6 .filter(ee.Filter.eq('IRS_BAND', 12))
7 // Filter cloudy scenes.
8 .filter(ee.Filter.lt('CLOUD_COVER', 20))
9 // Get 20 years of imagery.
10 .filterDate('1991-01-01', '2011-12-31')
11 // Need to have 3-band imagery for the video.
12 .select(['B4', 'B3', 'B2'])
13 // Need to make the data 8-bit.
14 .map(function(image) {
15   return image.multiply(S12).uint8();
16 });
17
18 // Define an area to export.
19 var polygon = ee.Geometry.Rectangle([21.087761616175, 68.71313131313131, 21.12561879881986, 68.45172237175186]);
20
21 // Export (change dimensions or scale for higher quality).
22 Export.video.toBlob({ collection: collection, description: 'aVideoExample', dimensions: 720, framesPerSecond: 13, region:
```

Συμπεράσματα

- Εργαλείο ανοικτού κώδικα ελεύθερο **προς όλους** για ανάλυση εικόνων και αλλά άλλων γεωγραφικών δεδομένων
- Ελάχιστες απαιτήσεις σε υπολογιστική δύναμη και χωρητικότητα δίσκων
- Ελάχιστες απαιτήσεις γνώσεων γλώσσας προγραμματισμού
- Ανεξάρτητο πλατφόρμας
- Γρήγορη επεξεργασία
- Μπορεί να αποτελέσει ένα άριστο εκπαιδευτικό εργαλείο για την εκμάθηση Ψηφιακής Επεξεργασίας Τηλεπισκοπικών Εικόνων (ΑΠΘ-TATM ακ. έτος 2016-17)
- Απεριόριστες δυνατότητες μέσω εκατοντάδων εντολών ανάλυσης ψηφιακών εικόνων και άλλων δεδομένων

Ευχαριστώ για την προσοχή σας!