Pengenalan Google Earth Engine (GEE)

Safran Yusri

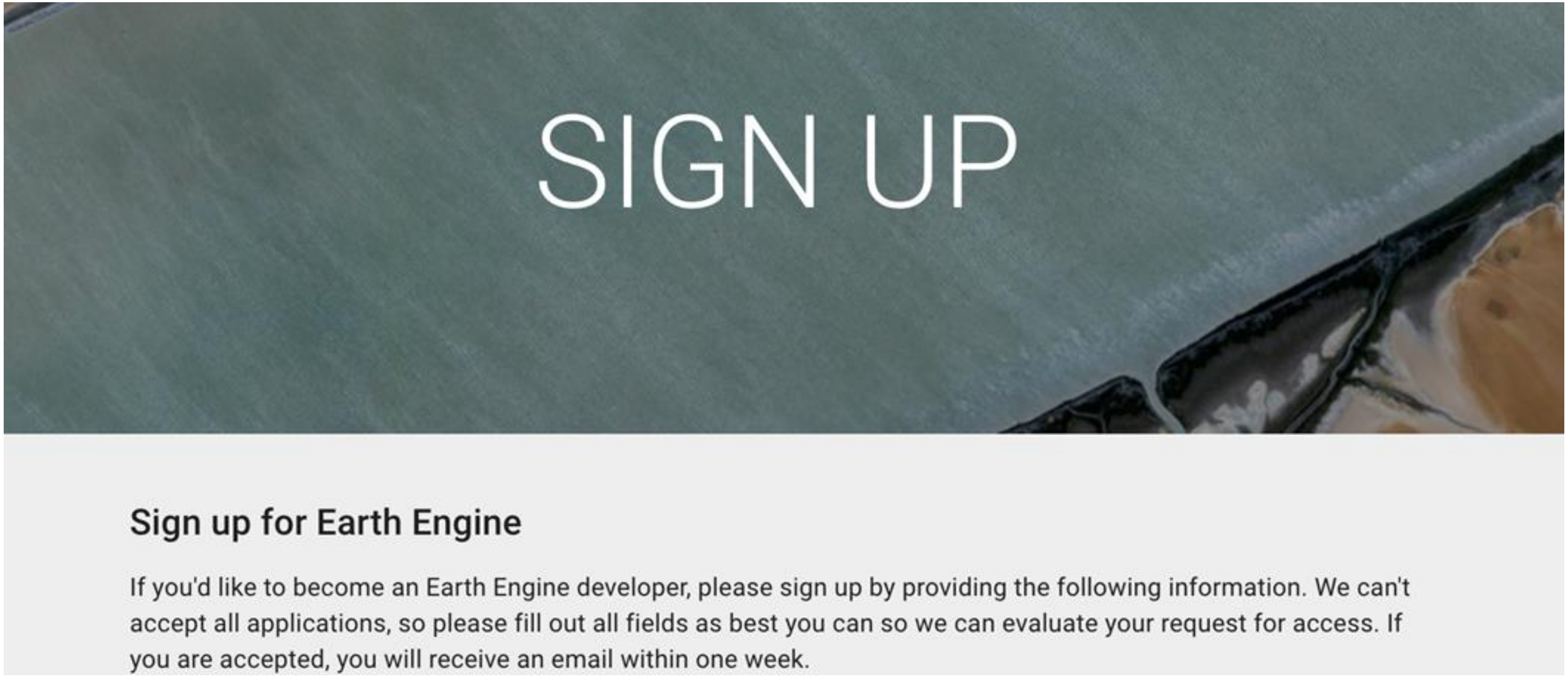
TERANGI

terangi.or.id

Daftar Isi

Sebelum mulai

› Sign up for Earth Engine earthengine.google.com/signup

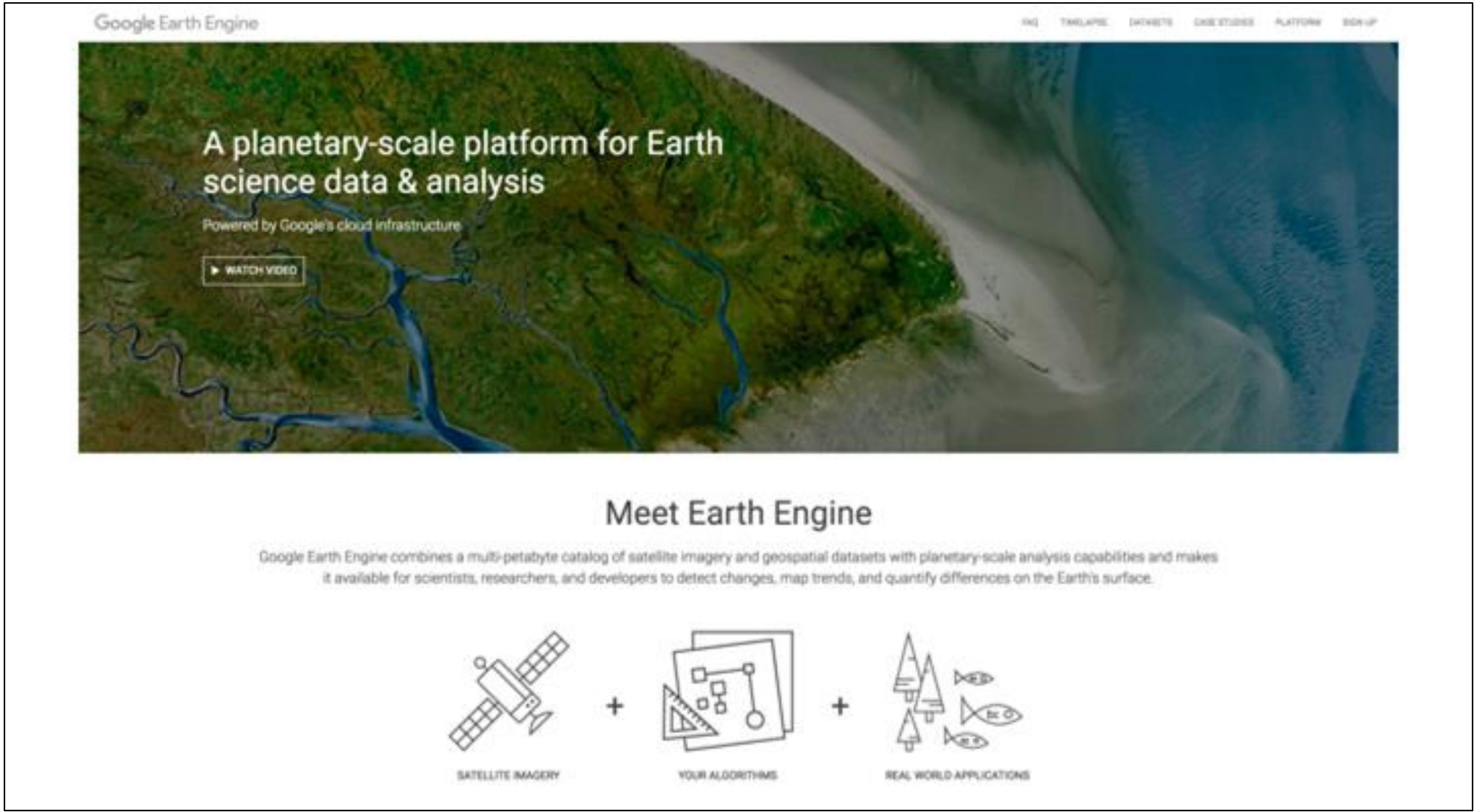


Pengenalan GEE

Safran Yusri dan Fakhrurrozi

Earth Engine adalah **wahana komputasi awan** milik Google's untuk memudahkan **analisis skala petabit** dari **citra satelit** dan **data geospasial** lainnya

Google Earth Engine official launch in COP-16, 2010



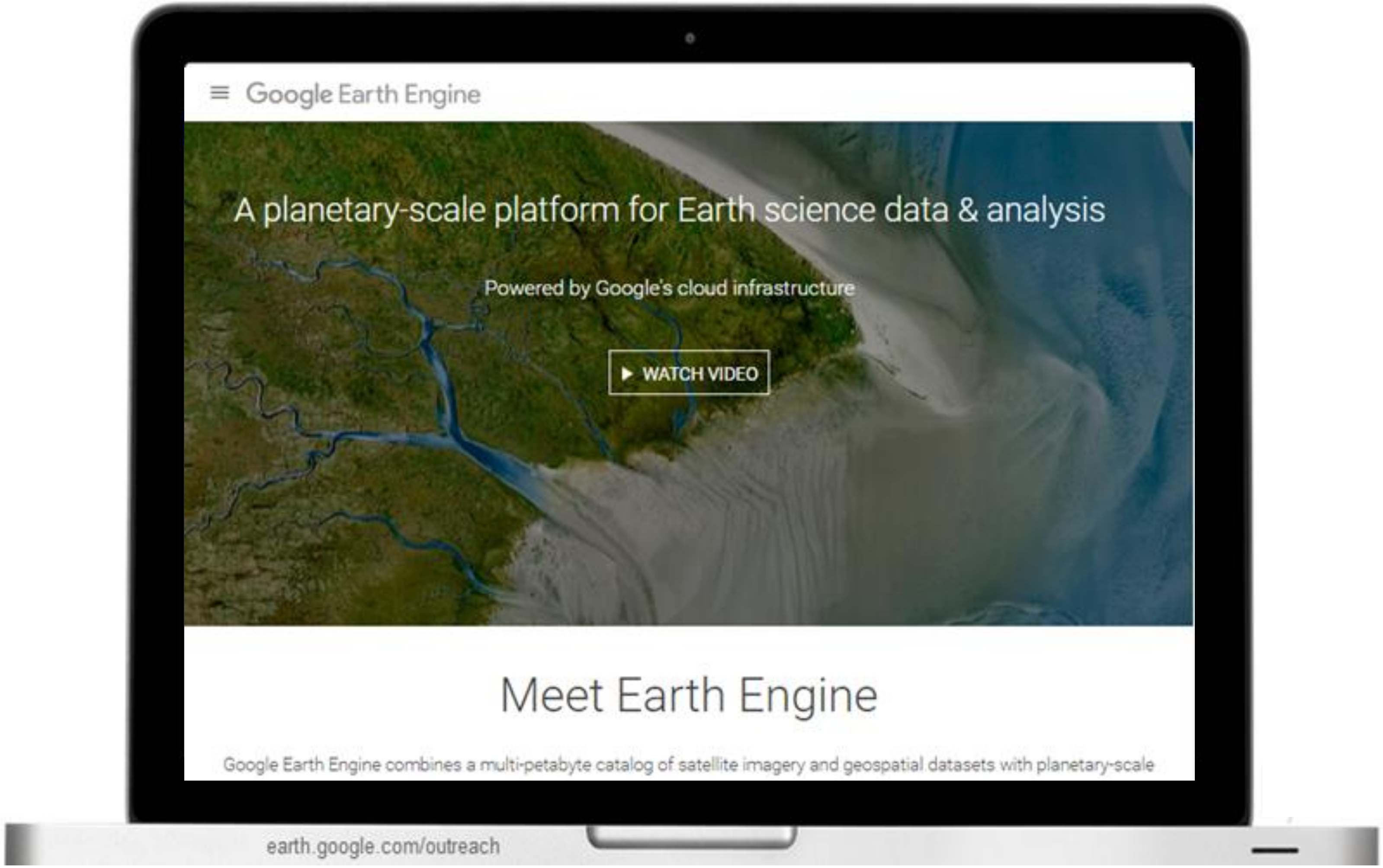
Kolokasi

- Big Data
- Komputasi skala besar
- Algoritma ilmiah

Dikhususkan untuk data geospasial

API untuk pengembangan aplikasi

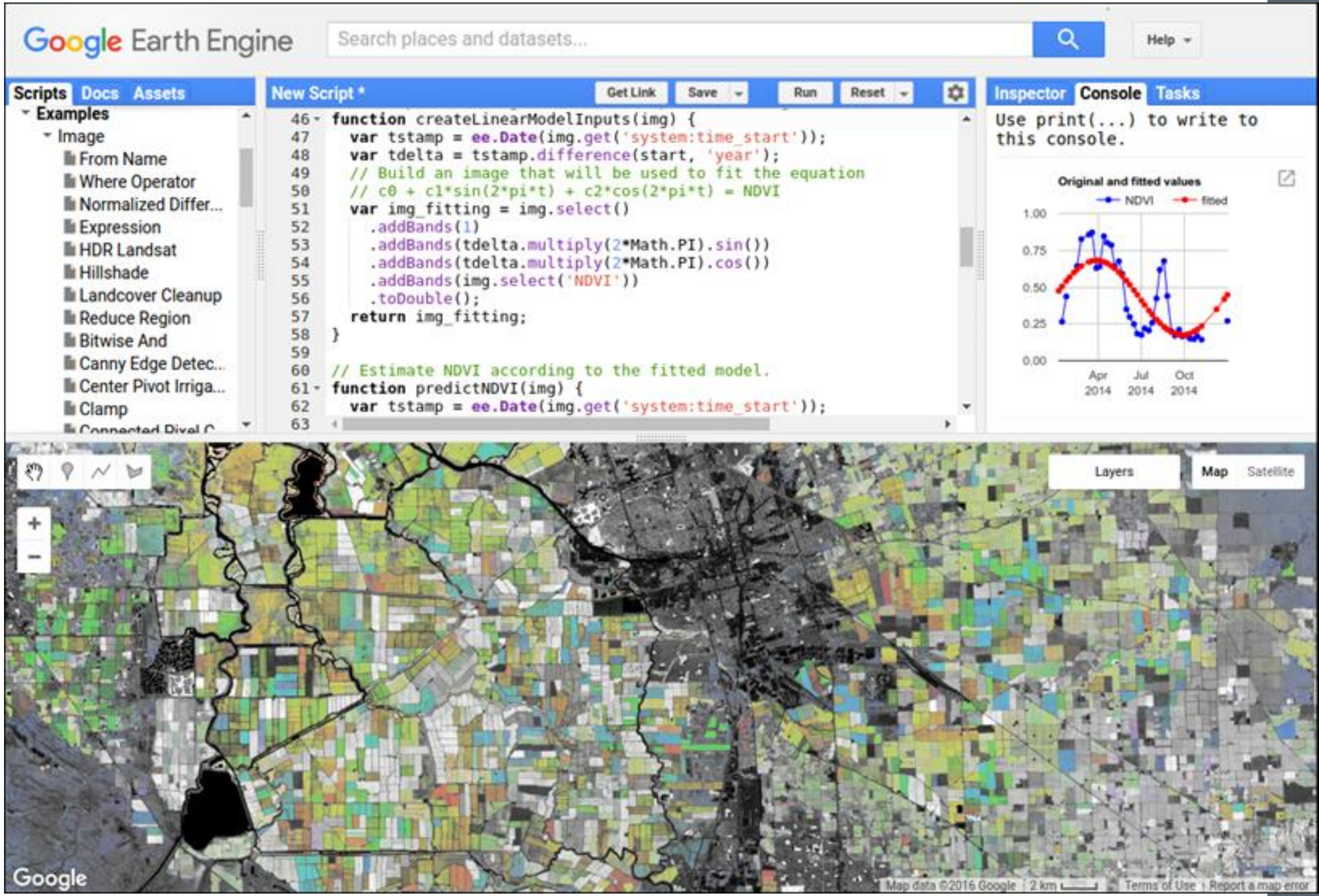
Go to earthengine.google.com



Platform > Code Editor

Code Editor

- Antarmuka interaktif untuk menggunakan GEE
- Javascript API



Tempat penulisan kode

Aset

Pencarian

Data Inspektor

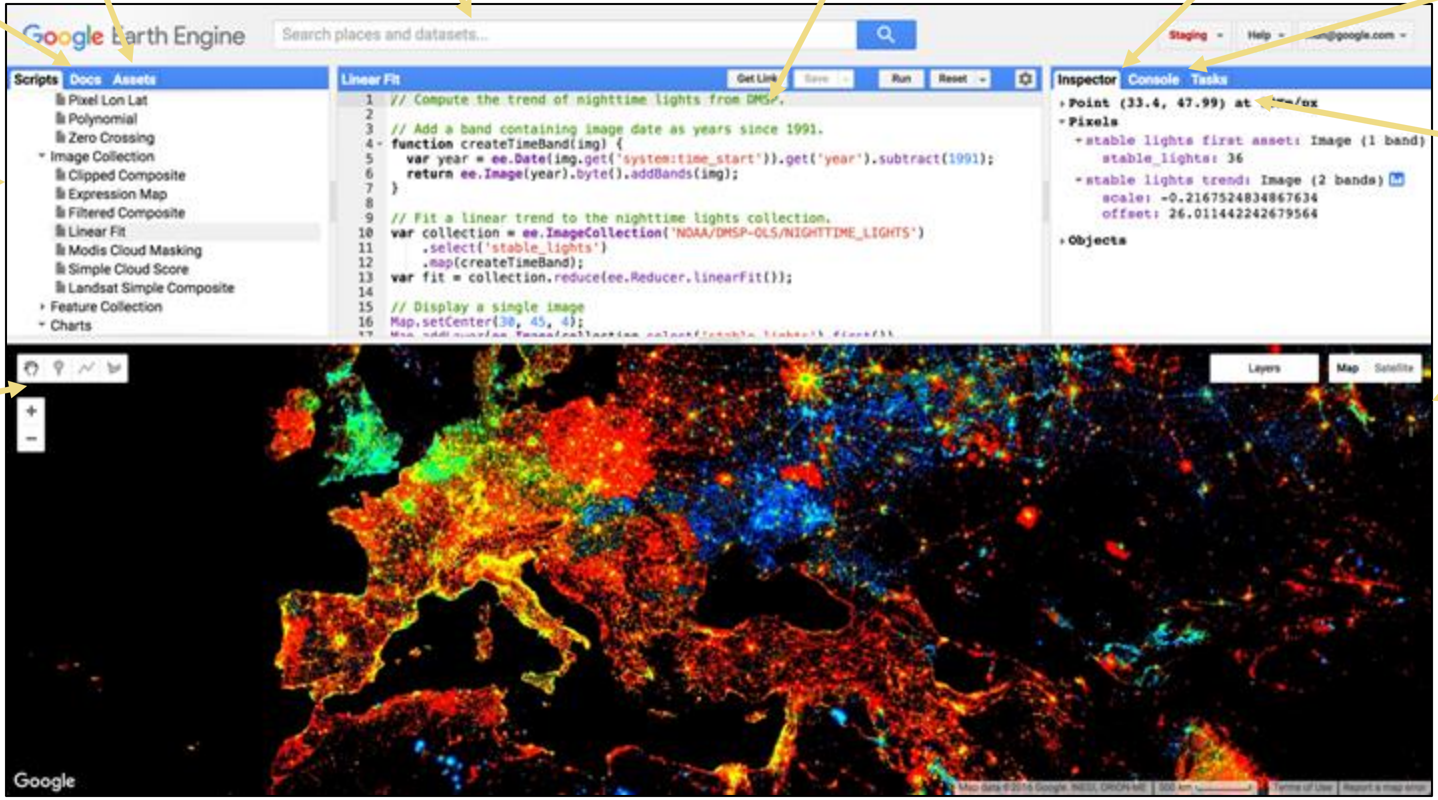
Dokumentasi API

Konsol output

Skrip kita

Contoh skrip

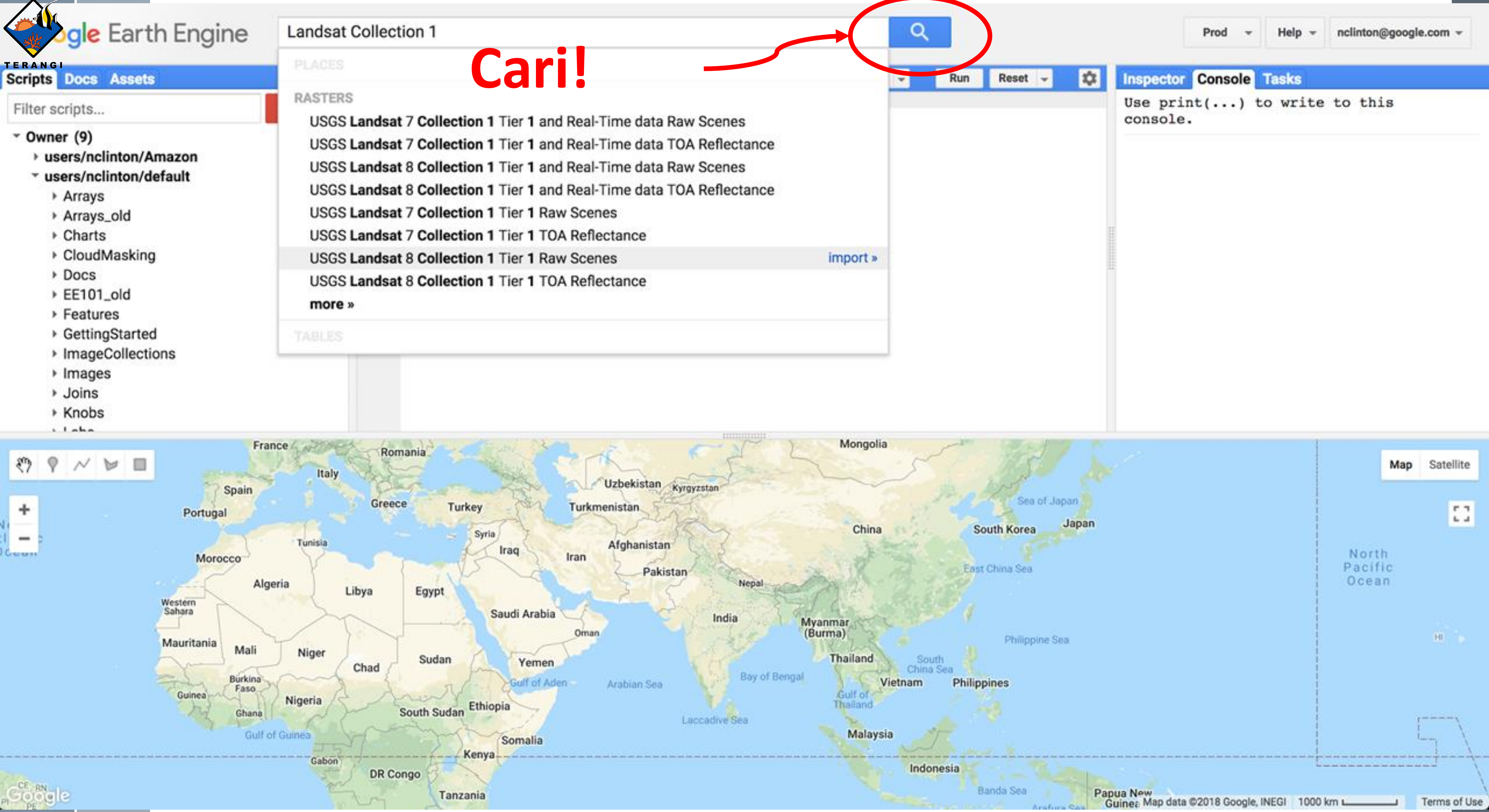
Tugas-tugas

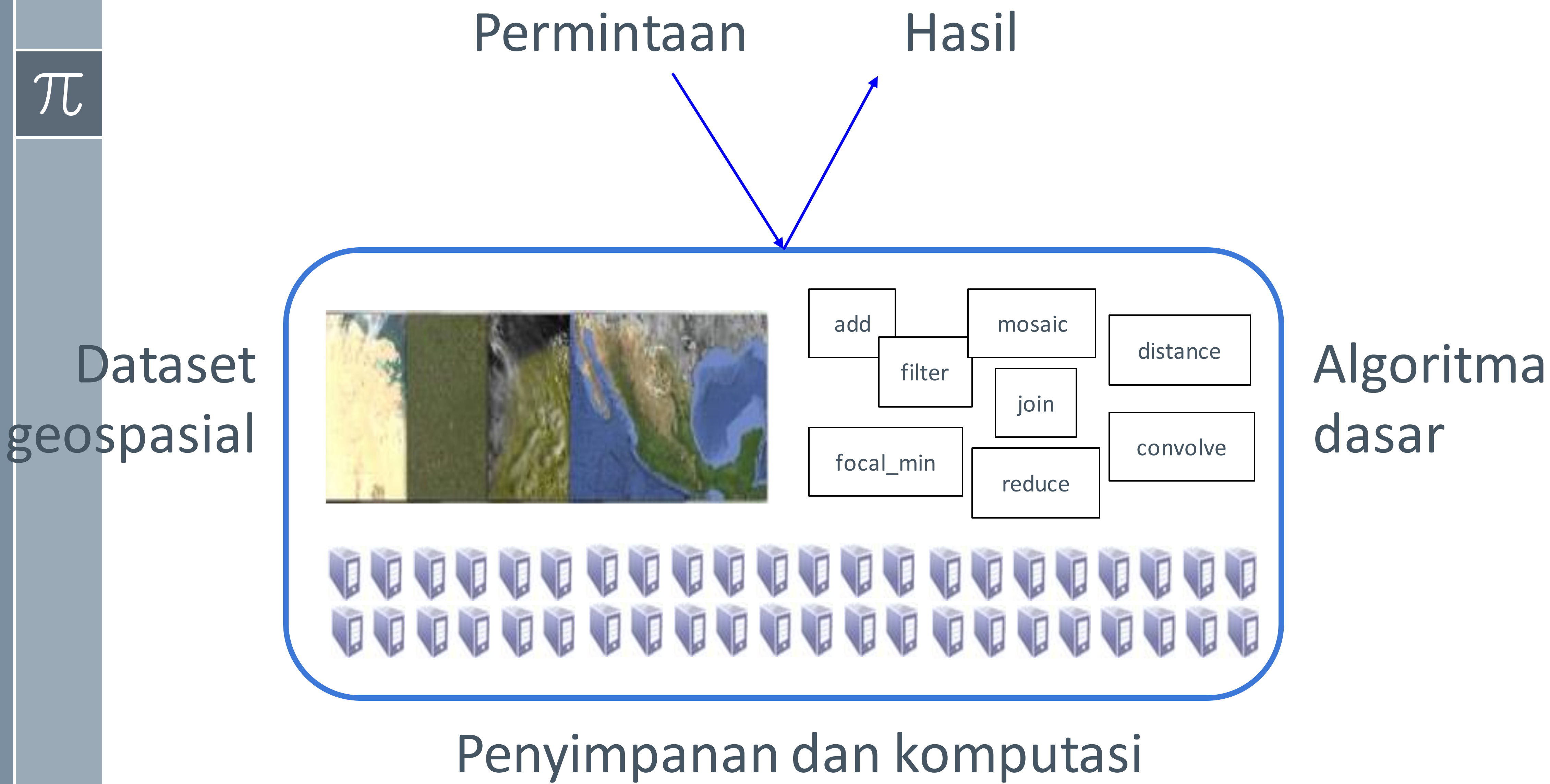


Alat gambar

Hasil peta

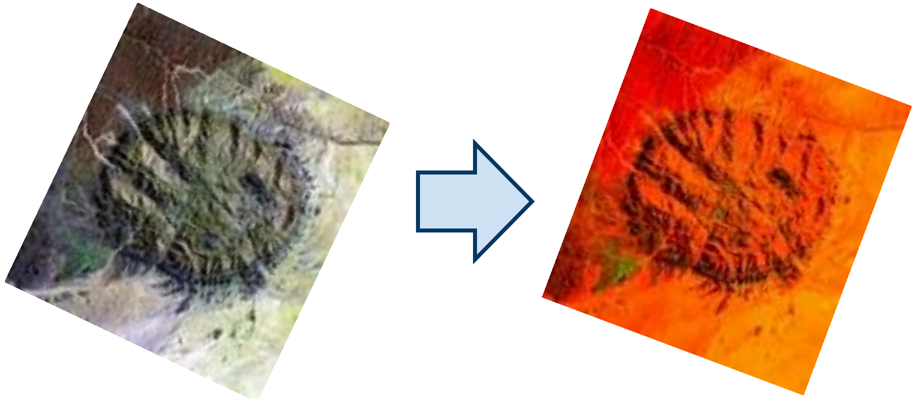
code.earthengine.google.com

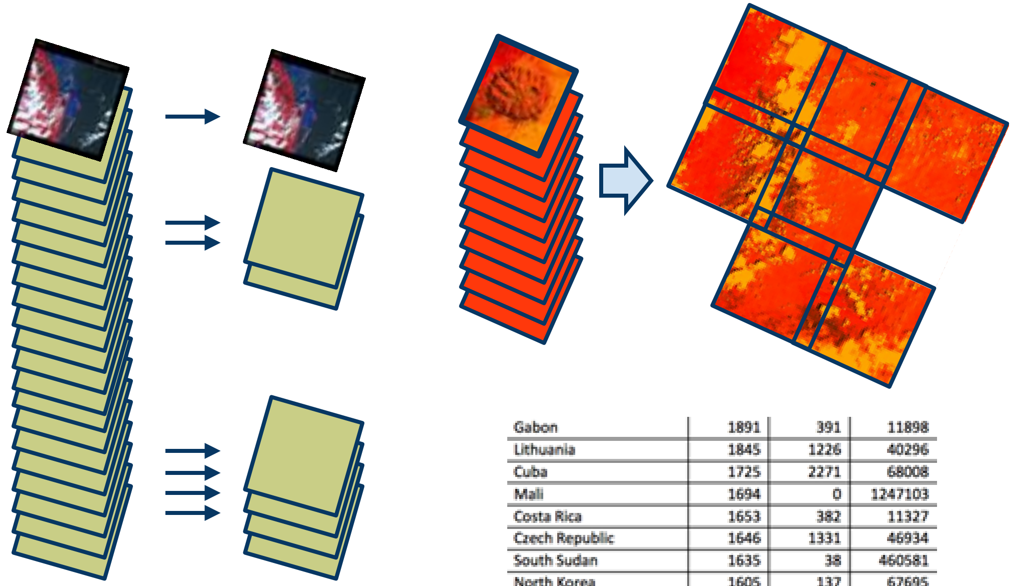

Google Earth Engine
Landsat Collection 1
Cari!
Prod
Help
nclinton@google.com
Scripts
Docs
Assets
Filter scripts...
Owner (9)
users/nclinton/Amazon
users/nclinton/default
Arrays
Arrays_old
Charts
CloudMasking
Docs
EE101_old
Features
GettingStarted
ImageCollections
Images
Joins
Knobs
PLACES
RASTERS
USGS Landsat 7 Collection 1 Tier 1 and Real-Time data Raw Scenes
USGS Landsat 7 Collection 1 Tier 1 and Real-Time data TOA Reflectance
USGS Landsat 8 Collection 1 Tier 1 and Real-Time data Raw Scenes
USGS Landsat 8 Collection 1 Tier 1 and Real-Time data TOA Reflectance
USGS Landsat 7 Collection 1 Tier 1 Raw Scenes
USGS Landsat 7 Collection 1 Tier 1 TOA Reflectance
USGS Landsat 8 Collection 1 Tier 1 Raw Scenes
import »
USGS Landsat 8 Collection 1 Tier 1 TOA Reflectance
more »
TABLES
Run
Reset
Inspector
Console
Tasks
Use print(...) to write to this console.
Map
Satellite
Map data ©2018 Google, INEGI
1000 km
Terms of Use


Permintaan
Hasil
Dataset geospasial
add
filter
mosaic
distance
join
focal_min
reduce
convolve
Algoritma dasar
Penyimpanan dan komputasi

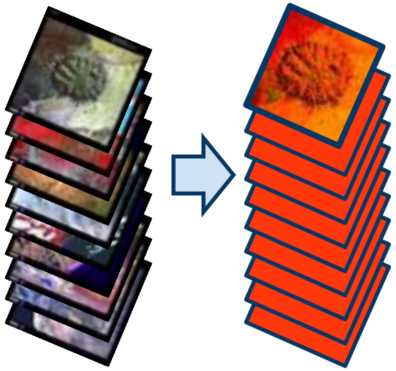
Dapat dilakukan GEE

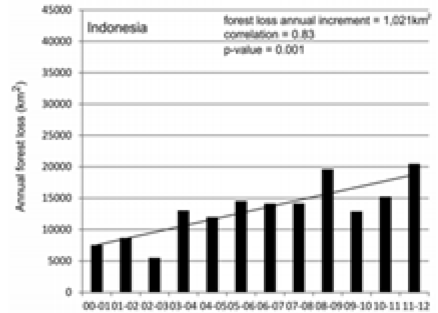
1. Mendapatkan citra
2. Implementasi algoritma pada citra
3. Memilih koleksi gambar
4. Implementasi algoritma pada koleksi gambar
5. Reduksi koleksi
6. Menghitung statistik agregat





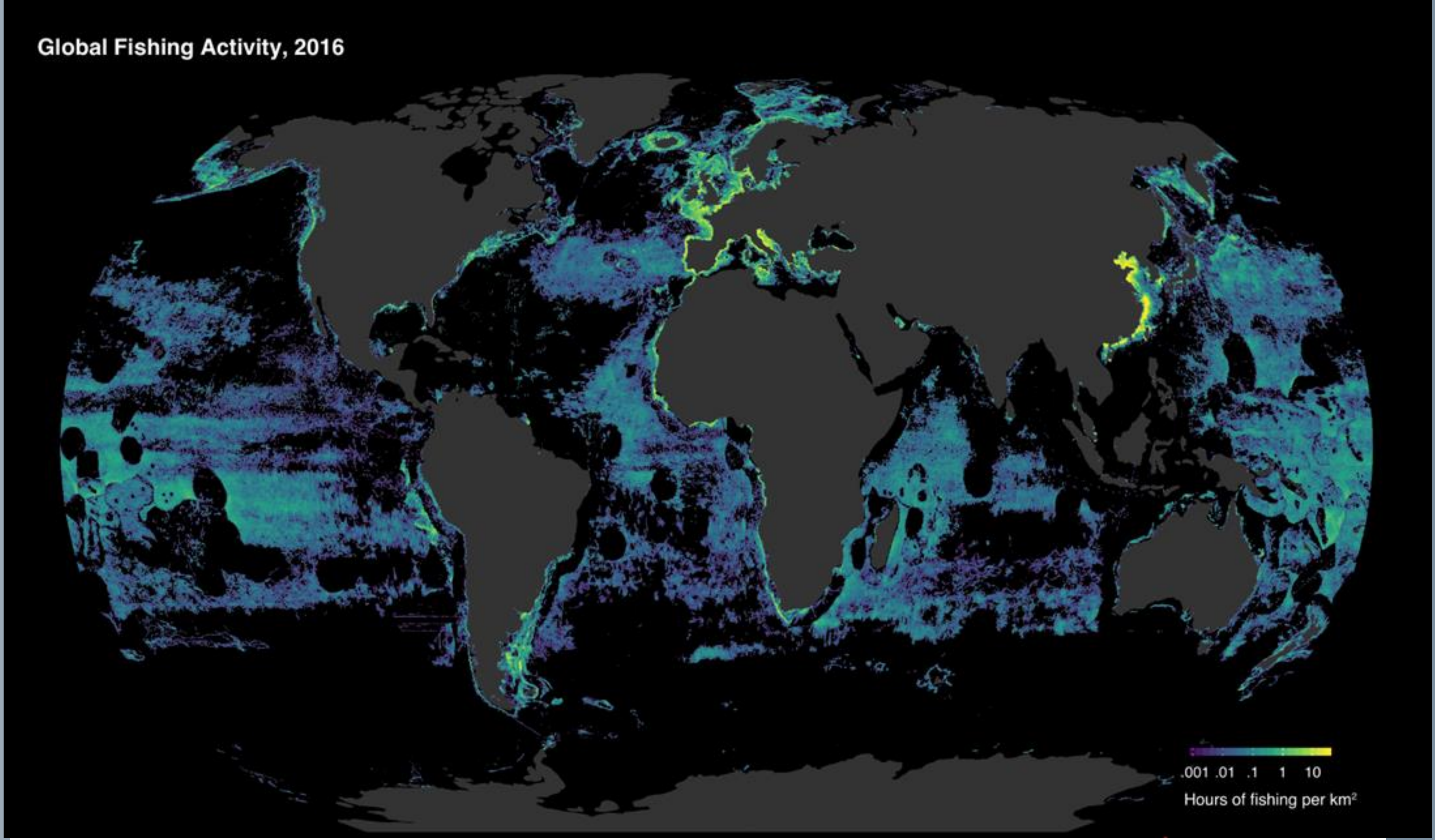
Gabon	1891	391	11898
Lithuania	1845	1226	40296
Cuba	1725	2271	68008
Mali	1694	0	1247103
Costa Rica	1653	382	11327
Czech Republic	1646	1331	46934
South Sudan	1635	38	460581
North Korea	1605	137	67695
Italy	1603	898	201331



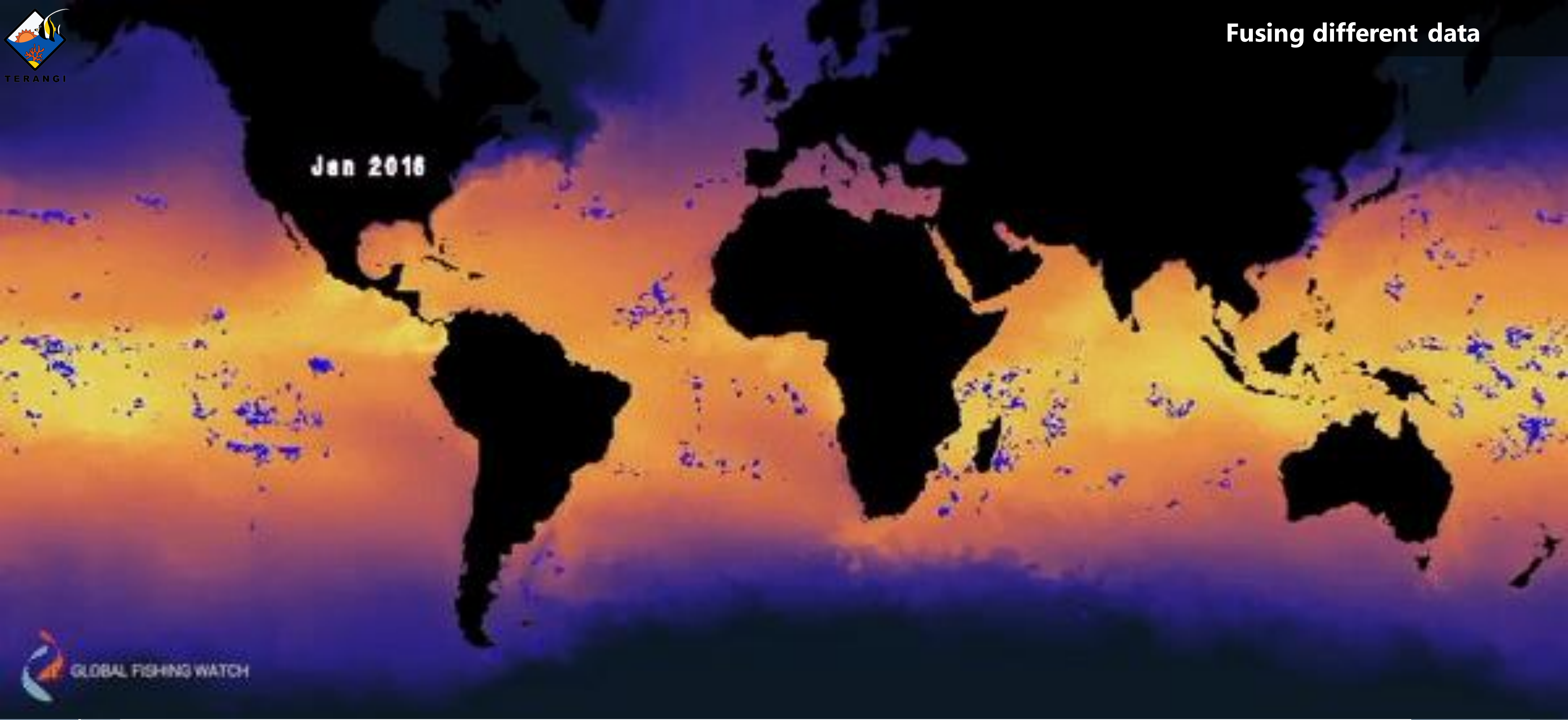


Beberapa contoh implementasi GEE

Safran Yusri dan Fakhrurrozi


Global Fishing Activity, 2016
.001 .01 .1 1 10
Hours of fishing per km²

Fusing different data

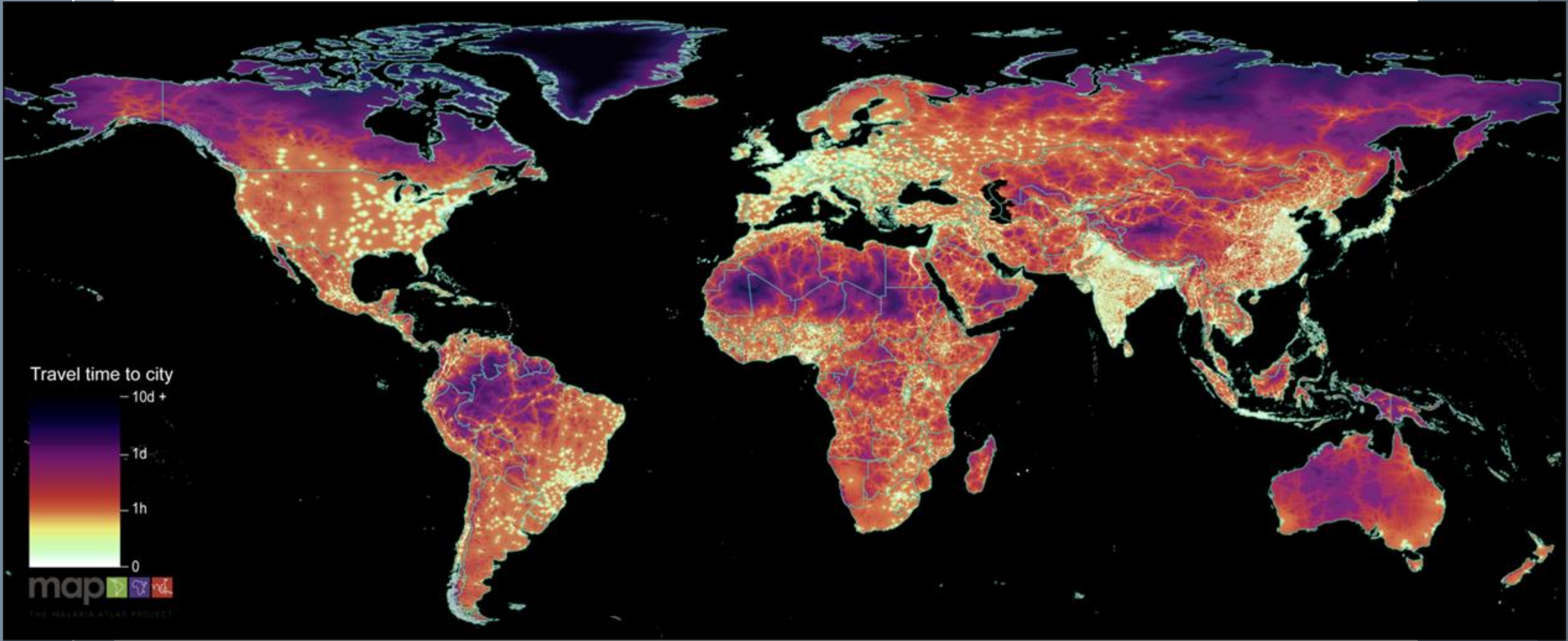


Longline fishing

Sea surface temperature

Google Earth Engine

Global Map of Accessibility to Cities *(Nature 2018)*



Source:

Understand Health care, jobs, extraction of resources

Space Data Journalism

Wonkblog • Analysis

Using the best data possible, we set out to find the middle of nowhere

By Andrew Van Dam February 20 at 10:33 AM Email the author

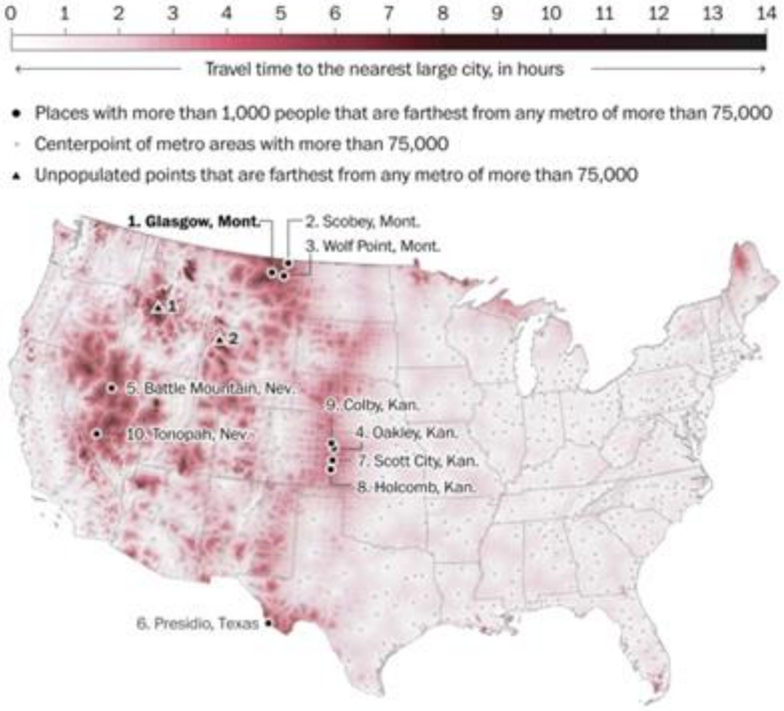
The northeastern Montana town of Glasgow, not far from the Fort Peck Indian Reservation, pictured above, is — according to an algorithm — just about as far as you can get from anywhere. (Jonathan Newton/The Washington Post)

In a triumph of data collection and analysis, a team of researchers based at Oxford University has built the tools necessary to calculate how far any dot on a map is from a city — or anything else.

The Washington Post

Where 'nowhere' is home

Small towns that are farthest from any metro, in terms of travel time



Note: Populations as of 2012-16. Small and midsize town sizes are based on populated places; travel is measured from their centerpoint. With the exception of Canadian border cities, city sizes are based on metro and micropolitan areas; travel is measured based on distance from any spot in their territory with a density of above 1,500 per square kilometer.

Sources: Malaria Atlas Project (travel); Census Bureau (cities); NASA (density, Canada, Mexico)

THE WASHINGTON POST

Praktik

Safran Yusri dan Fakhrurrozi

Antar muka GEE dan Javascript sederhana

1. Buka code.earthengine.google.com
2. Masukkan kode di samping
3. Klik Run
4. Perhatikan apa yang terjadi

```
// Antarmuka GEE dan javascript sederhana
// Dua garis miring jadi komentar
// titik koma jadi penanda akhir perintah
// String ditandai dengan tanda petik 1
var greeting = 'hello world';
print(greeting);
//Angka
var number = 42;
print(number);
//List
//tipe data list ditandai dengan kurung siku
var myList = ['a',greeting, number];
print(myList);

//Dictonary
// tipe data dictionary ditandai dengan kurung kurawal
var myDict = {
  'a': number,
  'b': myList,
  'c':1
};
print(myDict);
//Functions
//Parameter parameter di dalam kurung().
//Perintah ditandai dengan kurung kurawal {}.
var myFunction = function(input){
  return input
};
print(myFunction('hello!'));
```

Eksplorasi katalog data dalam GEE dan memilih dataset yang diinginkan

1. Pada kolom search, ketik SRTM
2. Pilih SRTM 30 M
3. Perhatikan cara memanggilnya
4. Kembali ke code editor dan ketikkan kode disamping

```
// Ambil data SRTM
var srtm = ee.Image('USGS/SRTMGL1_003');

// Zoom ke Puncak Jaya
Map.setCenter(137.930, -4.444, 9);

//Tampilkan gambar
Map.addLayer(srtm);

// Pilih band ketinggian
var elevation = srtm.select('elevation');

// Tampilkan gambar dengan pilihan rentang dan nama
Map.addLayer(elevation,{min: 0, max: 4000},'elevation');
```

Menampilkan koleksi citra

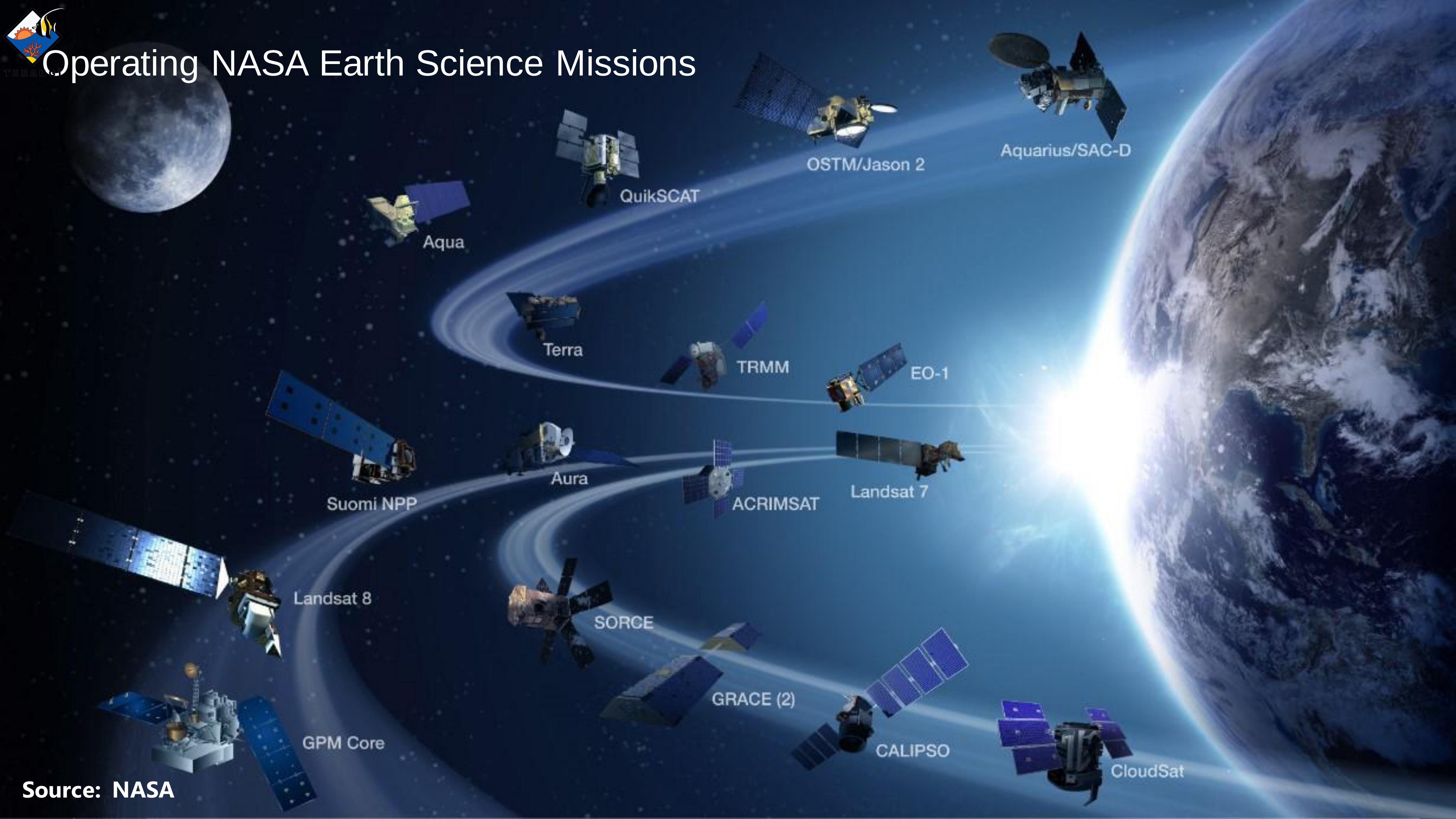
1. Buka code.earthengine.google.com
2. Masukkan kode di samping
3. Klik Run
4. Perhatikan dua tampilan peta

```
// Ambil koleksi gambar Landsat 8
var L8 =
ee.ImageCollection("LANDSAT/LC08/C01/T1_TOA");

// Zoom ke Jakarta
Map.setCenter(106.8420, -6.206, 8);

//Filter data sesuai tanggal yang diinginkan
var filtered = L8.filterDate('2017-01-01',
'2017-12-31');

//Tampilkan dalam peta
Map.addLayer(filtered);

// Tampilkan dengan kombinasi band
Map.addLayer(filtered, {min: 0, max :0.3,
bands:['B4', 'B3', 'B2']}, 'RGB');
```

Menggunakan citra LANDSAT 8 OLI - I

Safran Yusri

Operating NASA Earth Science Missions



Source: NASA

Citra Satelit

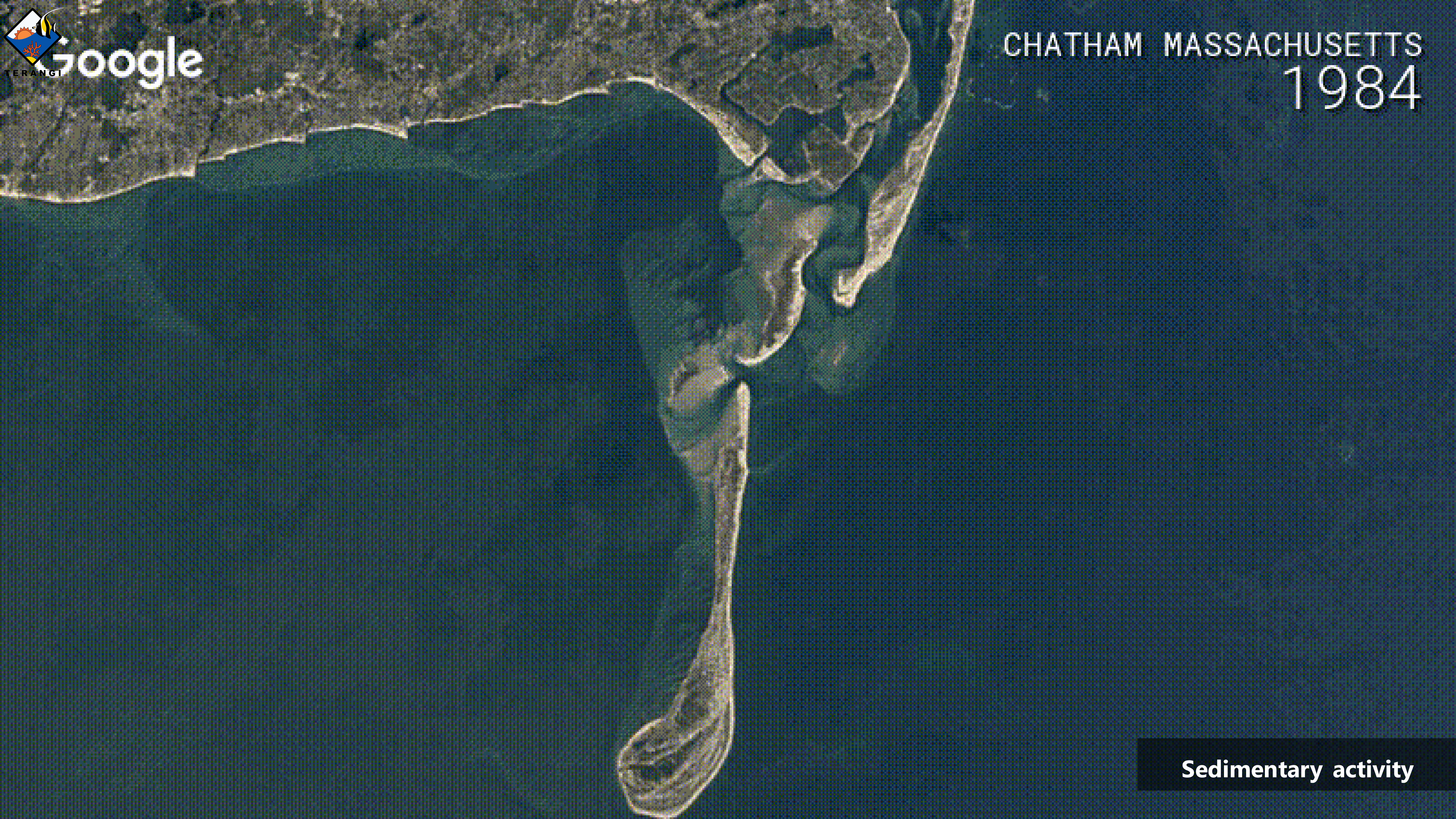
1 citra Landsat 8 :

- 64M piksel (resolusi 30m)
- 10 spectral bands
- 12 bits/band
- 600 citra/sehari

>6 JUTA CITRA
DARI 46 TAHUN MISI LANDSAT.

6 Million Landsat images *(1972-2018)*
2+ Petabytes stored on tapes at USGS

Earth Observation Data Archives


Google
TERANGI
CHATHAM MASSACHUSETTS
1984
Sedimentary activity

32 years
of satellite data

5,000,000+
Landsat scenes analyzed

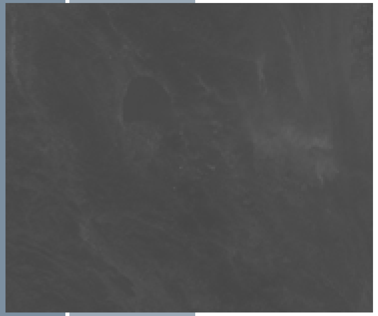
3
quadrillion pixels

More than **2M** hours of computation over **66,000** computers

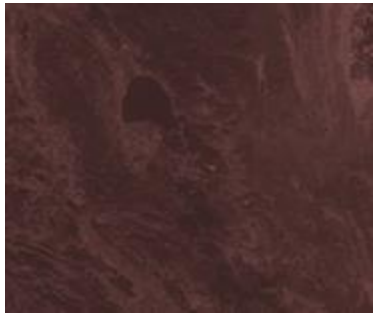
Elapsed time: **~1.5** days to build the mosaics

Landsat 8 Bands

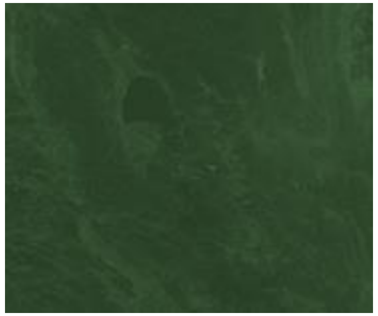
Coastal Aerosol



Red

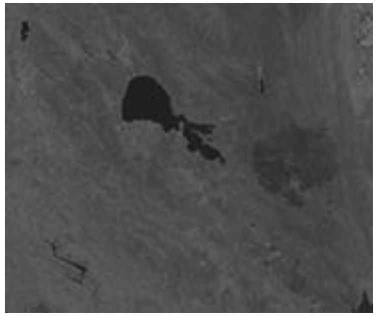


Green

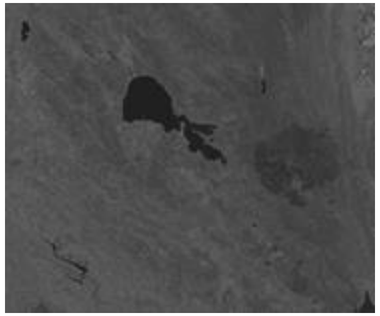


Blue

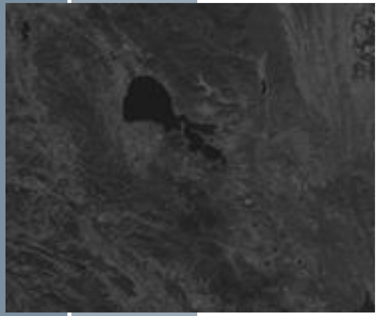
Near Infrared



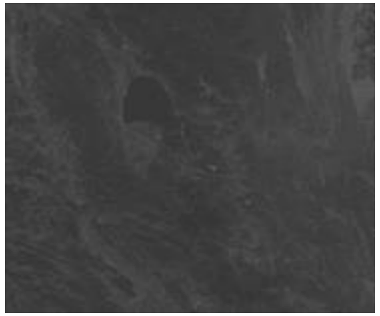
SWIR 1



SWIR 2

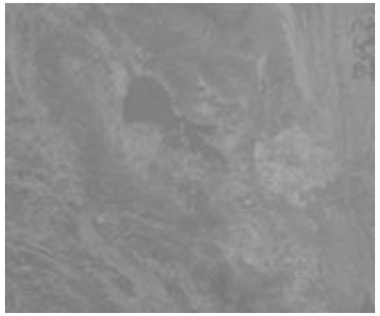


Panchromatic



Cirrus

Thermal 1



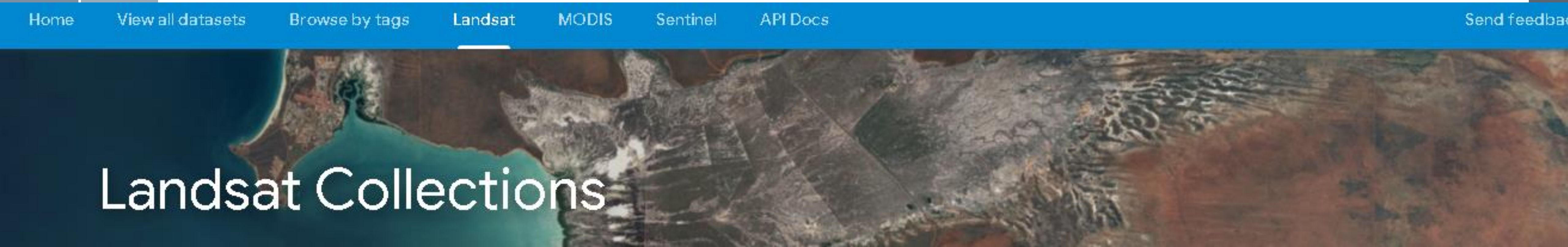
Thermal 2

QA

Setiap band punya nilai tersendiri

Koleksi citra LANDSAT 8

- Landsat 8: 2013 – sekarang
- Landsat 7: 2007 – sekarang
- Landsat 5: 1984 – 2012
- Landsat 4: 1982 – 1993
- Landsat 1 – 5 MSS: 1972 - 1999



Praktik

Safran Yusri dan Fakhrurrozi

Kategori data Landsat 8

› Raw ('LANDSAT/LC08/C01/T1')

› TOA ('LANDSAT/LC08/C01/T1_TOA')

› SR ('LANDSAT/LC08/C01/T1_SR')

Surface Reflectance

Atmospherically corrected surface reflectance from the Landsat 8 OLI/TIRS sensors

Dataset Availability: April 2013 - Present

Top of Atmosphere

Landsat 8 Collection 1 calibrated top-of-atmosphere (TOA) reflectance

Dataset Availability: April 2013 - Present

Raw Images

Landsat 8 Collection 1 DN values, representing scaled, calibrated at-sensor radiance.

Dataset Availability: April 2013 - Present

Menampilkan koleksi citra

1. Buka code.earthengine.google.com
2. Masukkan kode di samping
3. Klik Run
4. Perhatikan tiga tampilan peta

```
// Ambil koleksi gambar Landsat 8 raw
var L8 =
ee.ImageCollection("LANDSAT/LC08/C01/T1");
// Ambil koleksi gambar Landsat 8 TOA
var L8toa =
ee.ImageCollection("LANDSAT/LC08/C01/T1_TOA
");
// Ambil koleksi gambar Landsat 8 TOA
var L8sr =
ee.ImageCollection("LANDSAT/LC08/C01/T1_SR")
;
Map.setCenter(106.8420, -6.206, 8);
//Filter data sesuai tanggal yang diinginkan
var filteredraw = L8.filterDate('2017-01-01', '2017-
12-31');
var filteredtoa = L8toa.filterDate('2017-01-01',
'2017-12-31');
var filteredsr = L8sr.filterDate('2017-01-01', '2017-
12-31');
// Tampilkan dengan kombinasi band
Map.addLayer(filteredraw, {min:6000, max:60000,
bands:['B4', 'B3', 'B2']}, 'RGB raw');
Map.addLayer(filteredtoa, {min: 0, max:0.3,
bands:['B4', 'B3', 'B2']}, 'RGB toa');
Map.addLayer(filteredsr, {min:0, max:12000,
bands:['B4', 'B3', 'B2']}, 'RGB sr');
```

Membuat mosaik bebas awan

1. Buka code.earthengine.google.com
2. Masukkan kode di samping
3. Klik Run
4. Perhatikan dua tampilan peta

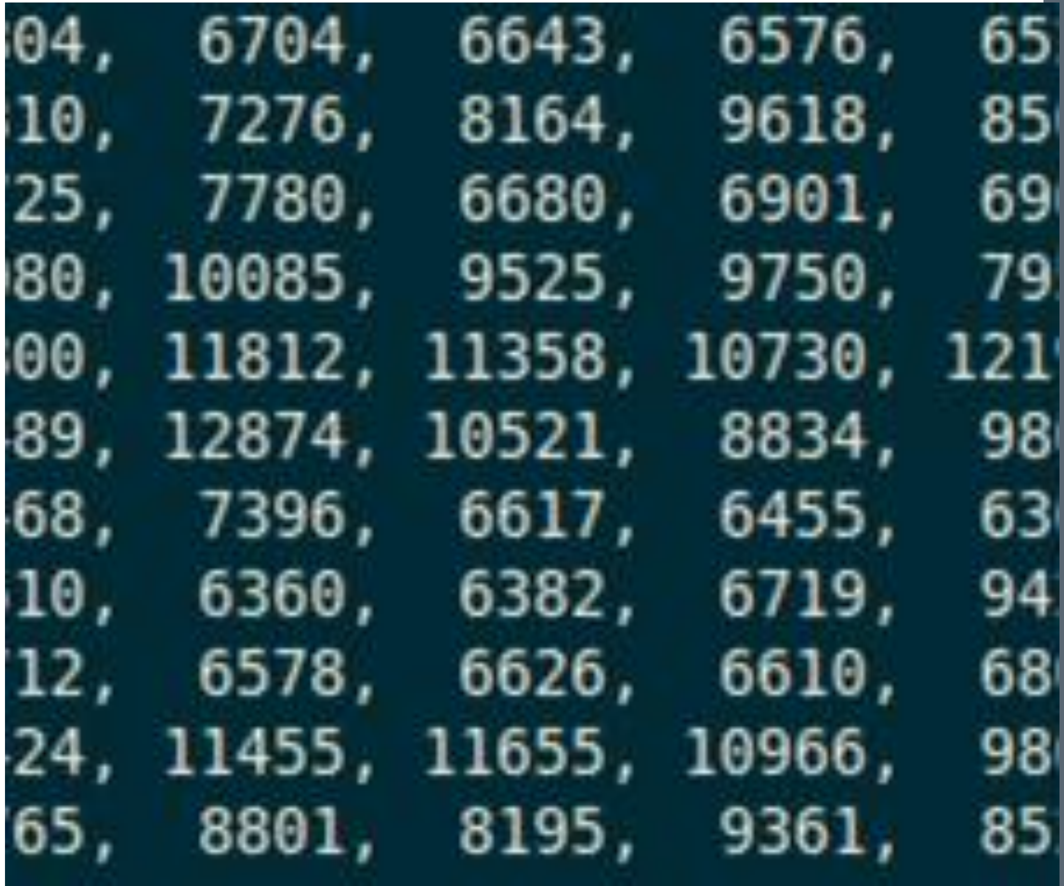
```
// Ambil koleksi gambar Landsat 8 raw
var L8 =
ee.ImageCollection("LANDSAT/LC08/C01/T1");
Map.setCenter(106.8420, -6.206, 8);
//Filter data sesuai tanggal yang diinginkan
var filteredraw = L8.filterDate('2017-01-01', '2017-
12-31');
// Perintah membuat komposit bebas awan.
var composite =
ee.Algorithms.Landsat.simpleComposite({
  collection: filteredraw,
  asFloat: true
});
// Tampilkan dengan kombinasi band
Map.addLayer(filteredraw, {min:6000, max:60000,
bands:['B4', 'B3', 'B2']}, 'RGB raw');
Map.addLayer(composite, {bands: ['B6', 'B5', 'B4'],
max: [0.3, 0.4, 0.3]}, 'composite');
```

Menggunakan citra LANDSAT 8 OLI - II

Safran Yusri

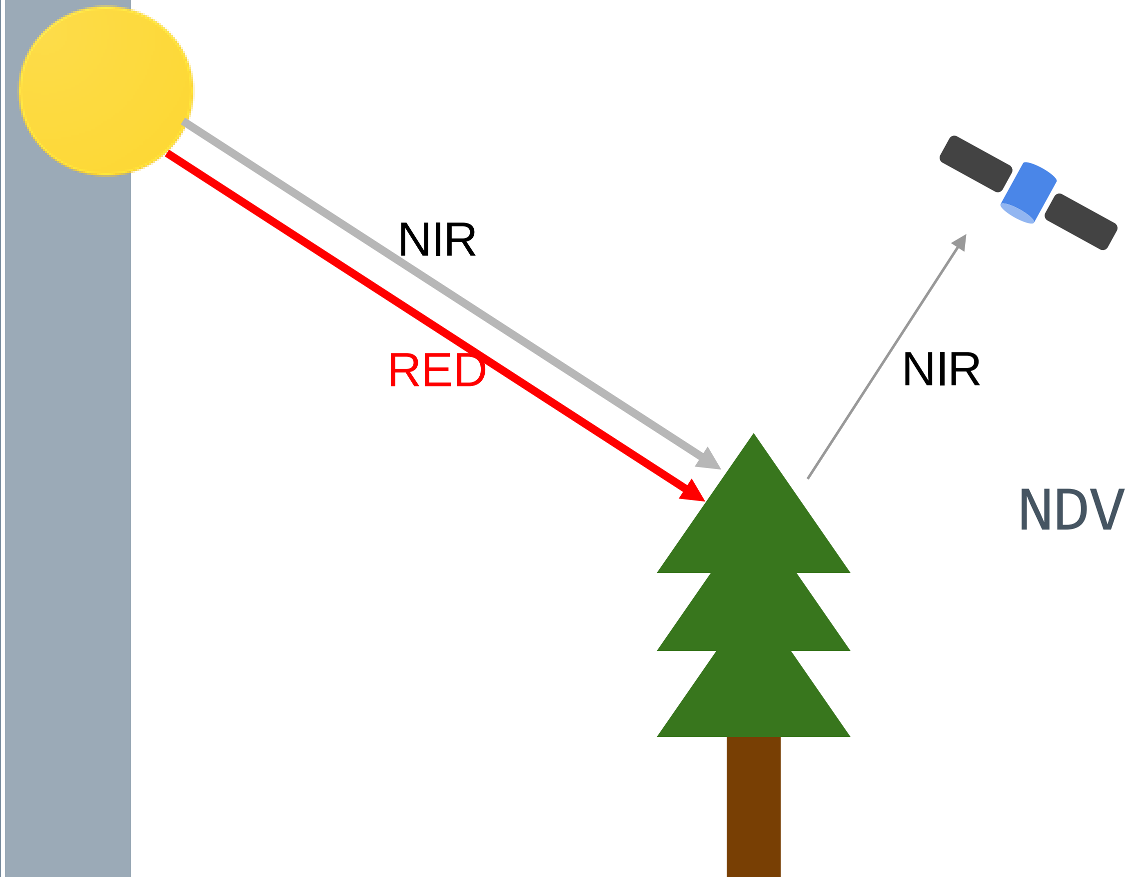
Nilai Dijital

- Setiap piksel memiliki nilai dijital dari masing-masing band -> operasi matematika dapat dilakukan

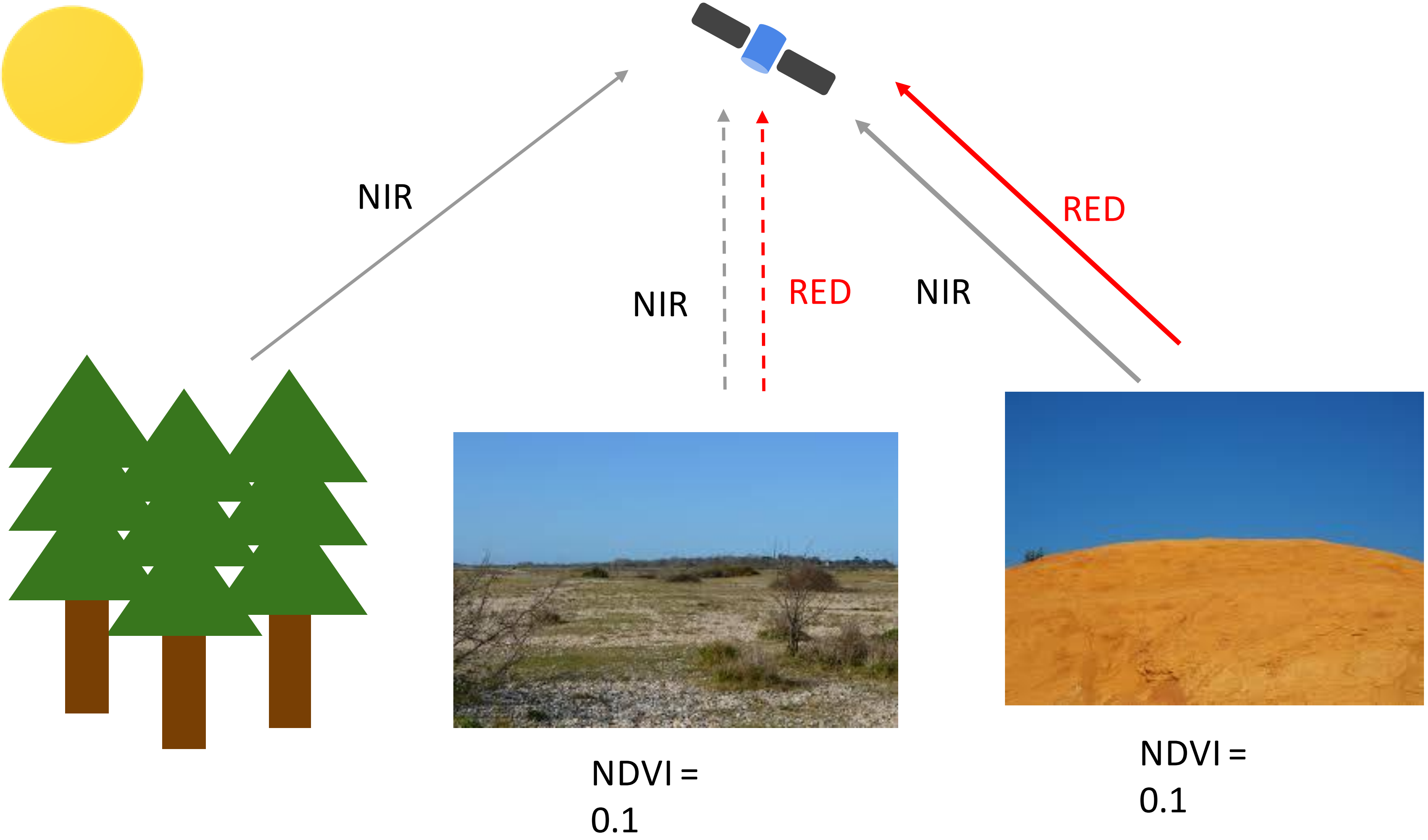


Normalized Difference Vegetation Index

› Indeks vegetasi yang biasa digunakan untuk mengetahui kondisi tutupan vegetasi

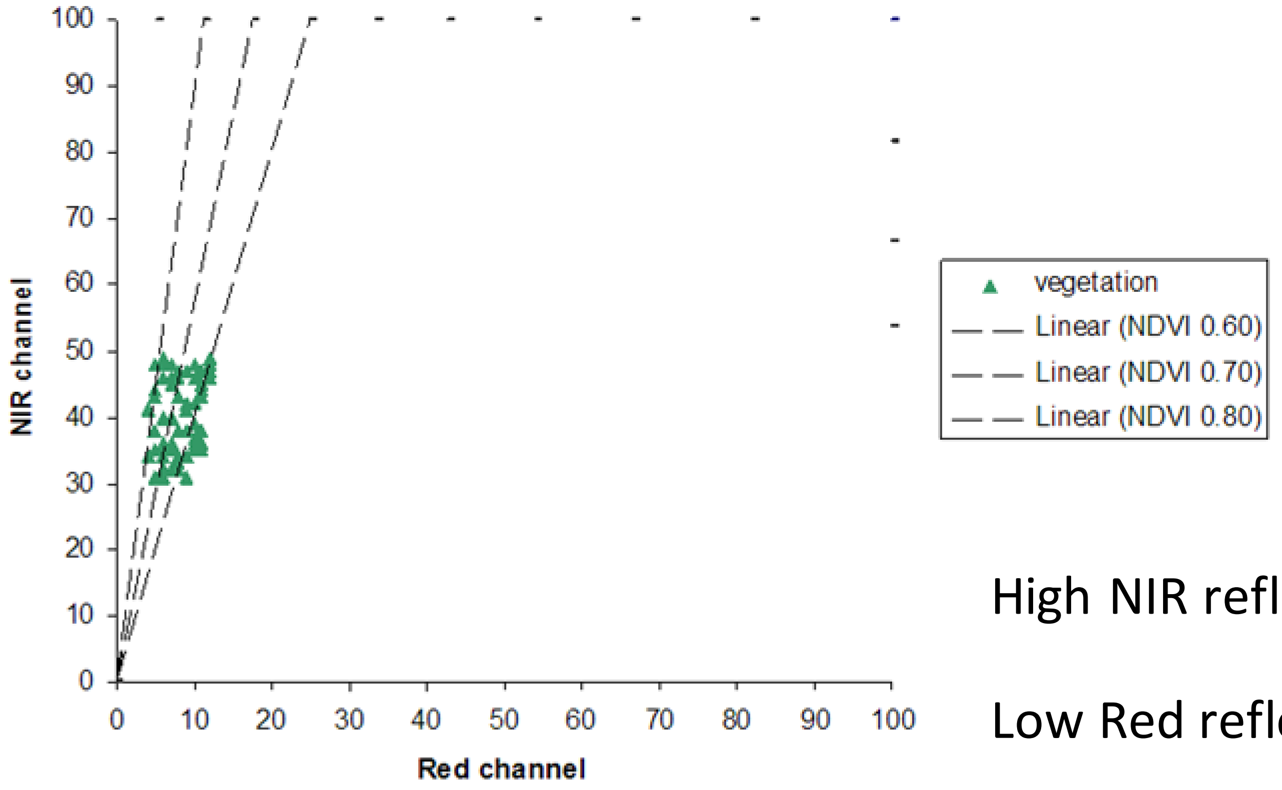


$$\text{NDVI} = \frac{(\text{Near Infrared} - \text{Red})}{(\text{Near Infrared} + \text{Red})}$$


NIR
NIR
RED
NIR
RED
NDVI =
0.1
NDVI =
0.1

NDVI | Hutan tropis

(NIR-Red)/(NIR+Red)



High NIR reflectance

Low Red reflectance

Landsat 8 Surface Reflectance

- Sudah dikoreksi atmosferik
- Terdiri atas 5 visible, NIR, SWIR, TIRS, pixel_qa
- Mosaik bebas awan menggunakan pixel_qa band
- Membutuhkan function yang membaca tiap gambar dalam koleksi

Surface Reflectance

Atmospherically corrected surface reflectance from the Landsat 8 OLI/TIRS sensors

Dataset Availability: April 2013 - Present

Land masking

- Membatasi analisis hanya pada perairan
- Menghilangkan daratan
- Dapat dilakukan menggunakan polygon
- Dapat dilakukan dengan layer masking
- Menggunakan hansen et al 2013
- Terdapat pixel air dan non air

TERANGI

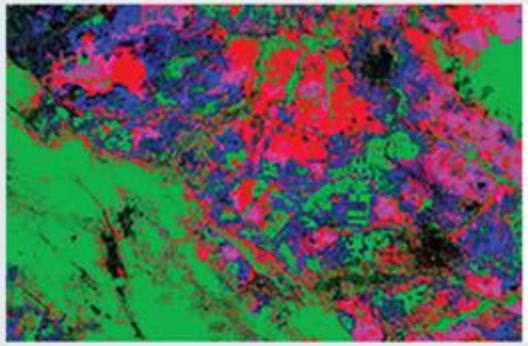
Science

AAAS.ORG | FEEDBACK | HELP | LIBRARIANS

AAAS

NEWS | SCIENCE JOURNALS | CAREERS

:: Special: The Heavily Connected Brain :: Science in the Classroom :: Special: C



15 NOVEMBER 2013

ECOLOGY

Seeing the Forest

Landsat data reveals details of forest losses and gains across the globe on an annual basis from 2000 to 2012.

REPORT

High-Resolution Global Maps of 21st-Century Forest Cover Change

M. C. Hansen[1,*], P. V. Potapov[1], R. Moore[2], M. Hancher[2], S. A. Turubanova[1], A. Tyukavina[1], D. Thau[2], S. V. Stehman[3], S. J. ...

+ See all authors and affiliations

Science 15 Nov 2013:
Vol. 342, Issue 6160, pp. 850-853
DOI: 10.1126/science.1244693

Article | Figures & Data | Info & Metrics | eLetters | PDF

You are currently viewing the abstract.

View Full Text

Forests in Flux

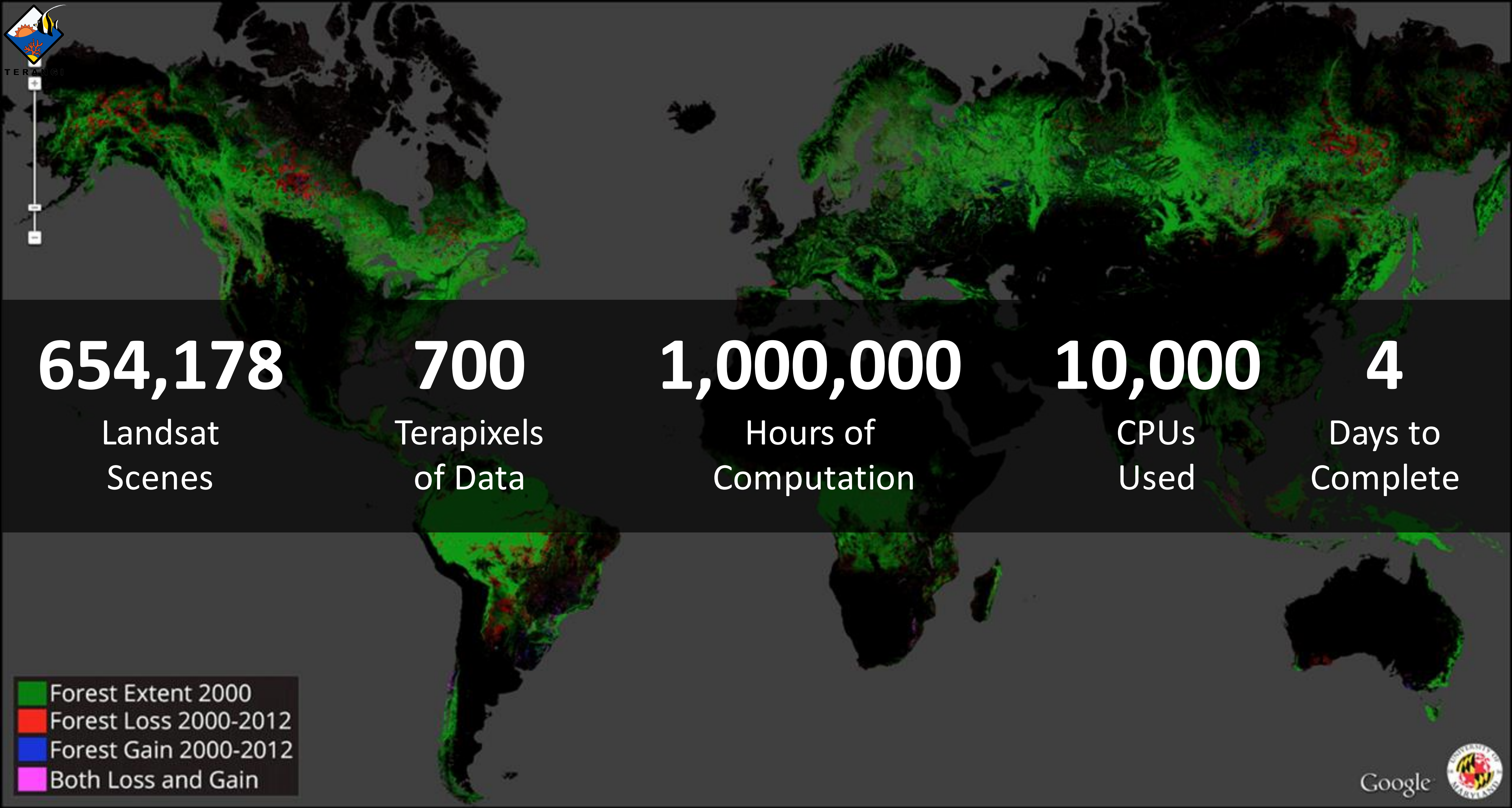
Forests worldwide are in a state of flux, with accelerating losses in some regions and gains in others. **Hansen *et al.*** (p. 850) examined global Landsat data at a 30-meter spatial resolution to characterize forest extent, loss, and gain from 2000 to 2012. Globally, 2.3 million square kilometers of forest were lost during the 12-year study period and 0.8 million square kilometers of new forest were gained. The tropics exhibited both the greatest losses and the greatest gains (through regrowth and plantation), with losses outstripping gains.

Abstract

Quantification of global forest change has been lacking despite the recognized importance of forest ecosystem services. In this study, Earth observation satellite data were used to map

Forest Extent 2000
Forest Loss 2000-2012
Forest Gain 2000-2012
Both Loss and Gain

Global Forest Change, 2000-2012 Source: Hansen, Potapov, Moore, Hancher, et al. (Science, 2013) Powered by Google Earth Engine



654,178 Landsat Scenes

700 Terapixels of Data

1,000,000 Hours of Computation

10,000 CPUs Used

4 Days to Complete

Global Forest Change, 2000-2012 Source: Hansen, Potapov, Moore, Hancher, et al. (Science, 2013) Powered by Google Earth Engine

Praktik

Safran Yusri dan Fakhrurrozi

Menghitung NDVI

1. Buka code.earthengine.google.com
2. Masukkan kode di samping
3. Klik Run
4. Perhatikan empat tampilan peta

```
// Ambil 1 citra landsat
var image =
ee.Image('LANDSAT/LC08/C01/T1_TOA/LC08_125063_20180524')
;
// tampilkan
Map.addLayer(image, {bands: ['B4', 'B3', 'B2'], max: 0.3, gamma: 1.2},
'Image');
Map.centerObject(image);
// hitung NDVI
var ndvi = image.normalizedDifference(['B5','B4']);
// Palette untuk NDVI
var palette = [
  'FFFFFF', 'CE7E45', 'DF923D', 'F1B555', 'FCD163', '99B718',
  '74A901', '66A000', '529400', '3E8601', '207401', '056201',
  '004C00', '023B01', '012E01', '011D01', '011301'];
// tampilkan the NDVI image
Map.addLayer(ndvi, {min: 0, max: 1, palette: palette}, 'Landsat NDVI',
false);
// Dapat pula digunakan untuk mosaik bebas awan
// Ambil koleksi citra landsat
var collection =
ee.ImageCollection('LANDSAT/LC08/C01/T1').filterDate('2017-01-
01', '2017-12-21');
// buat mosaik bebas awan
var composite =
ee.Algorithms.Landsat.simpleComposite(collection);
// hitung ndvi
var composite_ndvi = composite.normalizedDifference(['B5','B4']);
// tampilkan komposit dan komposit ndvi
Map.addLayer(composite, {bands: ['B4', 'B3', 'B2'], max: 128},
'Composite', false);
Map.addLayer(composite_ndvi, {min: 0, max: 1, palette: palette},
'Composite NDVI', false);
```

Mosaik bebas awan Landsat SR

1. Buka code.earthengine.google.com
2. Masukkan kode di samping
3. Klik Run
4. Perhatikan tampilan peta

```
/**
 * Function to mask clouds based on the pixel_qa band of Landsat 8 SR data.
 * @param {ee.Image} image input Landsat 8 SR image
 * @return {ee.Image} cloudmasked Landsat 8 image
 */
function maskL8sr(image) {
  // Bits 3 and 5 are cloud shadow and cloud, respectively.
  var cloudShadowBitMask = (1 << 3);
  var cloudsBitMask = (1 << 5);
  // Get the pixel QA band.
  var qa = image.select('pixel_qa');
  // Both flags should be set to zero, indicating clear conditions.
  var mask = qa.bitwiseAnd(cloudShadowBitMask).eq(0)
                 .and(qa.bitwiseAnd(cloudsBitMask).eq(0));
  return image.updateMask(mask);
}

var dataset = ee.ImageCollection('LANDSAT/LC08/C01/T1_SR')
                  .filterDate('2016-01-01', '2016-12-31')
                  .map(maskL8sr);

var visParams = {
  bands: ['B4', 'B3', 'B2'],
  min: 0,
  max: 3000,
  gamma: 1.4,
};
Map.setCenter(114.0079, -26.0765, 9);
Map.addLayer(dataset.median(), visParams);
```

Land/water masking

1. Buka code.earthengine.google.com
2. Masukkan kode di samping
3. Klik Run
4. Perhatikan tiga tampilan peta

```
/**
 * Function to mask clouds based on the pixel_qa band of Landsat 8 SR data.
 * @param {ee.Image} image input Landsat 8 SR image
 * @return {ee.Image} cloudmasked Landsat 8 image
 */
function maskL8sr(image) {
  // Bits 3 and 5 are cloud shadow and cloud, respectively.
  var cloudShadowBitMask = (1 << 3);
  var cloudsBitMask = (1 << 5);
  // Get the pixel QA band.
  var qa = image.select('pixel_qa');
  // Both flags should be set to zero, indicating clear conditions.
  var mask = qa.bitwiseAnd(cloudShadowBitMask).eq(0)
                 .and(qa.bitwiseAnd(cloudsBitMask).eq(0));
  return image.updateMask(mask);
}
var dataset = ee.ImageCollection('LANDSAT/LC08/C01/T1_SR')
                  .filterDate('2016-01-01', '2016-12-31')
                  .map(maskL8sr);
var visParams = {
  bands: ['B4', 'B3', 'B2'],
  min: 0,
  max: 3000,
  gamma: 1.4,
};
var composite = dataset.median()
Map.setCenter(106.28107, -5.81314, 14);
Map.addLayer(composite, visParams);
// Land masking

// ambil data Hansen et al
var hansenImage = ee.Image('UMD/hansen/global_forest_change_2015');

// Select the land/water mask.
var datamask = hansenImage.select('datamask');

// buat masking biner.
var maskland = datamask.eq(2);
var watermask= datamask.eq(1);

// Masking darat
var maskedComposite = composite.updateMask(maskland);
// masking laut
var landComposite = composite.updateMask(watermask);
// tampilkan komposit laut
Map.addLayer(maskedComposite, visParams, 'Landsat 8 Water Composite');
Map.addLayer(landComposite, visParams, 'Landsat 8 Land Composite');
```

Mangrove Index

Safran Yusri

Based on the work of Baloloy et al. in prep

Langkah-langkah

1. Buat filtertemporal dan spasial
2. Buka koleksi citra yang diinginkan
3. Buat mosaic komposit bebas awan
4. Lakukan masking sesuai kebutuhan
5. Masukkan persamaan ke dalam skrip
6. Ujicoba ambang batas (threshold)
7. Edit skrip untuk memasukkan ambang batas
8. Ekspor data

Logika thresholding

- Jika > 4 -> 1 (yes)
- Jika < 20 -> 1 (yes)
- Jika < 4 -> 1 (yes)
- Jika > 20 -> 1 (yes)
- 4 > x > 20 -> 2

Langkah 1 - 2

- // Step 1. Create filters

- // Center the map on aoi.
- var bound = aoi.bounds();
- Map.centerObject(bound, 12);

- //Construct start and end dates:
- var start = ee.Date('2017-01-01');
- var finish = ee.Date('2018-12-31');

- // Step 2. Load landsat 8 image collection of indonesia

- // Load Landsat 8 surface reflectance data
- var l8sr = ee.ImageCollection('LANDSAT/LC08/C01/T1_SR')
-     .filterBounds(bound)
-     .filterDate(start, finish);

```
// Step 3. Create a cloud free mosaic

// Function to cloud mask from the Fmask band of Landsat 8 SR data.
function maskL8sr(image) {
  // Bits 3 and 5 are cloud shadow and cloud, respectively.
  var cloudShadowBitMask = ee.Number(2).pow(3).int();
  var cloudsBitMask = ee.Number(2).pow(5).int();

  // Get the pixel QA band.
  var qa = image.select('pixel_qa');

  // Both flags should be set to zero, indicating clear conditions.
  var mask = qa.bitwiseAnd(cloudShadowBitMask).eq(0)
      .and(qa.bitwiseAnd(cloudsBitMask).eq(0));

  // Return the masked image, scaled to [0, 1].
  return image.updateMask(mask).divide(10000);
}
```

Langkah 4

- // Map the function over one year of data and take the median.
- var composite = l8sr.map(maskL8sr)
- .reduce(ee.Reducer.median());

- // Make a handy variable of visualization parameters.
- var visParams = {bands: ['B4_median', 'B3_median', 'B2_median'], min: 0, max: 0.2};

- // Display landsat 8 surface reflectance cloud free composite.
- Map.addLayer(composite, visParams, 'Landsat 8 Composite');

Langkah 5

```
// Step 5. Compute the MVI using an expression.
var mvi = lcomposite.expression(
      '(NIR - GREEN)/(SWIR - GREEN)', {
         'NIR': lcomposite.select('B5_median'),
         'GREEN': lcomposite.select('B3_median'),
         'SWIR': lcomposite.select('B6_median')
}).rename('mvi');

// Display MVI
Map.addLayer(mvi, {}, 'mvi');
```

Langkah 6

- // Step 6. Thresholding.
- // Tweak these values accordingly

- // Lower threshold
- var lower = 2;
- // Upper threshold
- var upper = 20;

- var mviina = mvi.lt(upper).add(mvi.gt(lower));

- Map.addLayer(mviina, {}, 'mviina');

Langkah 7 - 8

```
// Step 7. Masking for non mangrove

var mask = mviina.eq(2);
var maskedmvi = mviina.updateMask(mask).rename('mangrove');

// Display the final product
Map.addLayer(maskedmvi, {}, 'maskedmvi');

// Step 8. Export to drive
Export.image.toDrive({
  image: maskedmvi,
  description: 'maskedmvi',
  maxPixels: 1e11,
  scale: 20,
  region: aoi
});
```

```
// Step 4. Masking before analysis
// Masking for pixel above 50 m
var srtm = ee.Image('USGS/SRTMGL1_003');
var elevation = srtm.select('elevation');
var masksrtm = composite.lt(50);
var maskedsrtm = composite.updateMask(masksrtm);

// Water masking
var hansenImage = ee.Image('UMD/hansen/global_forest_change_2015');
var datamask = hansenImage.select('datamask');
var maskland = datamask.eq(1);
var maskedcomposite = maskedsrtm.updateMask(maskland);

Map.addLayer(maskedcomposite, imageVisParam, 'composite');

var lcomposite = maskedcomposite.clip(aoi);
```

Remothon

Safran Yusri