

BADAR MUBAROK YOGASWARA

DAFTAR ISI

PENDAHULUAN: Paradigma Baru QGIS untuk Analisis Spasial	1
FOCUSING: 3 Senjata Utama QGIS Modern	1
Tiga Perangkat dalam Satu Buku	1
Cara Membaca Buku Ini	1
Lisensi Perangkat dan Data	2
SEBELUM MULAI: Persiapan Perangkat dan Data	3
Untuk siapa dan bagaimana membaca buku ini	3
Perangkat lunak dan plugin	3
Kebutuhan komputer dan koneksi	4
Data latihan	4
Susunan folder yang disarankan	5
BAB 1: Fondasi Modern & Manajemen Data Cerdas	7
Bagian A: QGIS	7
1.1. Menarik Citra Satelit via STAC API (Tanpa Download)	7
1.2. Standardisasi Sistem Proyeksi (Reprojecting)	8
1.3. Memuat Data Latar Belakang Cepat (XYZ Tiles)	9
1.4. Eksekusi Tingkat Lanjut: Otomatisasi dengan PyQGIS 3	9
Bagian B: ArcGIS Pro	13
Bagian C: ArcMap 10.8	16
Ringkasan, Latihan, dan Bantuan Bab 1	19
Tabel Perbandingan Bab 1: QGIS, ArcGIS Pro, dan ArcMap 10.8	19
BAB 2: Sinkronisasi Lapangan (Ground-Truthing Cloud)	21
Bagian A: QGIS	21
2.1. Membangun Basis Data & Formulir Survei Cerdas di QGIS	21
2.2. Sinkronisasi ke Perangkat Seluler (QField/Mergin Maps)	22
2.3. Eksekusi Tingkat Lanjut: Validasi Otomatis via PyQGIS	22
Bagian B: ArcGIS Pro	24
Bagian C: ArcMap 10.8	28
Ringkasan, Latihan, dan Bantuan Bab 2	31
Tabel Perbandingan Bab 2: QGIS, ArcGIS Pro, dan ArcMap 10.8	31
BAB 3: Pertanian Presisi & Analisis Citra Drone (UAV)	34
Bagian A: QGIS	34
3.1. Mengekstrak Canopy Height Model (CHM)	34
3.2. Menghitung Jumlah Tegakan Pohon (Local Maxima)	34
3.3. Eksekusi Tingkat Lanjut: Pemrosesan Latar Belakang (QgsTask) untuk Ratusan Citra Drone	36
Bagian B: ArcGIS Pro	39
Bagian C: ArcMap 10.8	42
Ringkasan, Latihan, dan Bantuan Bab 3	45
Tabel Perbandingan Bab 3: QGIS, ArcGIS Pro, dan ArcMap 10.8	45
BAB 4: Kehutanan Lanjut – Analisis Point Cloud (LiDAR) & DTM Ekstraksi	47
Bagian A: QGIS	47
4.1. Memuat Point Cloud Secara Native (Tanpa Plugin Pihak Ketiga)	47
4.2. Memfilter Kanopi untuk Mendapatkan Bare Earth (Tanah Murni)	47

4.3. Ekspor Point Cloud Tanah menjadi DTM (Raster).....	48
Bagian B: ArcGIS Pro.....	49
Bagian C: ArcMap 10.8.....	51
Ringkasan, Latihan, dan Bantuan Bab 4.....	53
Tabel Perbandingan Bab 4: QGIS, ArcGIS Pro, dan ArcMap 10.8.....	53
BAB 5: Klasifikasi Berbasis Objek (OBIA) & Machine Learning.....	55
Bagian A: QGIS.....	55
5.1. Segmentasi Objek (Membuat Poligon Alami).....	55
5.2. Ekstraksi Fitur Objek (Zonal Statistics).....	55
5.3. Melatih Machine Learning (Random Forest).....	56
Bagian B: ArcGIS Pro.....	57
Bagian C: ArcMap 10.8.....	59
Ringkasan, Latihan, dan Bantuan Bab 5.....	60
Tabel Perbandingan Bab 5: QGIS, ArcGIS Pro, dan ArcMap 10.8.....	60
BAB 6: Otomatisasi Alur Kerja & Atlas Dinamis Tingkat Lanjut.....	62
Glosarium.....	72
Daftar Pustaka.....	76
Aplikasi Live Penulis.....	79
Bagian A: QGIS.....	62
6.1. Graphical Modeler dengan Percabangan Logika.....	62
6.2. Print Layout & Ekspor Laporan Otomatis (Atlas).....	62
6.3. Teks Dinamis Berbasis Expression Builder (Inti Laporan).....	63
6.4. Ekspor Massal Laporan Sektoral.....	63
Bagian B: ArcGIS Pro.....	64
Bagian C: ArcMap 10.8.....	67
Ringkasan, Latihan, dan Bantuan Bab 6.....	70
Tabel Perbandingan Bab 6: QGIS, ArcGIS Pro, dan ArcMap 10.8.....	70

PENDAHULUAN: Paradigma Baru QGIS untuk Analis Spasial

Selama bertahun-tahun, praktisi GIS di bidang kehutanan dan pertanian terjebak pada alur kerja yang lambat: mengunduh citra satelit berukuran *Gigabyte*, melakukan *layer stacking* manual, dan menunggu berjam-jam untuk proses pemotongan (*clipping*). Di era QGIS 4.x, pendekatan tersebut sudah usang.

Buku ini dirancang untuk analis, rimbawan, dan praktisi agronomi yang ingin bekerja cepat, akurat, dan terotomatisasi. Anda akan berfokus pada tiga pilar utama:

FOCUSING: 3 Senjata Utama QGIS Modern

1. **STAC & COG (Cloud-Optimized GeoTIFF)**: Anda tidak lagi mengunduh citra fisik ke *hard-drive*. Citra Sentinel/Landsat dipanggil langsung dari server *Cloud* dan dirender hanya pada area yang Anda *zoom* di layar.
2. **Native C++ Algorithms**: Anda meminimalkan penggunaan plugin pihak ketiga yang rawan *error* (seperti SAGA GIS lawas) dan bergeser sepenuhnya ke algoritma *Native QGIS* atau GDAL yang jauh lebih cepat.
3. **PyQGIS 3 (Python 3)**: Konsol Python bawaan QGIS kini sangat tangguh untuk memproses ratusan tugas berulang di latar belakang (*background thread*).

Tiga Perangkat dalam Satu Buku

Buku ini membahas satu rangkaian pekerjaan analisis spasial dengan tiga perangkat lunak: **QGIS**, **ArcGIS Pro**, dan **ArcMap 10.8**. Bab-babnya sama untuk ketiganya, sehingga Anda dapat membandingkan cara kerja tiap perangkat pada masalah yang sama.

QGIS adalah perangkat sumber terbuka dan menjadi acuan utama buku ini. ArcGIS Pro adalah perangkat berlisensi dari Esri yang masih dikembangkan. ArcMap 10.8 adalah versi terakhir ArcGIS Desktop; versi 10.8.2 dilaporkan telah pensiun pada 1 Maret 2026, sehingga tidak ada lagi pembaruan dan dukungan teknis. **[CEK: periksa pengumuman resmi Esri]**

Cara Membaca Buku Ini

- Setiap bab terdiri dari tiga bagian berurutan: **Bagian A** (QGIS), **Bagian B** (ArcGIS Pro), dan **Bagian C** (ArcMap 10.8). Anda boleh langsung membaca bagian untuk perangkat yang Anda pakai.
- Tabel perbandingan di akhir tiap bab merangkum alat yang setara di ketiga perangkat dan seberapa pasti padanannya.
- Bila suatu langkah tidak punya padanan di sebuah perangkat, hal itu ditulis terus terang, disertai alternatif bila ada.
- Tanda **[CEK]** berarti langkah, alamat, atau kode itu perlu Anda cocokkan dengan perangkat Anda sebelum dipakai.
- Kotak berwarna menandai jenis catatan: biru untuk konteks serta perangkat dan lisensi, hijau untuk tips dan cara membaca, kuning untuk peringatan, catatan lisensi, dan pembahasan kode.

Lisensi Perangkat dan Data

Sebagian alat ArcGIS memerlukan ekstensi berlisensi, misalnya **Spatial Analyst**, **3D Analyst**, atau **Image Analyst**. Kebutuhan itu disebut pada tiap bagian. QGIS bebas biaya, tetapi data dan peta dasar yang dipakai di buku ini berasal dari pihak ketiga dengan syarat lisensi masing-masing. Pembaca bertanggung jawab memastikan penggunaan perangkat dan data sesuai lisensinya.

SEBELUM MULAI: Persiapan Perangkat dan Data

Bagian ini membantu Anda menyiapkan semuanya agar tiap bab bisa langsung diikuti. Luangkan waktu di awal supaya tidak tersendat saat langkah di bab-bab berikutnya.

Untuk siapa dan bagaimana membaca buku ini

Buku ini punya dua jalur. Bagian **Cara Visual** memakai menu dan klik, cocok bagi analis yang belum memprogram. Bagian **Eksekusi Tingkat Lanjut** memakai Python (PyQGIS di QGIS, ArcPy di ArcGIS Pro dan ArcMap) untuk mengotomatiskan pekerjaan yang sama, cocok bila Anda sudah akrab dengan Python atau ingin mulai belajar. Anda boleh mengikuti jalur visual saja, lalu kembali ke jalur lanjut belakangan.

Kotak berwarna menandai jenis catatan. Biru (**Konteks Riset/Bisnis, Perangkat dan Lisensi**) menjelaskan masalah nyata yang diselesaikan di bab itu dan kebutuhan perangkatnya. Hijau (**TIPS**) berisi kebiasaan kerja yang disarankan. Kuning (**BEDAH KODE / BAGAIMANA INI BEKERJA, PERHATIAN**, dan catatan lisensi) menjelaskan cara kerja kode, peringatan, dan batasan.

Tiap bab disusun berurutan untuk tiga perangkat: **Bagian A** (QGIS), **Bagian B** (ArcGIS Pro), dan **Bagian C** (ArcMap 10.8). Anda boleh langsung membaca bagian untuk perangkat yang Anda pakai.

Perangkat lunak dan plugin

Berikut yang perlu terpasang, beserta bab yang memerlukannya.

- **QGIS 4.x.** Python Console sudah ada di dalam QGIS (Bab 1, 2, dan 3).
- **Plugin STAC API Browser**, dipasang lewat Plugins > Manage and Install Plugins (Bab 1).
- **QFieldSync** (atau Mergin Maps) dan aplikasi **QField** di ponsel (Bab 2). Layanan **QFieldCloud** memiliki paket gratis dengan batas penyimpanan dan paket berbayar; periksa ketentuan dan harga terbaru di situs resmi QField.
- **Alat point cloud (PDAL)** di Processing Toolbox (Bab 4). Alat ini memerlukan PDAL versi 2.5.0 atau lebih baru, dan alat Classify ground points tersedia mulai QGIS 4.0.
- **Orfeo Toolbox (OTB)** (Bab 5). OTB dipasang terpisah dari QGIS, lalu diatur di Settings > Options > Processing > Providers > OTB: isi **OTB folder** (akar instalasi OTB) dan **OTB application folder** (biasanya <OTB_FOLDER>/lib/otb/applications). Pada QGIS terbaru penyedia OTB tersedia lewat katalog plugin. Nama plugin yang persis dan kecocokannya dengan QGIS 4.x belum diverifikasi **[CEK]**. Tanpa pengaturan ini, menu OTB tidak muncul di Processing Toolbox.
- **ArcGIS Pro** memakai Python 3. Bab 1, 3, dan 5 memerlukan ekstensi Spatial Analyst atau Image Analyst, dan Bab 4 memerlukan 3D Analyst. Bab 2 memerlukan akun ArcGIS Online atau Enterprise untuk Field Maps. **[CEK]**
- **ArcMap 10.8** memakai Python 2.7. Bab 1, 3, dan 5 memerlukan Spatial Analyst, dan Bab 4 memerlukan 3D Analyst. Versi 10.8.2 dilaporkan telah pensiun pada 1 Maret 2026. **[CEK: periksa pengumuman resmi Esri]**

Tabel berikut merangkum kebutuhan khusus per bab.

Bab	QGIS	ArcGIS Pro	ArcMap 10.8
1	Python Console, plugin STAC API Browser, koneksi internet	Spatial Analyst atau Image Analyst; STAC bawaan	Spatial Analyst; tanpa STAC (image service)
2	QFieldSync atau Mergin Maps, aplikasi QField	Field Maps, akun ArcGIS Online atau Enterprise	Alur lewat ArcGIS Online dan Field Maps [CEK]; Enable Attachments butuh lisensi Standard atau Advanced
3	Bawaan QGIS	Spatial Analyst atau Image Analyst	Spatial Analyst
4	Alat point cloud PDAL (PDAL 2.5.0+)	3D Analyst; LAS Dataset To Raster juga bisa dengan Spatial Analyst	3D Analyst
5	Orfeo Toolbox (OTB)	Image Analyst atau Spatial Analyst	Spatial Analyst
6	Bawaan QGIS	Intersect dibatasi dua masukan pada lisensi Basic dan Standard	Intersect terbatas pada lisensi Basic dan Standard

Kebutuhan komputer dan koneksi

- Koneksi internet yang stabil untuk Bab 1, karena citra dibaca langsung dari cloud. Bila koneksi putus, layer gagal dimuat.
- Ruang penyimpanan yang cukup untuk Bab 3 dan Bab 4, karena ortofoto drone dan point cloud berukuran besar.
- RAM dan prosesor yang memadai. Spesifikasi di atas adalah perkiraan; data drone dan point cloud berukuran besar akan terasa lambat pada komputer dengan RAM kecil.

Data latihan

Agar pembaca bisa mengulang setiap langkah, buku ini memakai data contoh yang kecil. Nama layer dan field di bawah ini disesuaikan dengan yang dipakai pada skrip dan langkah di buku:

- Bab 1: batas area kajian bernama Batas_KPH.
- Bab 2: layer poligon klasifikasi bernama Poligon_Sawah dengan field Kelas_Citra, serta beberapa titik survei contoh (Titik_Survei dengan field Kondisi, Tinggi_Phnl, dan Foto).
- Bab 3: potongan kecil DSM dan DTM drone (DSM_Drone, DTM_Drone) serta beberapa ubin ortofoto.
- Bab 4: satu berkas point cloud (.laz) berukuran kecil.
- Bab 5: potongan citra komposit untuk segmentasi.
- Bab 6: layer Batas_KPH dengan field NAMA_KPH, dan layer titik panas Hotspot_Nasional dengan field ID_Titik.

Paket data latihan dan **semua skrip Python** dalam buku ini disediakan di **versi web buku**, sehingga skrip dapat disalin dengan mudah. Paket data berisi data **sintetis** (buatan), dengan nama layer dan field persis seperti di atas. Data sintetis tidak memiliki persoalan hak dan kerahasiaan lokasi, tetapi

hasil analisisnya bukan kondisi lapangan nyata. Sistem koordinat paket data adalah WGS 84 / UTM zone 49S (EPSG:32749).

Skrip QGIS dalam buku ini telah diuji di QGIS 4.0.2 dengan paket data sintetis tersebut. Skrip ArcPy untuk ArcGIS Pro dan ArcMap perlu Anda cocokkan dan ditandai **[CEK]**. Pembaca yang memakai QGIS versi lain disarankan mencoba skrip pada data contoh lebih dulu.

Susunan folder yang disarankan

STRUKTUR FOLDER

```
Masterclass_GIS/  
|-- 00_data_mentah/    (data asli, jangan diubah)  
|-- 01_proyek/         (berkas proyek: .qgz, .aprx, atau .mxd)  
|-- 02_hasil/          (keluaran analisis)  
|-- 03_skrip/          (skrip Python)
```

Skrip di Bab 3 memakai jalur contoh D:/Data_Drone/. Sesuaikan jalur itu dengan susunan folder Anda.

Hak cipta dan penggunaan

Hak cipta © 2026 Badar Mubarak Yogaswara. All Rights Reserved.

Hak cipta dilindungi undang-undang. Dilarang mengutip atau memperbanyak sebagian atau seluruh isi buku ini tanpa izin tertulis dari penulis. Untuk penggunaan lain, misalnya bahan ajar berbayar atau penerbitan ulang, hubungi penulis di badarmubarak@gmail.com.

Data latihan di buku ini buatan (sintetis), bukan data lapangan. Nama produk dan merek, misalnya QGIS dan ArcGIS, milik pemiliknya masing-masing.

Unduhan: buku dan paket data

Buku ini dan semua berkas latihannya tersedia gratis di situs penulis. Alamat lengkapnya tertulis di bawah, jadi bisa Anda ketik langsung bila membaca versi cetak.

Berkas	Tautan	Isi
Halaman unduhan M1	https://badarmubarak.id/buku-gis/m1/unduhannya.html	semua berkas seri ini
Buku (PDF)	https://badarmubarak.id/buku-gis/m1/unduhannya/BUKU-M1.pdf	buku ini dalam PDF
Paket data latihan	https://badarmubarak.id/buku-gis/m1/unduhannya/data-sintetis-m1.zip	data sintetis latihan
Skrip (zip)	https://badarmubarak.id/buku-gis/m1/unduhannya/skrip-m1.zip	skrip Python dan lainnya
Seluruh seri	https://badarmubarak.id/buku-gis/	12 buku Basic sampai Master

Pindai kode QR ini dengan kamera ponsel untuk membuka halaman unduhan M1.



BAB 1: Fondasi Modern & Manajemen Data Cerdas

Konteks Riset/Bisnis: Anda adalah Analis GIS di sebuah perusahaan Hutan Tanaman Industri (HTI) atau perkebunan sawit. Anda diminta menyajikan pantauan tutupan lahan terbaru untuk area konsesi seluas 50.000 Hektar. Tenggat waktu sangat ketat. Mengunduh data mentah Sentinel-2 dari portal Copernicus akan memakan waktu berjam-jam hanya untuk satu tanggal perekaman. Anda akan menyelesaikannya dalam 3 menit.

CARA MEMBACA BAB INI: Bab ini memuat tiga bagian berurutan. **Bagian A** untuk QGIS, **Bagian B** untuk ArcGIS Pro, dan **Bagian C** untuk ArcMap 10.8. Tabel perbandingan ada di akhir bab. Tanda **[CEK]** berarti langkah itu perlu Anda cocokkan dengan perangkat Anda dan harus dicoba lebih dulu sebelum dipakai. Bagian yang tidak punya padanan ditulis terus terang.

BAGIAN A | QGIS

1.1. Menarik Citra Satelit via STAC API (Tanpa Download)

Spatiotemporal Asset Catalog (STAC) adalah standar modern pencarian data spasial. QGIS 4.x dapat membaca format ini dengan sangat baik.

A. Langkah Eksekusi (Cara Visual):

1. Buka QGIS. Pastikan Anda telah memasang plugin **STAC API Browser** (cari di *Plugins > Manage and Install Plugins*).
2. Tampilkan batas konsesi atau area kajian Anda di kanvas peta.
3. Buka ikon **STAC API Browser** di *toolbar*.

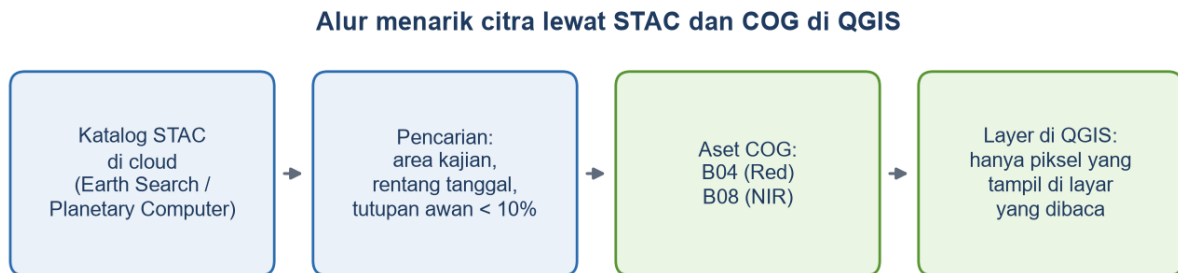
Ilustrasi 1.1: Antarmuka STAC API Browser

```
+-----+
| STAC API BROWSER |
+-----+
| Connection : [ Earth Search (AWS) / Microsoft Planetary] |
| Collection : [ Sentinel-2 Level-2A (Surface Reflectance)] |
| Extent      : [x] Fetch from Map Canvas Extent           |
| Date Range  : 2026-01-01 TO 2026-03-31                 |
| Cloud Cover: < 10%                                     |
|                                                       |
| [ SEARCH ] ---> Ditemukan: 4 Scene.                     |
|                                                       |
| Aset Tersedia:                                         |
| [x] B04 (Red)   [ Add as Cloud Optimized GeoTIFF (COG)] |
| [x] B08 (NIR)   [ Add as Cloud Optimized GeoTIFF (COG)] |
+-----+
```

1. Klik **Add as COG** untuk Band 4 (Merah) dan Band 8 (Inframerah Dekat).

2. Citra resolusi 10m langsung muncul di layar Anda. Anda hanya men-*streaming* piksel yang tampil di layar, tanpa membebani penyimpanan komputer Anda.

Ilustrasi 1.2: Alur menarik citra lewat STAC dan COG di QGIS



Tidak ada berkas citra utuh yang diunduh ke hard-drive.

1.2. Standardisasi Sistem Proyeksi (Reprojecting)

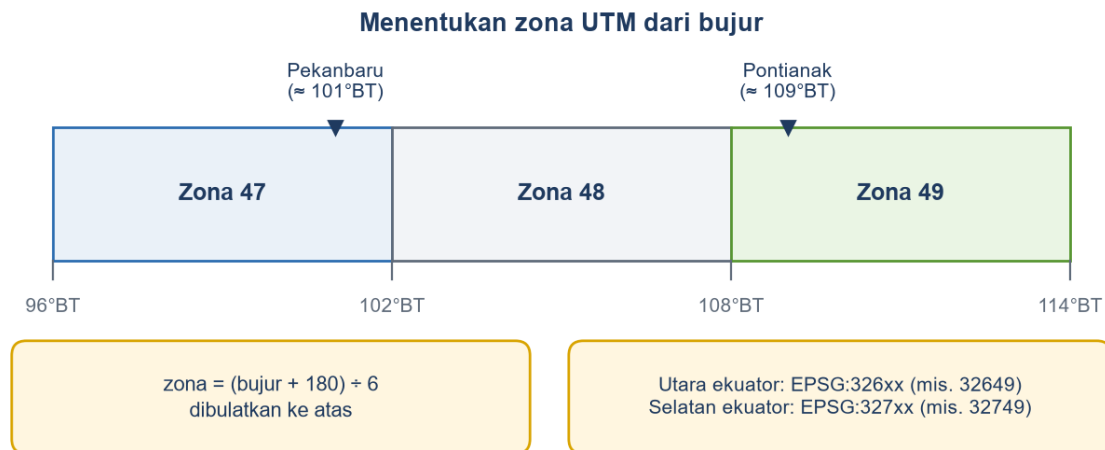
Agar analisis spasial (seperti perhitungan luas dalam Hektar) akurat, data tidak boleh berada dalam sistem derajat/geografis (WGS 84 / EPSG:4326). Pastikan kanvas proyek memakai sistem metrik/proyeksi seperti Universal Transverse Mercator (UTM).

A. Langkah Eksekusi:

1. Klik kode EPSG di sudut kanan bawah antarmuka QGIS (misal: EPSG:4326).
2. Jendela **Project Properties - CRS** akan terbuka.
3. Cari sistem UTM lokal Anda. (*Contoh: Pekanbaru, Riau, berada di sekitar bujur 101° BT sehingga masuk zona 47; Pontianak, Kalimantan Barat, di sekitar bujur 109° BT sehingga masuk zona 49. Nomor zona dihitung dari bujur: (bujur + 180) ÷ 6, dibulatkan ke atas. Pilih belahan Utara (N) untuk area di utara ekuator dan belahan Selatan (S) untuk area di selatan ekuator. Misalnya WGS 84 / UTM zone 49N = EPSG:32649, dan WGS 84 / UTM zone 49S = EPSG:32749.*)
4. Klik **Apply**.

TIPS LINGKUNGAN KERJA: Biasakan mengubah CRS Proyek di awal. Di QGIS 4.x, fitur *On-the-fly (OTF) Projection* secara otomatis menyelaraskan visualisasi layer STAC/COG Anda ke dalam ukuran meter secara presisi tanpa perlu membuat file *reprojected* baru di *hard-drive*.

Ilustrasi 1.3: Menentukan zona UTM dari bujur

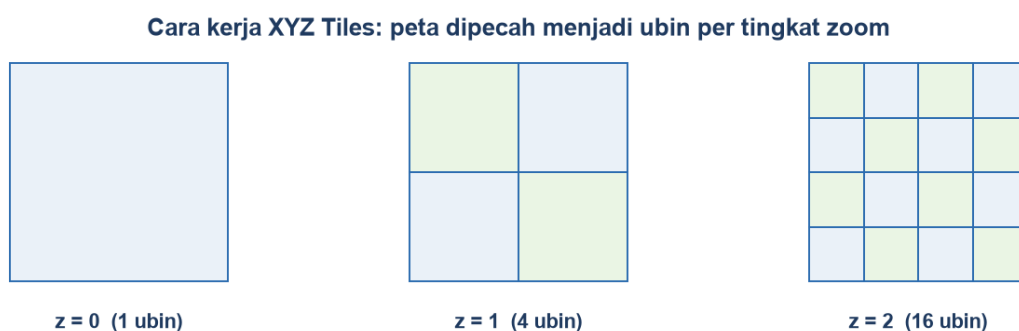


1.3. Memuat Data Latar Belakang Cepat (XYZ Tiles)

Lupakan plugin berat untuk memuat peta dasar. QGIS 4.x sudah memiliki fitur *native* di panel *Browser*.

1. Buka panel **Browser** di sisi kiri.
2. Cari menu **XYZ Tiles**.
3. Klik kanan > **New Connection**.
4. Masukkan nama: EOX Sentinel-2 Cloudless.
5. Masukkan URL: https://tiles.maps.eox.at/wmts/1.0.0/s2cloudless-2024_3857/default/g/{z}/{y}/{x}.jpg **[CEK]**
6. Seret EOX Sentinel-2 Cloudless ke panel layer. Ini sangat berguna untuk validasi batas area tebangan atau kebun secara visual. Cantumkan atribusi EOX pada hasil kerja Anda (lihat catatan lisensi di akhir bagian ini).

Ilustrasi 1.4: Cara kerja XYZ Tiles



`https://.../{z}/{x}/{y}.png` z = zoom x = kolom y = baris

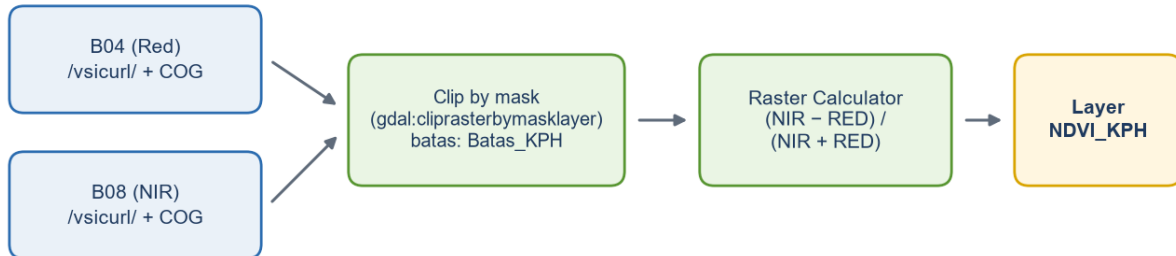
1.4. Eksekusi Tingkat Lanjut: Otomatisasi dengan PyQGIS 3

Bagi analis tingkat mahir, mengklik menu berulang kali adalah pemborosan waktu. Anda bisa memakai *Python Console* (PyQGIS) untuk memuat layer STAC, mengatur simbologi NDVI, dan memotong citra secara otomatis.

Skenario Python: Anda ingin menghitung *Normalized Difference Vegetation Index (NDVI)* secara *on-the-fly* dari URL citra COG di *cloud*, lalu memotongnya sesuai batas Vektor Area Kajian.

Ilustrasi 1.5: Alur Skrip 1.1 di QGIS

Alur Skrip 1.1 di QGIS: dari citra cloud ke NDVI



Hanya piksel di dalam batas yang ditarik dari server.

Buka **Plugins > Python Console** dan jalankan skrip berikut:

PYTHON

```

# [SKRIP 1.1: Memuat COG & Eksekusi NDVI Virtual di QGIS 4.x]
# Penulis: Badar Mubarok Yogaswara
# Syarat: Pastikan layer batas area (misal: 'Batas_KPH') aktif di QGIS.

import processing
from qgis.core import QgsProject, QgsRasterLayer

print("🔗 Memulai koneksi ke Cloud-Optimized GeoTIFF (COG)...")

# 1. Tentukan URL COG dari STAC API (Contoh URL dummy Sentinel-2)
url_b4 = "/vsicurl/https://sentinel-cogs.s3.us-west-2.amazonaws.com/sentinel-s2-l2a-cogs/2026/T48M/B04.tif"
url_b8 = "/vsicurl/https://sentinel-cogs.s3.us-west-2.amazonaws.com/sentinel-s2-l2a-cogs/2026/T48M/B08.tif"

# 2. Muat Layer secara Virtual
layer_b4 = QgsRasterLayer(url_b4, "B04_Red_Cloud", "gdal")
layer_b8 = QgsRasterLayer(url_b8, "B08_NIR_Cloud", "gdal")

if not layer_b4.isValid() or not layer_b8.isValid():
    print("❌ Gagal memuat citra dari Cloud. Cek koneksi internet.")
else:
    QgsProject.instance().addMapLayers([layer_b4, layer_b8])
    print("✅ Citra Red dan NIR berhasil dimuat.")

# 3. Ambil layer Vektor untuk batas pemotongan
batas_kph = QgsProject.instance().mapLayersByName('Batas_KPH')[0]

# 4. Potong (Clip) Citra langsung dari Cloud menggunakan Algoritma Native GDAL
print("✂️ Memotong citra berdasarkan batas KPH...")

# Clip Band 8 (NIR)
clip_b8 = processing.run("gdal:cliprasterbymasklayer", {
    'INPUT': layer_b8,
    'MASK': batas_kph,
    'CROP_TO_CUTLINE': True,
    'OUTPUT': 'TEMPORARY_OUTPUT'
})['OUTPUT']

# Clip Band 4 (Red)
clip_b4 = processing.run("gdal:cliprasterbymasklayer", {

```

```

        'INPUT': layer_b4,
        'MASK': batas_kph,
        'CROP_TO_OUTLINE': True,
        'OUTPUT': 'TEMPORARY_OUTPUT'
    })['OUTPUT']

# 5. Kalkulasi NDVI Menggunakan Raster Calculator Native
# Rumus: (NIR - RED) / (NIR + RED)
print("🇮🇩 Menghitung NDVI...")

ndvi_result = processing.run("qgis:rastercalculator", {
    'EXPRESSION': '("B08@1" - "B04@1") / ("B08@1" + "B04@1")',
    'LAYERS': [clip_b8, clip_b4],
    'CELLSIZE': 10,
    'OUTPUT': 'memory:NDVI_KPH'
})

QgsProject.instance().addMapLayer(ndvi_result['OUTPUT'])
print("🚀 Selesai! Layer NDVI_KPH telah ditambahkan ke kanvas.")

```

BEDAH KODE (Bagi Praktisi Python): Perhatikan awalan /vsicurl/ pada URL. Ini adalah teknologi **VSI (Virtual Systems Interface)** milik GDAL. Alih-alih mengunduh citra 1 GB, GDAL hanya "mengintip" *header* file di server AWS, lalu fungsi `gdal:cliprasterbymasklayer` hanya akan menarik piksel yang berada pas di dalam poligon Batas_KPH. Efisiensi komputasi meningkat hingga 90% dibandingkan metode pengunduhan konvensional.

CATATAN LISENSI PETA DASAR: Citra pada langkah 1.3 berasal dari EOX (Sentinel-2 cloudless). Lisensinya CC BY-NC-SA 4.0 untuk pemakaian non-komersial; pemakaian komersial memerlukan lisensi terpisah dari EOX IT Services GmbH. Atribusi wajib terlihat di mana pun citra ditampilkan: "EOXCloudless <https://cloudless.eox.at> by EOX IT Services GmbH (Contains modified Copernicus Sentinel data 2024)". Pembaca bertanggung jawab memastikan penggunaannya sesuai lisensi. Alamat tile pada langkah 1.3 **[CEK]**: periksa di dokumentasi EOX bila alamat berubah.

HASIL UJI SKRIP 1.1: Skrip 1.1 diuji di QGIS 4.0.2 dengan data sintetis (URL cloud pada contoh adalah URL dummy, jadi uji memakai berkas lokal sebagai pengganti). Langkah memuat dan memotong berjalan, tetapi dua baris terakhir gagal: keluaran raster ke 'memory:' mengembalikan teks, bukan layer, sehingga `addMapLayer()` menolaknya. Selain itu, dokumentasi QGIS hanya menyebut `native:rastercalc`, bukan `qgis:rastercalculator`. Alternatif berikut sudah diuji: layer NDVI terbentuk dan nilainya sama dengan hitungan langsung pada delapan titik uji.

Skrip alternatif (memakai `native:rastercalc`):

PYTHON (ALTERNATIF)

```

# [SKRIP 1.1 - ALTERNATIF: Memuat COG & Eksekusi NDVI di QGIS 4.x]
# Penulis: Badar Mubarak Yogaswara
# Syarat: Pastikan layer batas area (misal: 'Batas_KPH') aktif di QGIS.
# Perbedaan dengan Skrip 1.1: langkah 5 memakai native:rastercalc (terdokumentasi) dan hasilnya
# disimpan ke berkas sementara lalu dimuat sebagai layer. Keluaran 'memory:' tidak menghasilkan layer raster.

import processing
from qgis.core import QgsProject, QgsRasterLayer

```

```

print("🔗 Memulai koneksi ke Cloud-Optimized GeoTIFF (COG)...")

# 1. Tentukan URL COG dari STAC API (Contoh URL dummy Sentinel-2)
url_b4 = "/vsicurl/https://sentinel-cogs.s3.us-west-2.amazonaws.com/sentinel-s2-l2a-cogs/2026/T48M/B04.tif"
url_b8 = "/vsicurl/https://sentinel-cogs.s3.us-west-2.amazonaws.com/sentinel-s2-l2a-cogs/2026/T48M/B08.tif"

# 2. Muat Layer secara Virtual
layer_b4 = QgsRasterLayer(url_b4, "B04_Red_Cloud", "gdal")
layer_b8 = QgsRasterLayer(url_b8, "B08_NIR_Cloud", "gdal")

if not layer_b4.isValid() or not layer_b8.isValid():
    print("❌ Gagal memuat citra dari Cloud. Cek koneksi internet.")
else:
    QgsProject.instance().addMapLayers([layer_b4, layer_b8])
    print("✅ Citra Red dan NIR berhasil dimuat.")

# 3. Ambil layer Vektor untuk batas pemotongan
batas_kph = QgsProject.instance().mapLayersByName('Batas_KPH')[0]

# 4. Potong (Clip) Citra langsung dari Cloud menggunakan Algoritma Native GDAL
print("✂️ Memotong citra berdasarkan batas KPH...")

# Clip Band 8 (NIR)
clip_b8 = processing.run("gdal:cliprasterbymasklayer", {
    'INPUT': layer_b8,
    'MASK': batas_kph,
    'CROP_TO_OUTLINE': True,
    'OUTPUT': 'TEMPORARY_OUTPUT'
})['OUTPUT']

# Clip Band 4 (Red)
clip_b4 = processing.run("gdal:cliprasterbymasklayer", {
    'INPUT': layer_b4,
    'MASK': batas_kph,
    'CROP_TO_OUTLINE': True,
    'OUTPUT': 'TEMPORARY_OUTPUT'
})['OUTPUT']

# 5. Kalkulasi NDVI Menggunakan Raster Calculator Native
# Rumus: (NIR - RED) / (NIR + RED)
print("📊 Menghitung NDVI...")

# Beri nama pada hasil potongan agar bisa dirujuk di ekspresi sebagai "B08@1" dan "B04@1"
lyr_b8 = QgsRasterLayer(clip_b8, "B08", "gdal")
lyr_b4 = QgsRasterLayer(clip_b4, "B04", "gdal")
QgsProject.instance().addMapLayers([lyr_b8, lyr_b4], False)

ndvi_path = processing.run("native:rastercalc", {
    'EXPRESSION': '("B08@1" - "B04@1") / ("B08@1" + "B04@1")',
    'LAYERS': [lyr_b8, lyr_b4],
    'OUTPUT': 'TEMPORARY_OUTPUT'
})['OUTPUT']

QgsProject.instance().addMapLayer(QgsRasterLayer(ndvi_path, "NDVI_KPH", "gdal"))
print("🚀 Selesai! Layer NDVI_KPH telah ditambahkan ke kanvas.")

```

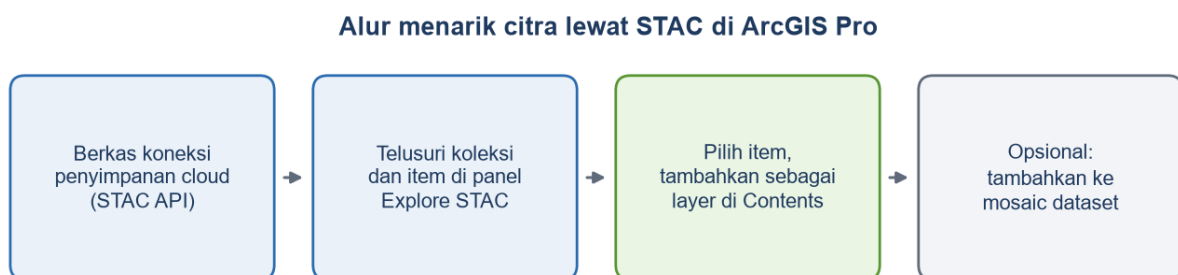
BAGIAN B | ArcGIS Pro

PERANGKAT DAN LISENSI: ArcGIS Pro memakai **Python 3**. Alat **Project** dan **Project Raster** tidak memerlukan ekstensi. Alat analisis raster seperti **Raster Calculator** memerlukan ekstensi **Spatial Analyst** atau **Image Analyst**.

1.1. Menarik Citra Satelit via STAC (ArcGIS Pro)

Di ArcGIS Pro, dukungan STAC sudah bawaan, jadi tidak perlu plugin seperti pada QGIS. Prinsipnya sama: Anda mencari citra dari katalog di cloud lalu menambahkannya ke peta.

Ilustrasi 1.6: Alur menarik citra lewat STAC di ArcGIS Pro



[UJI] Nama menu dan kemampuan memilih band satu per satu belum diverifikasi.

A. Langkah Eksekusi:

1. Buat koneksi ke **STAC API** memakai berkas koneksi penyimpanan cloud (*cloud storage connection*), seperti dijelaskan pada dokumentasi Esri "Introduction to STAC". **[CEK]**
2. Telusuri koleksi (*collections*) dan item pada panel **Explore STAC**. **[CEK]**
3. Pilih item yang Anda perlukan, lalu tambahkan sebagai layer ke peta (panel *Contents*). **[CEK]**
4. Opsional: tambahkan item ke *mosaic dataset* bila akan dianalisis lebih lanjut.

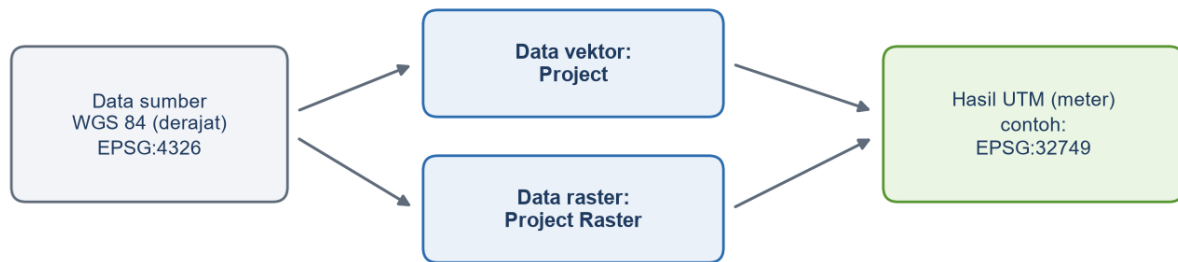
PERHATIAN [CEK]: Dokumentasi yang dirujuk tidak menyebutkan format COG, filter tutupan awan, atau persyaratan lisensi. Karena itu bagian ini belum memakai tangkapan layar asli. Letak menu dan kemampuan memilih band satu per satu harus Anda periksa sendiri di ArcGIS Pro.

1.2. Standardisasi Sistem Proyeksi (ArcGIS Pro)

Alasannya sama dengan Bagian A: perhitungan luas dan jarak yang akurat memerlukan sistem koordinat metrik seperti UTM, bukan derajat (WGS 84 / EPSG:4326).

Ilustrasi 1.7: Mengubah proyeksi data di ArcGIS Pro

Mengubah proyeksi data dengan Project dan Project Raster (ArcGIS Pro)



Project dan Project Raster tersedia pada semua level lisensi, tanpa ekstensi.

A. Langkah Eksekusi:

1. Buka properti peta (*Map Properties*) dan pilih tab sistem koordinat (*Coordinate Systems*). **[CEK]**
2. Pilih zona UTM sesuai lokasi. Nomor zona dihitung dari bujur: $(\text{bujur} + 180) \div 6$, dibulatkan ke atas. Contoh: WGS 84 / UTM zone 49S = EPSG:32749, dan WGS 84 / UTM zone 49N = EPSG:32649.
3. Klik OK.
4. Bila data fisiknya perlu diubah, buka panel **Geoprocessing**, lalu pakai **Project** untuk data vektor atau **Project Raster** untuk raster (kelompok *Data Management Tools*). Keduanya tersedia pada semua level lisensi tanpa ekstensi.

TIPS: Pada **Project Raster**, pilih metode resampling sesuai jenis data (kategorikal atau kontinu). Dokumentasi Esri menyebut galat umumnya di bawah setengah piksel.

1.3. Memuat Data Latar Belakang (ArcGIS Pro)

1. Buka galeri **Basemap** pada tab **Map**. **[CEK]**
2. Pilih peta dasar yang tersedia, lalu periksa syarat penggunaannya.

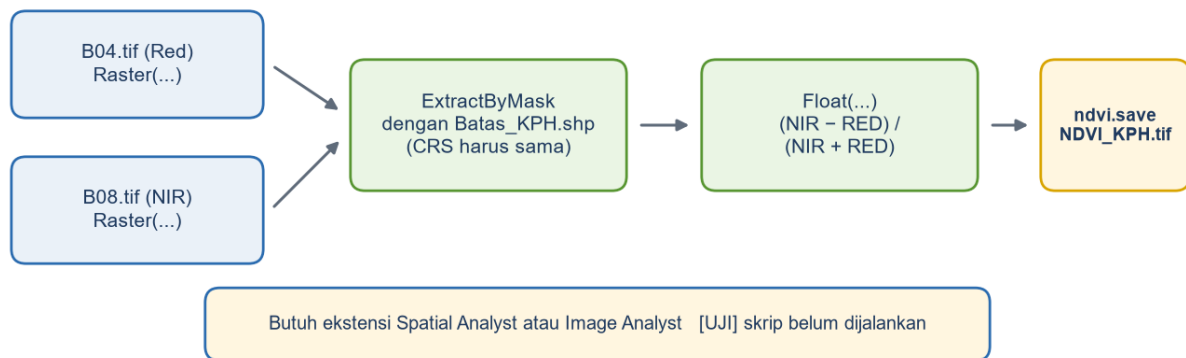
TIDAK ADA PADANAN YANG DIJAMIN: QGIS punya panel **XYZ Tiles** di Browser. Dukungan layer XYZ murni di ArcGIS Pro **belum terverifikasi**, sehingga langkah 1.3 Bagian A tidak bisa dipindahkan begitu saja. Uji dulu di perangkat Anda sebelum menjanjikannya kepada pembaca.

1.4. Eksekusi Tingkat Lanjut: Otomatisasi dengan ArcPy (Python 3)

Skenario sama dengan Bagian A: menghitung NDVI dari band Red dan NIR, setelah dipotong sesuai batas area kajian. Bedanya, skrip ini bekerja pada **berkas lokal**.

Ilustrasi 1.8: Alur skrip NDVI dengan ArcPy di ArcGIS Pro

Alur skrip NDVI dengan ArcPy (ArcGIS Pro, Python 3)



Buka jendela Python di ArcGIS Pro dan jalankan skrip berikut:

PYTHON (ARCPY, PYTHON 3)

```

# [SKRIP 1.1-Pro: Memotong citra dan menghitung NDVI dengan ArcPy]
# Penulis: Badar Mubarak Yogaswara
# [CEK] Cocokkan dengan versi Anda. Butuh ekstensi Spatial Analyst atau Image Analyst.
import arcpy
from arcpy.sa import Raster, Float, ExtractByMask

arcpy.CheckOutExtension("Spatial")
arcpy.env.overwriteOutput = True
arcpy.env.workspace = r"C:\data_kph" # ganti dengan folder Anda

# 1. Masukan: band Red, band NIR, dan batas area (CRS harus sama, lihat 1.2)
red = Raster("B04.tif")
nir = Raster("B08.tif")
batas = "Batas_KPH.shp"

# 2. Potong kedua band sesuai batas
red_clip = ExtractByMask(red, batas)
nir_clip = ExtractByMask(nir, batas)

# 3. NDVI = (NIR - RED) / (NIR + RED)
ndvi = (Float(nir_clip) - Float(red_clip)) / (Float(nir_clip) + Float(red_clip))
ndvi.save("NDVI_KPH.tif")

arcpy.CheckInExtension("Spatial")
print("Selesai: NDVI_KPH.tif tersimpan.")
  
```

BEDAH KODE (Bagi Praktisi Python): Skrip ini **tidak** memakai awalan `/vsicurl/` seperti Skrip 1.1 di Bagian A. Membaca citra COG langsung dari cloud dengan ArcPy **belum diverifikasi**, jadi unduh dulu kedua band atau muat lewat layer yang sudah ada di peta. `Float()` memastikan perhitungan memakai bilangan desimal. **[CEK]**

BAGIAN C | ArcMap 10.8

PERANGKAT DAN LISENSI: ArcMap memakai **Python 2.7**. ArcGIS Desktop 10.8.2 dilaporkan telah pensiun pada 1 Maret 2026. Akibatnya, tidak ada lagi pembaruan dan dukungan teknis.

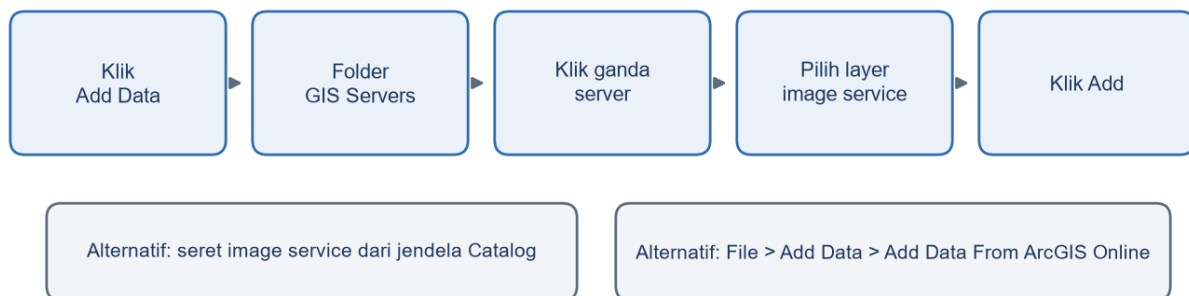
[CEK: periksa pengumuman resmi Esri]. Alat **Project** dan **Project Raster** tidak memerlukan ekstensi. Alat analisis raster memerlukan **Spatial Analyst** (dan **3D Analyst** untuk LiDAR).

1.1. Menarik Citra Satelit (ArcMap): Tidak Ada Padanan STAC

Dokumentasi ArcMap yang diperiksa penulis tidak memuat dukungan STAC. Alternatif terdekat adalah menambahkan **image service**. Ini citra yang disajikan server dan ditampilkan di peta tanpa menyalin seluruh datanya.

Ilustrasi 1.9: Menambahkan image service di ArcMap

Menambahkan image service di ArcMap



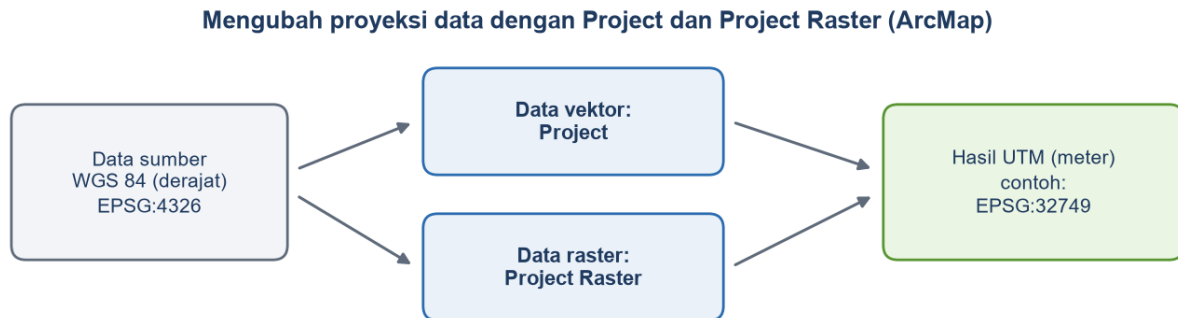
A. Langkah Eksekusi:

1. Klik tombol **Add Data**.
2. Pada daftar *Look in*, buka folder **GIS Servers**.
3. Klik ganda server yang Anda inginkan (atau tambahkan koneksi ArcGIS Server baru bila belum ada).
4. Pilih layer **image service** (ikonnya khas), lalu klik **Add**. Cara lain: seret image service dari jendela **Catalog**.
5. Alternatif: **File > Add Data > Add Data From ArcGIS Online** untuk mencari citra yang dibagikan komunitas.

PERHATIAN: Image service bukan pengganti penuh STAC dan COG. Anda memakai citra yang disediakan layanan itu, bukan memilih band Sentinel-2 satu per satu. Beberapa layanan bersifat *cached*. Anda bisa memakai *Identify*, tetapi tidak bisa mengubah properti seperti metode mosaik.

1.2. Standardisasi Sistem Proyeksi (ArcMap)

Ilustrasi 1.10: Mengubah proyeksi data di ArcMap



Project Raster: metode resampling Nearest, Bilinear, Cubic, Majority; galat kurang dari setengah piksel.

A. Langkah Eksekusi:

1. Buka properti *data frame* dan pilih tab sistem koordinat. **[CEK]**
2. Pilih zona UTM sesuai lokasi (rumus dan kode EPSG sama seperti Bagian B).
3. Bila data fisiknya perlu diubah, pakai **Project** (vektor) atau **Project Raster** (raster) di *Data Management*. Keduanya tersedia pada semua level lisensi tanpa ekstensi.

TIPS: Project Raster memberi jaminan galat kurang dari setengah piksel dan menyediakan metode resampling **Nearest, Bilinear, Cubic, dan Majority**. Untuk data kategorikal (misalnya kelas tutupan lahan), pilih Nearest atau Majority. Untuk data kontinu, pilih Bilinear atau Cubic.

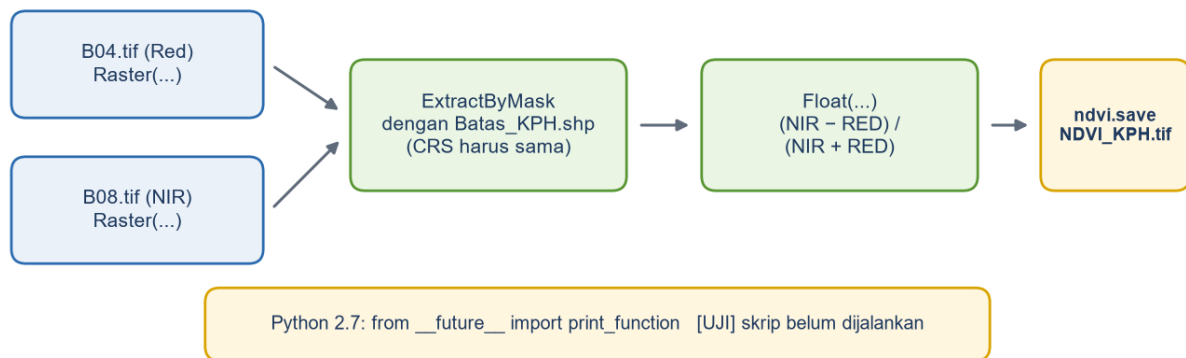
1.3. Memuat Data Latar Belakang (ArcMap)

1. Pilih **File > Add Data > Add Basemap**. **[CEK]**
2. Pilih peta dasar yang tersedia, lalu periksa syarat penggunaannya.

MASALAH YANG DIKETAHUI [CEK]: Dialog *Add Basemap* dapat gagal menambahkan peta dasar bila nama pengguna Windows terlalu panjang. Solusi sementara: buka folder C: \Users\<nama pengguna>\Documents\ArcGIS\Packages. Cari folder berekstensi .lpk dengan nama peta dasar itu, buka, lalu tambahkan berkas layer di dalamnya. Layer XYZ murni tidak ditemukan di ArcMap.

1.4. Eksekusi Tingkat Lanjut: Otomatisasi dengan ArcPy (Python 2.7)

Skenario dan hasilnya sama dengan Bagian B, tetapi kode harus mengikuti Python 2.7.

Ilustrasi 1.11: Alur skrip NDVI dengan ArcPy di ArcMap**Alur skrip NDVI dengan ArcPy (ArcMap, Python 2.7)**

Buka jendela Python di ArcMap dan jalankan skrip berikut:

PYTHON (ARCPY, PYTHON 2.7)

```

# [SKRIP 1.1-ArcMap: Memotong citra dan menghitung NDVI dengan ArcPy]
# Penulis: Badar Mubarak Yogaswara
# [CEK] Cocokkan dengan versi Anda. Butuh ekstensi Spatial Analyst.
from __future__ import print_function # agar print() berlaku di Python 2.7
import arcpy
from arcpy.sa import Raster, Float, ExtractByMask

arcpy.CheckOutExtension("Spatial")
arcpy.env.overwriteOutput = True
arcpy.env.workspace = r"C:\data_kph" # ganti dengan folder Anda

red = Raster("B04.tif")
nir = Raster("B08.tif")
batas = "Batas_KPH.shp"

red_clip = ExtractByMask(red, batas)
nir_clip = ExtractByMask(nir, batas)

ndvi = (Float(nir_clip) - Float(red_clip)) / (Float(nir_clip) + Float(red_clip))
ndvi.save("NDVI_KPH.tif")

arcpy.CheckInExtension("Spatial")
print("Selesai: NDVI_KPH.tif tersimpan.")

```

BEDAH KODE (Bagi Praktisi Python): Skrip ArcPy untuk ArcMap berjalan di Python 2.7, sedangkan ArcGIS Pro memakai Python 3. Karena itu, skrip lama sering perlu diubah sebelum dipakai di Pro. Seperti pada Bagian B, pembacaan COG langsung dari cloud tidak tersedia di skrip ini. **[CEK]**

Ringkasan, Latihan, dan Bantuan Bab 1

Ringkasan

- Citra satelit dapat dibaca langsung dari cloud lewat STAC dan COG, tanpa mengunduh berkas utuh.
- Ubah CRS proyek ke sistem metrik (UTM) di awal agar luas dan jarak akurat.
- XYZ Tiles memuat peta dasar tanpa plugin tambahan.
- PyQGIS dapat memuat COG, memotong, dan menghitung NDVI dalam satu skrip.
- **[ArcGIS Pro]** STAC bawaan; Project dan Project Raster untuk mengubah proyeksi; ArcPy dengan Python 3.
- **[ArcMap]** Tidak ada padanan STAC (alternatif: image service); Project dan Project Raster; ArcPy dengan Python 2.7.

Latihan

1. Cari citra Sentinel-2 dengan tutupan awan di bawah 10% untuk area Anda, lalu muat Band 4 dan Band 8 sebagai COG.
2. Tentukan zona UTM area Anda dengan rumus di langkah 1.2, lalu atur CRS proyek dan catat kode EPSG-nya.
3. Jalankan skrip 1.1 dengan batas area Anda sendiri. Bagian mana pada hasil NDVI yang tampak paling hijau pekat, dan apakah sesuai dengan kenyataan di lapangan?
4. **[ArcGIS Pro / ArcMap]** Proyeksikan satu citra dengan **Project Raster**. Catat metode resampling yang Anda pilih dan alasannya.
5. **[ArcGIS Pro / ArcMap]** Jalankan Skrip 1.1 versi Anda dengan berkas B04, B08, dan batas area sendiri, lalu bandingkan hasil NDVI-nya dengan hasil di QGIS.

Jika terjadi masalah

- Layer gagal dimuat: periksa koneksi internet dan pastikan alamat citra diawali /vsicurl/. **[CEK]**
- Luas area tampak tidak masuk akal: periksa apakah CRS proyek masih berupa derajat (EPSG:4326).
- **[ArcGIS Pro / ArcMap]** Skrip berhenti dengan galat lisensi: pastikan ekstensi **Spatial Analyst** (atau **Image Analyst** di ArcGIS Pro) tersedia untuk akun Anda. **[CEK]**
- **[ArcMap]** Skrip dari ArcGIS Pro gagal dijalankan: periksa apakah ia memakai sintaks Python 3, karena ArcMap memakai Python 2.7.
- **[ArcMap]** Dialog Add Basemap gagal menambahkan peta dasar: lihat catatan *Masalah yang Diketahui* di bagian 1.3 ArcMap.

Tabel Perbandingan Bab 1: QGIS, ArcGIS Pro, dan ArcMap 10.8

Kolom *Status* menjelaskan seberapa pasti padanan itu. **Terbukti**: dokumentasi resmi sudah dibaca penulis. **Kemungkinan**: berdasarkan informasi umum. **[CEK]**: perlu dicoba di perangkat lunaknya.

Langkah	QGIS	ArcGIS Pro	ArcMap 10.8	Status
1.1 Citra cloud	STAC API Browser (plugin); COG	Dukungan STAC bawaan; panel Explore STAC	Tidak ada padanan STAC; image service atau Add Data From ArcGIS Online	Pro: ada, menu [CEK]. ArcMap: tidak ada padanan langsung
1.2 CRS proyek	Project Properties - CRS	Map Properties - Coordinate Systems	Data frame properties	Letak menu [CEK]
1.2 Ubah proyeksi data	Reproject layer	Project; Project Raster	Project; Project Raster	Terbukti ada di Pro dan ArcMap; tanpa ekstensi
1.3 Peta dasar	XYZ Tiles (panel Browser)	Galeri Basemap; XYZ murni belum terverifikasi	Add Basemap; Add Data From ArcGIS Online	Menu [CEK]; XYZ murni: tidak ada padanan yang dijamin
1.4 Bahasa otomatis	PyQGIS 3 (Python 3)	ArcPy (Python 3)	ArcPy (Python 2.7)	Terbukti (versi Python)
1.4 Baca cloud langsung	/vsicurl/ lewat GDAL	Belum diverifikasi	Belum diverifikasi	Belum diverifikasi
1.4 Potong citra	gdal:clprasterbymas klayer	Extract by Mask	Extract by Mask	Kemungkinan; butuh Spatial Analyst [CEK]
1.4 Hitung NDVI	qgis:rastercalculator	Raster Calculator atau ekspresi arcpy.sa	Ekspresi arcpy.sa (Spatial Analyst)	Pro: Raster Calculator butuh Spatial Analyst atau Image Analyst (terbukti). ArcMap: kemungkinan

Ilustrasi 1.12: Ringkasan ketersediaan padanan pada Bab 1

Ringkasan ketersediaan padanan pada Bab 1

	QGIS	ArcGIS Pro	ArcMap 10.8
Citra cloud / STAC	Ada: plugin STAC API Browser	Ada: bawaan, menu [UJI]	Tidak ada padanan STAC (alternatif: image service)
Ubah proyeksi data	Ada: Reproject layer	Ada: Project, Project Raster	Ada: Project, Project Raster
Peta dasar XYZ	Ada: XYZ Tiles	XYZ murni belum terverifikasi	Add Basemap; XYZ murni tidak ditemukan
Bahasa otomatis	PyQGIS 3 (Python 3)	ArcPy (Python 3)	ArcPy (Python 2.7)

BAB 2: Sinkronisasi Lapangan (Ground-Truthing Cloud)

Konteks Riset/Bisnis: Anda telah membuat draf peta deforestasi atau peta kesehatan tanaman dari citra satelit. Namun, peta ini belum sah sebelum divalidasi dengan kondisi nyata di lapangan (*ground-truthing*). Anda tidak perlu lagi mencetak peta kertas dan mencatat di buku tulis. Cukup minta tim *surveyor* memakai *smartphone* dan memotret temuan (misal: tunggul kayu atau hama). Datanya otomatis masuk ke QGIS di kantor Anda secara *real-time*.

BAGIAN A | QGIS

2.1. Membangun Basis Data & Formulir Survei Cerdas di QGIS

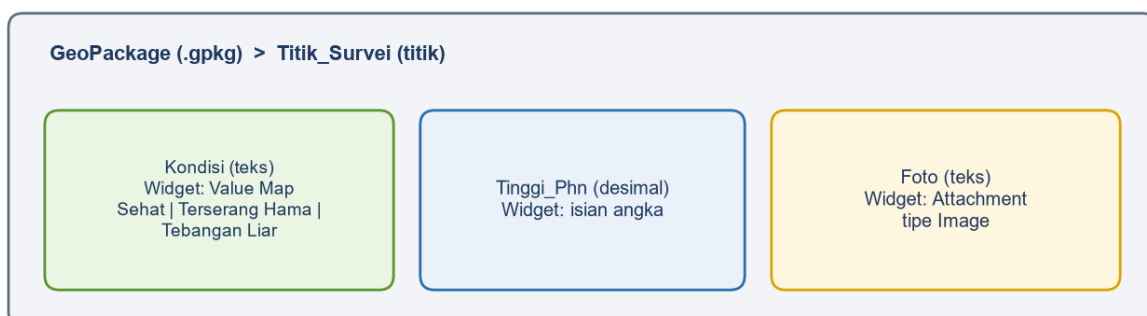
Anda memakai format *GeoPackage* (.gpkg) yang ringan dan merupakan standar modern pengganti *Shapefile*.

A. Langkah Eksekusi (Persiapan Desktop):

1. Buat layer poin baru: **Layer > Create Layer > New GeoPackage Layer**.
2. Beri nama tabel: *Titik_Survei*. Tambahkan *field*: Kondisi (Teks), Tinggi_Phnl (Desimal), dan Foto (Teks).
3. Klik kanan layer *Titik_Survei* > **Properties > Attributes Form**.
4. Ubah **Attribute editor layout** dari *Autogenerate* menjadi **Drag and drop designer**. Dengan ini, Anda bisa mendesain antarmuka aplikasi seluler langsung dari QGIS.
5. Konfigurasi setiap *field*:
 - Klik *field* Kondisi. Pada bagian *Widget Type*, ubah menjadi **Value Map**. Masukkan pilihan: Sehat, Terserang Hama, Tebangan Liar.
 - Klik *field* Foto. Pada *Widget Type*, ubah menjadi **Attachment**. Centang *Integrated document viewer* dan pilih tipe *Image*.

Ilustrasi 2.1: Struktur data survei di QGIS

Struktur data survei di QGIS

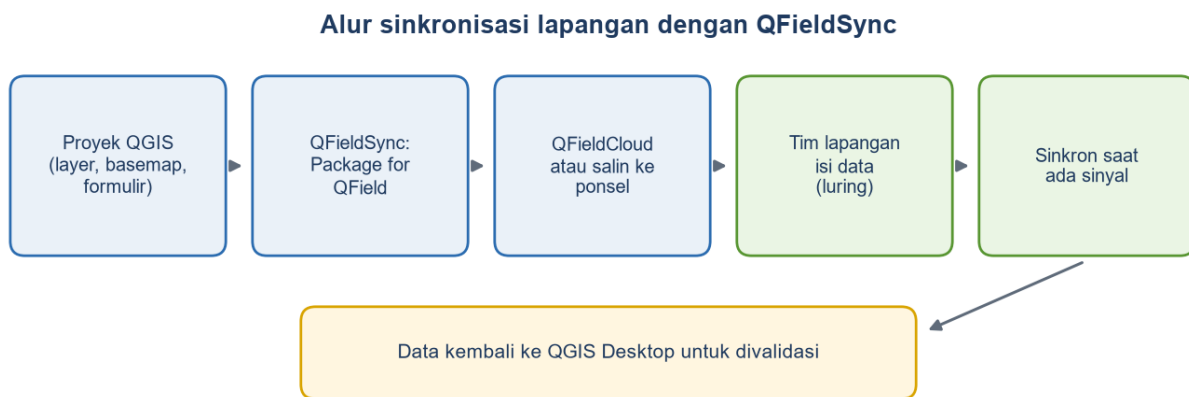


2.2. Sinkronisasi ke Perangkat Seluler (QField/Mergin Maps)

Anda akan mengirim proyek ini ke *Cloud* agar bisa diakses oleh tim lapangan.

1. Buka **Plugins > Manage and Install Plugins**, instal **QFieldSync** (atau Mergin Maps).
2. Simpan proyek QGIS Anda.
3. Klik ikon **QFieldSync > Package for QField**.
4. Pilih folder tujuan. QGIS akan memaketkan seluruh layer, basemap, dan formulir ke dalam satu folder terenkripsi.
5. Unggah proyek ke *QFieldCloud* (atau *copy* langsung ke *smartphone* Android/iOS tim Anda). Tim lapangan kini dapat mengisi data secara luring (*offline*) di tengah hutan. Aplikasi akan otomatis menyinkronkan data saat mendapat sinyal internet.

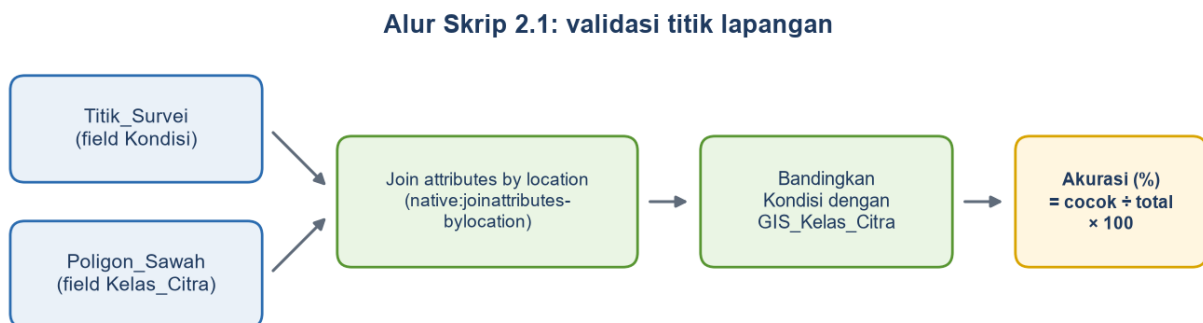
Ilustrasi 2.2: Alur sinkronisasi lapangan dengan QFieldSync



2.3. Eksekusi Tingkat Lanjut: Validasi Otomatis via PyQGIS

Saat tim lapangan mensinkronisasi ratusan titik survei kembali ke QGIS Desktop, Anda tidak perlu mengeceknya satu per satu. Gunakan skrip Python untuk melakukan *Spatial Join* antara titik survei lapangan dan poligon klasifikasi Anda, lalu berikan skor akurasi.

Ilustrasi 2.3: Alur Skrip 2.1: validasi titik lapangan



PYTHON

```
# [SKRIP 2.1: Validasi Otomatis Titik Ground-Truth vs Poligon Klasifikasi]
# Penulis: Badar Mubarak Yogaswara

import processing
from qgis.core import QgsProject

print("🔵 Memulai Validasi Lapangan Otomatis...")

layer_survei = QgsProject.instance().mapLayersByName('Titik_Survei')[0]
layer_klasifikasi = QgsProject.instance().mapLayersByName('Poligon_Sawah')[0]

# Melakukan Spatial Join (Menarik atribut poligon ke titik survei yang berada di dalamnya)
hasil_join = processing.run("native:joinattributesbylocation", {
    'INPUT': layer_survei,
    'JOIN': layer_klasifikasi,
    'PREDICATE': [0], # 0 = Intersects
    'JOIN_FIELDS': ['Kelas_Citra'], # Field dari citra satelit
    'METHOD': 0,
    'DISCARD_NONMATCHING': False,
    'PREFIX': 'GIS_',
    'OUTPUT': 'memory:Validasi_Hasil'
})['OUTPUT']

# Menambahkan layer hasil ke QGIS
QgsProject.instance().addMapLayer(hasil_join)

# Menghitung Akurasi
titik_cocok = 0
total_titik = hasil_join.featureCount()

for fitur in hasil_join.getFeatures():
    if str(fitur['Kondisi']).lower() == str(fitur['GIS_Kelas_Citra']).lower():
        titik_cocok += 1

akurasi = (titik_cocok / total_titik) * 100 if total_titik > 0 else 0

print(f"🇮🇩 Laporan Validasi: {titik_cocok} dari {total_titik} titik cocok.")
print(f"🎯 Akurasi Keseluruhan (Overall Accuracy): {akurasi:.2f}%")
```

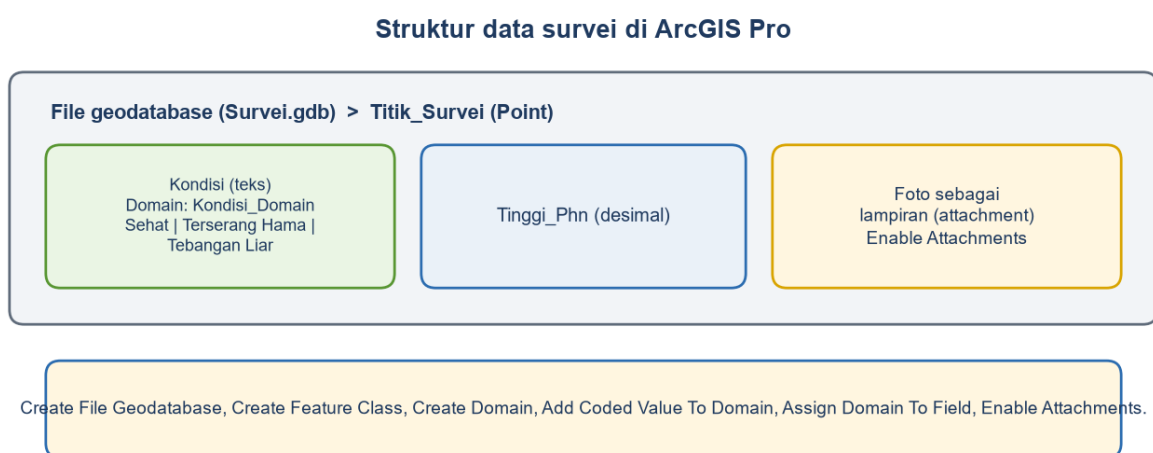
BAGIAN B | ArcGIS Pro

PERANGKAT DAN LISENSI: Alat pada bagian ini (**Create File Geodatabase, Create Feature Class, Create Domain, Add Coded Value To Domain, Assign Domain To Field, Enable Attachments**, dan **Spatial Join**) tersedia pada semua level lisensi tanpa ekstensi. Aplikasi lapangan **Field Maps** memerlukan akun ArcGIS Online atau Enterprise. **[CEK]**

2.1. Membangun Basis Data dan Formulir Survei (ArcGIS Pro)

Padanan GeoPackage di ArcGIS adalah **file geodatabase**. Pilihan isian yang dibatasi (Value Map di QGIS) dibuat dengan **domain**, dan foto disimpan sebagai **lampiran (attachment)**.

Ilustrasi 2.4: Struktur data survei di ArcGIS Pro



A. Langkah Eksekusi (Persiapan Desktop):

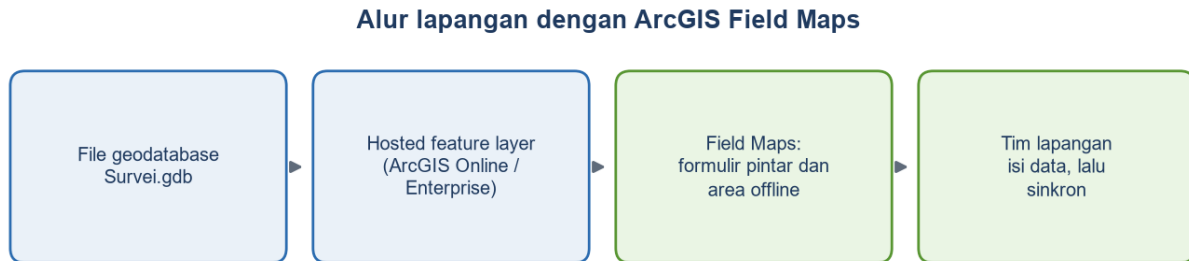
1. Buat file geodatabase dengan alat **Create File Geodatabase**, misalnya bernama Survei.gdb.
2. Buat layer titik dengan alat **Create Feature Class** (tipe geometri *Point*) dan beri nama **Titik_Survei**.
3. Tambahkan field **Kondisi** (teks) dan **Tinggi_Phnl** (desimal) pada Titik_Survei.
4. Buat domain dengan **Create Domain** (jenis *coded value*), misalnya Kondisi_Domain. Isi pilihannya dengan **Add Coded Value To Domain**: Sehat, Terseang Hama, dan Tebangan Liar.
5. Pasang domain ke field Kondisi dengan **Assign Domain To Field**.
6. Aktifkan lampiran foto dengan **Enable Attachments** pada Titik_Survei. Foto disimpan sebagai lampiran, sehingga Anda tidak perlu field Foto bertipe teks seperti pada QGIS.
7. Rancang tampilan formulir seluler di aplikasi web Field Maps setelah layer diterbitkan (lihat 2.2). **[CEK]**

TIPS: Domain memastikan surveyor hanya memilih nilai yang sah. Pakai istilah yang sama persis dengan kelas pada poligon klasifikasi, supaya validasi di 2.3 dapat berjalan.

2.2. Sinkronisasi ke Perangkat Seluler (ArcGIS Pro)

Aplikasi lapangan Esri adalah **ArcGIS Field Maps**, yang menggantikan Collector, Explorer, dan Tracker.

Ilustrasi 2.5: Alur lapangan dengan ArcGIS Field Maps



[UJI] Nama menu dan langkah persisnya belum diverifikasi.

1. Terbitkan Titik_Survei sebagai *hosted feature layer* di ArcGIS Online atau Enterprise. Field Maps hanya dapat mengedit layer yang di-host, bukan layer dari berkas yang tersimpan di dalam peta. **[CEK]**
2. Buat peta web yang berisi layer itu.
3. Atur formulir di Field Maps: urutkan isian dengan seret dan lepas (*smart forms*) dan wajibkan isian tertentu. **[CEK]**
4. Untuk kerja tanpa sinyal, siapkan area peta luring (*offline map area*). **[CEK]**
5. Bagikan peta ke tim. Tim membukanya di aplikasi Field Maps pada ponsel, dan data disinkronkan saat ada sinyal.

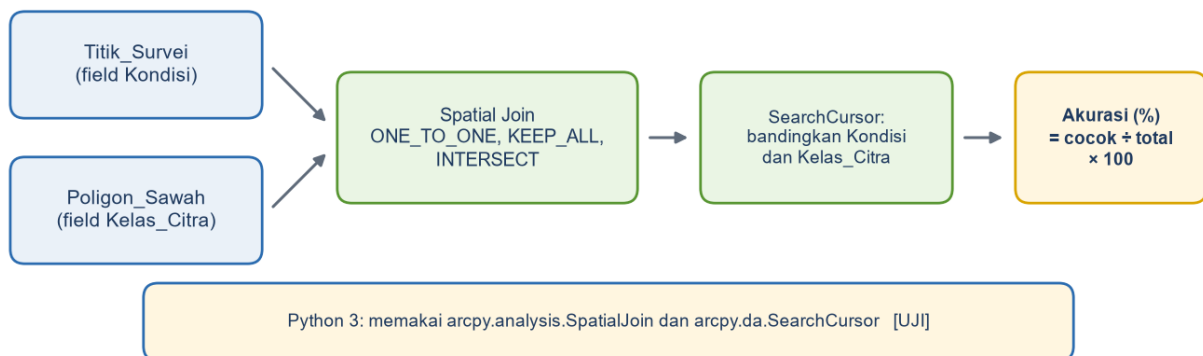
PERHATIAN [CEK]: Letak menu dan langkah persis di ArcGIS Online dan Field Maps belum diverifikasi, sehingga bagian ini belum memakai tangkapan layar asli. Periksa sendiri di akun Anda sebelum dipakai.

2.3. Eksekusi Tingkat Lanjut: Validasi Otomatis dengan ArcPy

Skenarionya sama dengan Bagian A: menggabungkan titik survei dengan poligon klasifikasi, lalu menghitung akurasi. Alat penggabungnya adalah **Spatial Join**.

Ilustrasi 2.6: Alur Skrip 2.1 dengan ArcPy di ArcGIS Pro

Alur Skrip 2.1 dengan ArcPy (ArcGIS Pro, Python 3)



Buka jendela Python di ArcGIS Pro dan jalankan skrip berikut:

PYTHON (ARCPY, PYTHON 3)

```

# [SKRIP 2.1-Pro: Validasi Titik Ground-Truth vs Poligon Klasifikasi dengan ArcPy]
# Penulis: Badar Mubarak Yogaswara
# [CEK] Cocokkan dengan versi Anda. Spatial Join tersedia pada semua level lisensi.
import arcpy

arcpy.env.workspace = r"C:\data_kph\Survei.gdb"      # ganti dengan lokasi Anda
arcpy.env.overwriteOutput = True

survei = "Titik_Survei"
klasifikasi = "Poligon_Sawah"
hasil = "Validasi_Hasil"

print("Memulai validasi lapangan otomatis...")

# 1. Spatial Join: tarik atribut poligon ke titik survei yang berada di dalamnya
arcpy.analysis.SpatialJoin(survei, klasifikasi, hasil,
                           "JOIN_ONE_TO_ONE", "KEEP_ALL", "", "INTERSECT")

# 2. Hitung akurasi: bandingkan Kondisi dengan Kelas_Citra
total_titik = 0
titik_cocok = 0
with arcpy.da.SearchCursor(hasil, ["Kondisi", "Kelas_Citra"]) as kursor:
    for kondisi, kelas in kursor:
        total_titik += 1
        if str(kondisi).lower() == str(kelas).lower():
            titik_cocok += 1

akurasi = (titik_cocok / total_titik) * 100 if total_titik > 0 else 0
print(f"Laporan Validasi: {titik_cocok} dari {total_titik} titik cocok.")
print(f"Akurasi Keseluruhan (Overall Accuracy): {akurasi:.2f}%")

```

BEDAH KODE (Bagi Praktisi Python): **Spatial Join** memakai opsi JOIN_ONE_TO_ONE, KEEP_ALL, dan INTERSECT. Setiap titik tetap ada, dan atribut poligon yang memotongnya ditarik ke titik. arcpy.da.SearchCursor membaca baris hasil. Seperti pada QGIS, skrip membandingkan teks Kondisi dengan Kelas_Citra, jadi kedua field harus memakai istilah yang sama. Bila kedua layer punya field bernama sama, ArcGIS dapat mengubah nama kolom hasil join. **[CEK]**

ALTERNATIF: Attribute Rules memungkinkan aturan validasi langsung di geodatabase (jenis *calculation*, *constraint*, dan *validation*). Dataset yang menggunakannya tidak kompatibel dengan ArcMap.

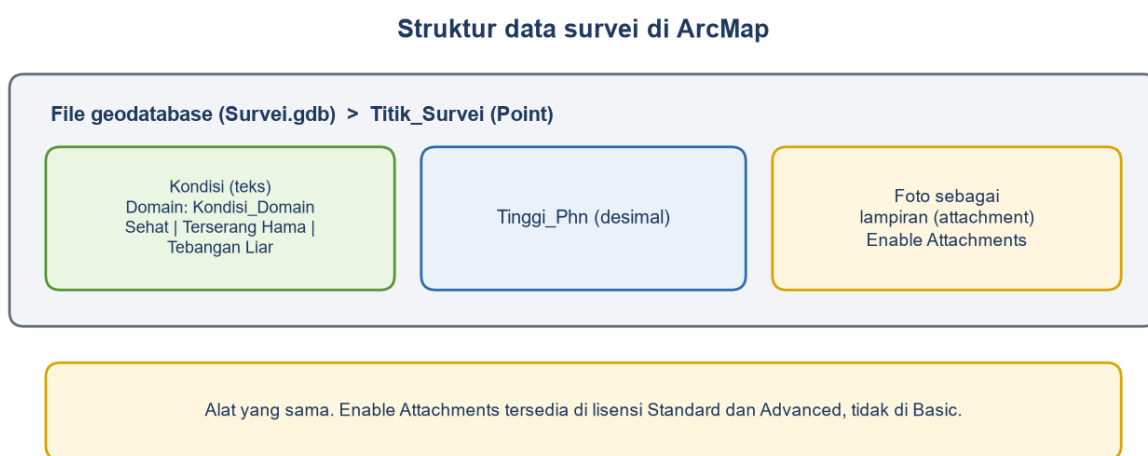
BAGIAN C | ArcMap 10.8

PERANGKAT DAN LISENSI: ArcMap memakai **Python 2.7**. Alat pada bagian ini tersedia pada semua level lisensi, **kecuali Enable Attachments** yang hanya ada pada lisensi **Standard** dan **Advanced**.

2.1. Membangun Basis Data dan Formulir Survei (ArcMap)

Konsepnya sama dengan Bagian B: **file geodatabase**, **domain** untuk pilihan terbatas, dan **lampiran** untuk foto.

Ilustrasi 2.7: Struktur data survei di ArcMap



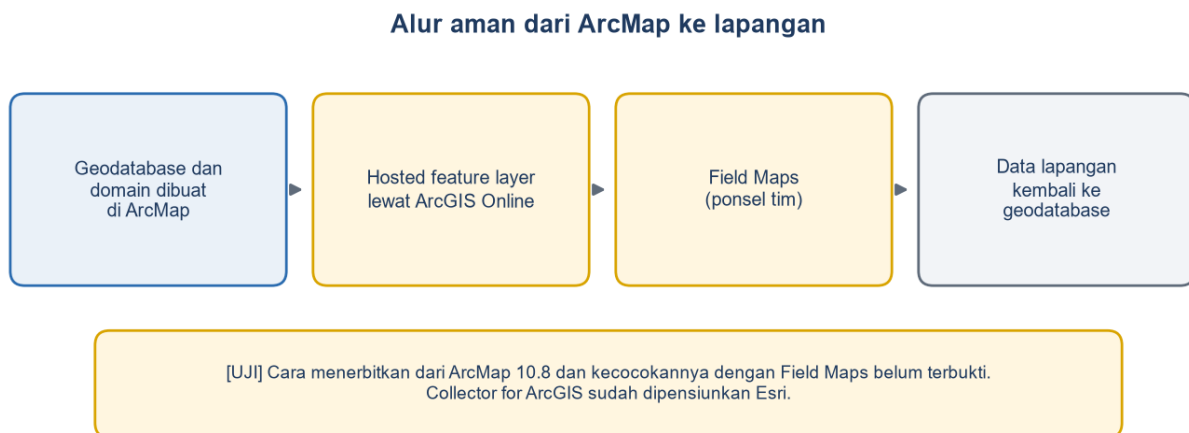
A. Langkah Eksekusi (Persiapan Desktop):

1. Buat file geodatabase dengan alat **Create File GDB**, misalnya bernama Survei.gdb. Folder tujuan harus sudah ada.
2. Buat layer titik dengan alat **Create Feature Class** (tipe geometri *POINT*) dan beri nama **Titik_Survei**.
3. Tambahkan field **Kondisi** (teks) dan **Tinggi_Phnl** (desimal).
4. Buat domain dengan **Create Domain** (jenis *CODED*), lalu isi pilihannya dengan **Add Coded Value To Domain**: Sehat, Terserang Hama, dan Tebangan Liar.
5. Pasang domain ke field Kondisi dengan **Assign Domain To Field**.
6. Aktifkan lampiran foto dengan **Enable Attachments**. Alat ini memerlukan lisensi Standard atau Advanced.

TIDAK ADA PADANAN LANGSUNG: ArcMap tidak punya fitur setara *Drag and drop designer* untuk merancang formulir seluler. Tampilan formulir diatur di aplikasi lapangan (2.2). **[CEK]**

2.2. Sinkronisasi ke Perangkat Seluler (ArcMap)

Ilustrasi 2.8: Alur aman dari ArcMap ke lapangan



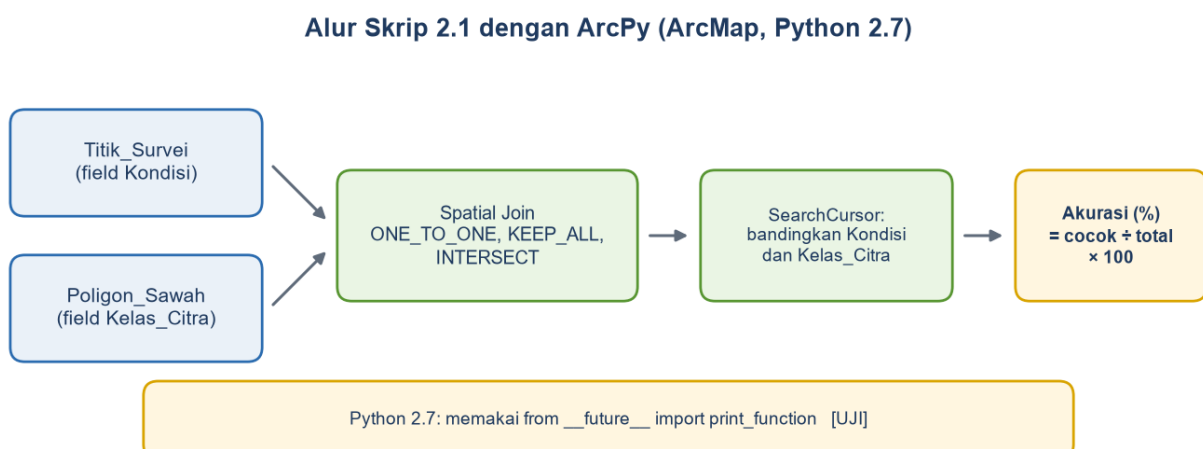
1. Siapkan data di ArcMap seperti pada 2.1.
2. Terbitkan sebagai *hosted feature layer* lewat ArcGIS Online. Cara persis menerbitkan dari ArcMap 10.8 belum diverifikasi. **[CEK]**
3. Pakai Field Maps untuk pengumpulan data. Collector for ArcGIS sudah dipensiunkan Esri.
4. Setelah data terkumpul, salin hasilnya kembali ke geodatabase. **[CEK]**

PERLU DICEK PENGGUNA: Kecocokan ArcMap 10.8 dengan Field Maps belum terbukti. Bagian ini ditulis sebagai alur paling aman, bukan alur yang sudah teruji.

2.3. Eksekusi Tingkat Lanjut: Validasi Otomatis dengan ArcPy (Python 2.7)

Skenario dan hasil sama dengan Bagian B, tetapi kode harus mengikuti Python 2.7.

Ilustrasi 2.9: Alur Skrip 2.1 dengan ArcPy di ArcMap



Buka jendela Python di ArcMap dan jalankan skrip berikut:

PYTHON (ARCPY, PYTHON 2.7)

```
# [SKRIP 2.1-ArcMap: Validasi Titik Ground-Truth vs Poligon Klasifikasi dengan ArcPy]
# Penulis: Badar Mubarak Yogaswara
# [CEK] Cocokkan dengan versi Anda. Spatial Join tersedia pada semua level lisensi.
from __future__ import print_function # agar print() berlaku di Python 2.7
import arcpy

arcpy.env.workspace = r"C:\data_kph\Survei.gdb" # ganti dengan lokasi Anda
arcpy.env.overwriteOutput = True

survei = "Titik_Survei"
klasifikasi = "Poligon_Sawah"
hasil = "Validasi_Hasil"

print("Memulai validasi lapangan otomatis...")

# 1. Spatial Join: tarik atribut poligon ke titik survei yang berada di dalamnya
arcpy.SpatialJoin_analysis(survei, klasifikasi, hasil,
                           "JOIN_ONE_TO_ONE", "KEEP_ALL", "", "INTERSECT")

# 2. Hitung akurasi: bandingkan Kondisi dengan Kelas_Citra
total_titik = 0
titik_cocok = 0
with arcpy.da.SearchCursor(hasil, ["Kondisi", "Kelas_Citra"]) as kursor:
    for kondisi, kelas in kursor:
        total_titik += 1
        if str(kondisi).lower() == str(kelas).lower():
            titik_cocok += 1

akurasi = (float(titik_cocok) / total_titik) * 100 if total_titik > 0 else 0
print("Laporan Validasi: {} dari {} titik cocok.".format(titik_cocok, total_titik))
print("Akurasi Keseluruhan (Overall Accuracy): {:.2f}%".format(akurasi))
```

BEDAH KODE (Bagi Praktisi Python): Python 2.7 membagi dua bilangan bulat dengan pembulatan ke bawah, jadi skrip memakai float() agar akurasi tidak menjadi 0 secara keliru. Seperti pada Bagian B, kedua field yang dibandingkan harus memakai istilah yang sama. **[CEK]**

TIDAK ADA PADANAN: ArcMap tidak punya padanan **Attribute Rules**. Dataset yang memakai Attribute Rules tidak kompatibel dengan ArcMap, sehingga aturan validasi di ArcMap harus dibuat lewat skrip seperti di atas.

Ringkasan, Latihan, dan Bantuan Bab 2

Ringkasan

- GeoPackage menggantikan Shapefile sebagai wadah data lapangan yang ringan.
- Value Map membatasi isian ke pilihan baku, dan Attachment menyimpan foto.
- QFieldSync mengemas proyek ke ponsel dan menyinkronkan data kembali ke QGIS.
- Spatial Join dan skrip dapat menghitung akurasi titik lapangan terhadap poligon klasifikasi.
- **[ArcGIS Pro]** File geodatabase, domain, Enable Attachments, dan Field Maps; validasi dengan Spatial Join dan ArcPy (Python 3).
- **[ArcMap]** Alat geodatabase yang sama; Enable Attachments tidak tersedia di lisensi Basic; alur lapangan perlu diuji; tidak ada Attribute Rules.

Latihan

1. Buat layer Titik_Survei dengan tiga field seperti di langkah 2.1, lalu isi 10 titik contoh di sekitar tempat Anda.
2. Sinkronkan proyek ke ponsel, tambahkan 2 titik baru dengan foto, lalu sinkronkan kembali ke QGIS.
3. Jalankan skrip 2.1 dan catat akurasinya.
4. **[ArcGIS Pro / ArcMap]** Buat Survei.gdb, feature class Titik_Survei, dan domain Kondisi_Domain dengan tiga pilihan seperti di 2.1.
5. **[ArcGIS Pro / ArcMap]** Jalankan Skrip 2.1 versi Anda dan bandingkan akurasinya dengan hasil di QGIS.

Jika terjadi masalah

- Akurasi sangat rendah padahal peta tampak benar: skrip membandingkan teks field Kondisi dengan Kelas_Citra (huruf besar dan kecil dianggap sama). Pastikan pilihan Value Map memakai istilah yang sama persis dengan kelas pada poligon. **[CEK]**
- Skrip tidak menemukan layer: nama layer harus persis Titik_Survei dan Poligon_Sawah.
- **[ArcGIS Pro / ArcMap]** Field Kelas_Citra tidak ditemukan setelah Spatial Join: periksa nama field hasil join, karena namanya dapat berubah bila ada field bernama sama. **[CEK]**
- **[ArcMap]** Enable Attachments tidak dapat dijalankan: alat ini memerlukan lisensi Standard atau Advanced.
- **[ArcMap]** Dataset tidak dapat dibuka: periksa apakah dataset itu memakai Attribute Rules, karena tidak kompatibel dengan ArcMap.

Tabel Perbandingan Bab 2: QGIS, ArcGIS Pro, dan ArcMap 10.8

Kolom *Status* menjelaskan seberapa pasti padanan itu. **Terbukti**: dokumentasi resmi sudah dibaca penulis. **Kemungkinan**: berdasarkan informasi umum. **[CEK]**: perlu dicoba di perangkat lunaknya.

Langkah	QGIS	ArcGIS Pro	ArcMap 10.8	Status
2.1 Wadah data	GeoPackage (.gpkg)	File geodatabase (Create File Geodatabase)	File geodatabase (Create File GDB)	Terbukti: alat ada di Pro dan ArcMap

Langkah	QGIS	ArcGIS Pro	ArcMap 10.8	Status
2.1 Layer titik	New GeoPackage Layer	Create Feature Class (Point)	Create Feature Class (POINT)	Terbukti
2.1 Pilihan terbatas	Value Map	Create Domain, Add Coded Value To Domain, Assign Domain To Field	Alat yang sama	Terbukti; semua level lisensi
2.1 Foto	Widget Attachment	Enable Attachments	Enable Attachments (tidak di lisensi Basic)	Terbukti
2.1 Formulir seluler	Drag and drop designer	Field Maps (form builder)	Tidak ada padanan langsung	Pro: kemungkinan [CEK]. ArcMap: belum diverifikasi
2.2 Kirim ke ponsel	QFieldSync, Package for QField	Hosted feature layer, Field Maps, area offline	Alur aman lewat ArcGIS Online lalu Field Maps	[CEK]; ArcMap perlu diuji pengguna
2.3 Gabung atribut	native:joinattributesbylocation	Spatial Join	Spatial Join	Terbukti ada di Pro dan ArcMap
2.3 Hitung akurasi	Loop fitur PyQGIS	arcpy.da.SearchCursor (Python 3)	arcpy.da.SearchCursor (Python 2.7)	Kemungkinan; arcpy.da ada di keduanya (terbukti dari panduan migrasi)
2.3 Aturan validasi atribut	Skrip PyQGIS	Attribute Rules	Tidak ada padanan	Terbukti: dataset ber-rule tidak kompatibel dengan ArcMap

Ilustrasi 2.10: Ringkasan ketersediaan padanan pada Bab 2

Ringkasan ketersediaan padanan pada Bab 2

	QGIS	ArcGIS Pro	ArcMap 10.8
Wadah data	Ada: GeoPackage	Ada: file geodatabase	Ada: file geodatabase
Pilihan terbatas	Ada: Value Map	Ada: coded value domain	Ada: coded value domain
Foto lapangan	Ada: widget Attachment	Ada: Enable Attachments	Ada, tidak di lisensi Basic
Aplikasi lapangan	Ada: QField / Mergin	Ada: Field Maps [UJI]	Perlu diuji pengguna
Aturan validasi atribut	Skrip PyQGIS	Ada: Attribute Rules	Tidak ada padanan

BAB 3: Pertanian Presisi & Analisis Citra Drone (UAV)

Konteks Riset/Bisnis: Anda menerbangkan drone di atas perkebunan sawit atau hutan produksi. Anda memiliki ratusan ubin ortofoto resolusi 5 cm/piksel dan model elevasi permukaan (DSM). Tugas Anda: menghitung secara presisi jumlah pohon, lokasinya, dan tinggi setiap pohon.

BAGIAN A | QGIS

3.1. Mengekstrak Canopy Height Model (CHM)

Digital Surface Model (DSM) merekam pucuk pohon sekaligus permukaan tanah. *Digital Terrain Model* (DTM) hanya merekam tanah murni. Untuk mengetahui tinggi pohon murni (*Canopy Height Model*), Anda harus mengurangkan keduanya.

Langkah Eksekusi (Cara Visual):

1. Buka **Processing Toolbox > Raster Analysis > Raster Calculator**.
2. Masukkan ekspresi: "DSM_Drone@1" - "DTM_Drone@1".
3. Simpan sebagai CHM_Kebun.tif.
4. Atur simbologi CHM_Kebun ke **Singleband pseudocolor** dengan gradasi warna Hijau ke Merah. Pohon-pohon tinggi akan tampak merah menyala, sedangkan tanah kosong bernilai 0 (hijau).

Ilustrasi 3.1: CHM adalah DSM dikurangi DTM



3.2. Menghitung Jumlah Tegakan Pohon (Local Maxima)

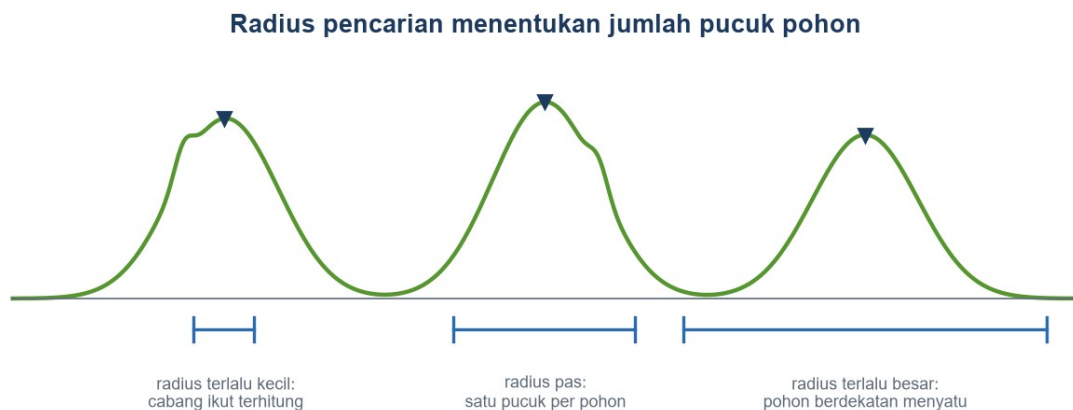
Anda mencari titik tertinggi pada setiap gundukan kanopi di citra CHM. Setiap titik tertinggi dianggap satu pucuk pohon.

Langkah Eksekusi:

1. Buka **Processing Toolbox > Raster Analysis > Find Local Peaks** (Algoritma *native* QGIS 4.x).

2. **Input Raster:** CHM_Kebun.tif.
3. **Distance/Radius:** Atur ke 4 (Meter). *Ini sangat krusial. Jika jarak rata-rata antar pohon (misalnya sawit) adalah 8 meter, masukkan setengahnya agar algoritma tidak menghitung cabang pohon sebagai pohon baru.*
4. **Output:** Layer titik vektor Pucuk_Pohon.shp. Tabel atributnya otomatis berisi nilai tinggi dari citra CHM.

Ilustrasi 3.2: Radius pencarian menentukan jumlah pucuk pohon



JIKA ALAT PENCARI PUNCAK DI QGIS: Alat bernama *Find Local Peaks* tidak ditemukan di dokumentasi inti QGIS dan tidak ada di daftar algoritma QGIS 4.0.2 yang diuji. Pendekatan yang umum dipakai: haluskan CHM dengan *Gaussian Filter*, lalu pakai alat SAGA **Local Minima and Maxima** yang menghasilkan titik maksimum lokal. Alat SAGA memerlukan SAGA yang terhubung ke Processing, dan ketersediaannya di QGIS Anda perlu Anda cek. **[CEK]**

3.3. Eksekusi Tingkat Lanjut: Pemrosesan Latar Belakang (QgsTask) untuk Ratusan Citra Drone

Citra drone sangat berat. Memproses 50 ubin (*tiles*) ortofoto secara berurutan akan membuat antarmuka QGIS *hang* dan *Not Responding*. Anda dapat mengatasinya dengan membuat tugas paralel di latar belakang memakai QgsTask pada Python 3.

Ilustrasi 3.3: Pemrosesan berurutan dan di latar belakang

Memproses 50 ubin: berurutan vs latar belakang

Berurutan (antarmuka hang)



satu proses menunggu proses sebelumnya selesai; QGIS Not Responding

Latar belakang (QgsTask)



PYTHON

```
# [SKRIP 3.1: Batch Processing Citra Drone di Latar Belakang (Non-Blocking)]
# Penulis: Badar Mubarak Yogaswara

import os
import processing
from qgis.core import QgsTask, QgsApplication, QgsMessageLog, Qgis

folder_input = "D:/Data_Drone/Ortofoto_Mentah/"
folder_output = "D:/Data_Drone/NDVI_Hasil/"

def proses_batch_ndvi(task, folder_in, folder_out):
    QgsMessageLog.logMessage("Memulai Batch Processing di Latar Belakang...", "Analisis Drone", Qgis.Info)

    file_list = [f for f in os.listdir(folder_in) if f.endswith('.tif')]
    total_files = len(file_list)

    for i, file_name in enumerate(file_list):
        if task.isCanceled():
            return False

        path_in = os.path.join(folder_in, file_name)
        path_out = os.path.join(folder_out, f"NDVI_{file_name}")

        processing.run("qgis:rastercalculator", {
            'EXPRESSION': '("{}@1" - "{}@2") / ("{}@1" + "{}@2")'.format(path_in, path_in, path_in, path_in),
            'LAYERS': [path_in],
            'OUTPUT': path_out
        })

        task.setProgress((i + 1) / total_files * 100)

    return True

def task_selesai(exception, result=None):
```

```

if result:
    QgsMessageLog.logMessage("✅ Batch Processing Selesai 100%!", "Analisis Drone", Qgs.Success)
else:
    QgsMessageLog.logMessage("❌ Proses dibatalkan atau error.", "Analisis Drone", Qgs.Warning)

task_ndvi = QgsTask.fromFunction("Kalkulasi NDVI 50 Ubin", proses_batch_ndvi,
                                on_finished=task_selesai,
                                folder_in=folder_input, folder_out=folder_output)

QgsApplication.taskManager().addTask(task_ndvi)

```

HASIL UJI SKRIP 3.1: Skrip 3.1 diuji di QGIS 4.0.2 dengan tiga ubin sintetis: tiga berkas NDVI terbentuk dan nilainya sama dengan hitungan langsung. ID qgis:rastercalculator tidak tercantum di dokumentasi QGIS, yang hanya menyebut native:rastercalc. Bila skrip di atas tidak jalan di versi QGIS Anda, pakai alternatif berikut (juga sudah diuji).

Skrip alternatif (memakai native:rastercalc):

PYTHON (ALTERNATIF)

```

# [SKRIP 3.1 - ALTERNATIF: Batch Processing Citra Drone di Latar Belakang (Non-Blocking)]
# Penulis: Badar Mubarak Yogaswara
# Perbedaan dengan Skrip 3.1: memakai native:rastercalc (terdokumentasi) dengan layer bernama,
# bukan qgis:rastercalculator dengan nama layer berupa jalur berkas.

import os
import processing
from qgis.core import QgsTask, QgsApplication, QgsMessageLog, Qgs, QgsRasterLayer

folder_input = "D:/Data_Drone/Ortofoto_Mentah/"
folder_output = "D:/Data_Drone/NDVI_Hasil/"

def proses_batch_ndvi(task, folder_in, folder_out):
    QgsMessageLog.logMessage("Memulai Batch Processing di Latar Belakang...", "Analisis Drone", Qgs.Info)

    file_list = [f for f in os.listdir(folder_in) if f.endswith('.tif')]
    total_files = len(file_list)

    for i, file_name in enumerate(file_list):
        if task.isCanceled():
            return False

        path_in = os.path.join(folder_in, file_name)
        path_out = os.path.join(folder_out, f"NDVI_{file_name}")

        # Beri nama layer agar bisa dirujuk di ekspresi
        layer_in = QgsRasterLayer(path_in, "ortofoto", "gdal")

        processing.run("native:rastercalc", {
            'EXPRESSION': ' ("ortofoto@1" - "ortofoto@2") / ("ortofoto@1" + "ortofoto@2")',
            'LAYERS': [layer_in],
            'OUTPUT': path_out
        })

        task.setProgress((i + 1) / total_files * 100)

    return True

def task_selesai(exception, result=None):
    if result:
        QgsMessageLog.logMessage("✅ Batch Processing Selesai 100%!", "Analisis Drone", Qgs.Success)
    else:

```

```
QgsMessageLog.logMessage("X Proses dibatalkan atau error.", "Analisis Drone", Qgs.Warning)

task_ndvi = QgsTask.fromFunction("Kalkulasi NDVI 50 Ubin", proses_batch_ndvi,
                                on_finished=task_selesai,
                                folder_in=folder_input, folder_out=folder_output)

QgsApplication.taskManager().addTask(task_ndvi)
```

BAGIAN B | ArcGIS Pro

PERANGKAT DAN LISENSI: **Raster Calculator**, **Focal Statistics**, dan **Con** memerlukan ekstensi **Spatial Analyst** atau **Image Analyst**. **Raster to Point** tersedia pada semua level lisensi tanpa ekstensi.

3.1. Mengekstrak Canopy Height Model (ArcGIS Pro)

Konsepnya sama dengan Bagian A: tinggi pohon diperoleh dengan mengurangkan DTM dari DSM.

Langkah Eksekusi (Cara Visual):

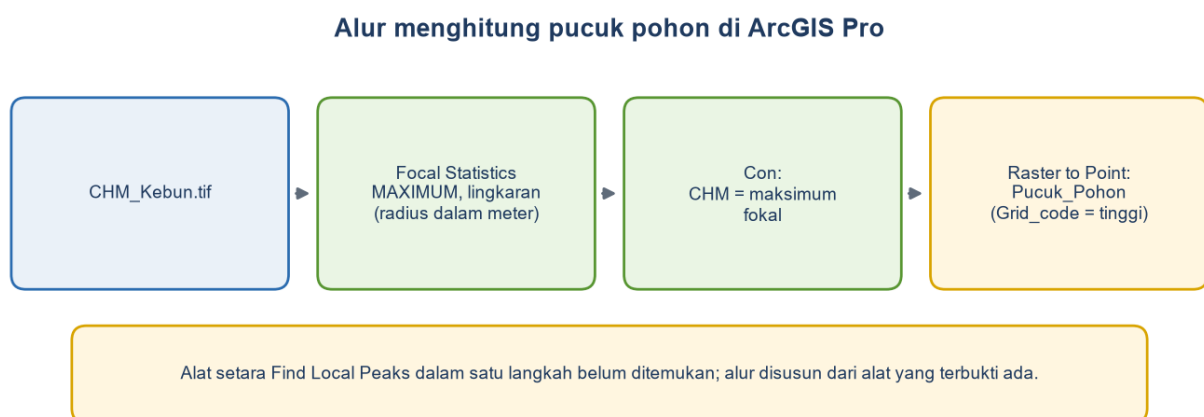
1. Buka alat **Raster Calculator** (kelompok *Spatial Analyst Tools*). **[CEK]**
2. Masukkan ekspresi: "DSM_Drone" - "DTM_Drone".
3. Simpan hasilnya sebagai CHM_Kebun.tif.
4. Atur simbologi CHM_Kebun dengan gradasi warna hijau ke merah, agar pohon tinggi tampak merah dan tanah kosong hijau. **[CEK]**

TIPS: Pastikan DSM dan DTM memakai sistem koordinat dan ukuran piksel yang sama sebelum dikurangkan.

3.2. Menghitung Jumlah Tegakan Pohon (ArcGIS Pro)

Tujuannya sama: setiap titik tertinggi pada gundukan kanopi dianggap satu pucuk pohon. Di ArcGIS Pro hal ini disusun dari tiga alat.

Ilustrasi 3.4: Alur menghitung pucuk pohon di ArcGIS Pro



Langkah Eksekusi:

1. Jalankan **Focal Statistics** pada CHM_Kebun.tif. Pilih statistik **MAXIMUM** dan lingkungan berbentuk lingkaran (*Circle*) dengan radius 4 dalam satuan peta (meter). Simpan hasilnya, misalnya sebagai Fokal_Maks.tif.
2. Pilih sel yang nilainya sama dengan maksimum di sekitarnya, misalnya lewat Raster Calculator: `Con("CHM_Kebun.tif" == "Fokal_Maks.tif", "CHM_Kebun.tif")`. Sel lain menjadi NoData. **[CEK]**

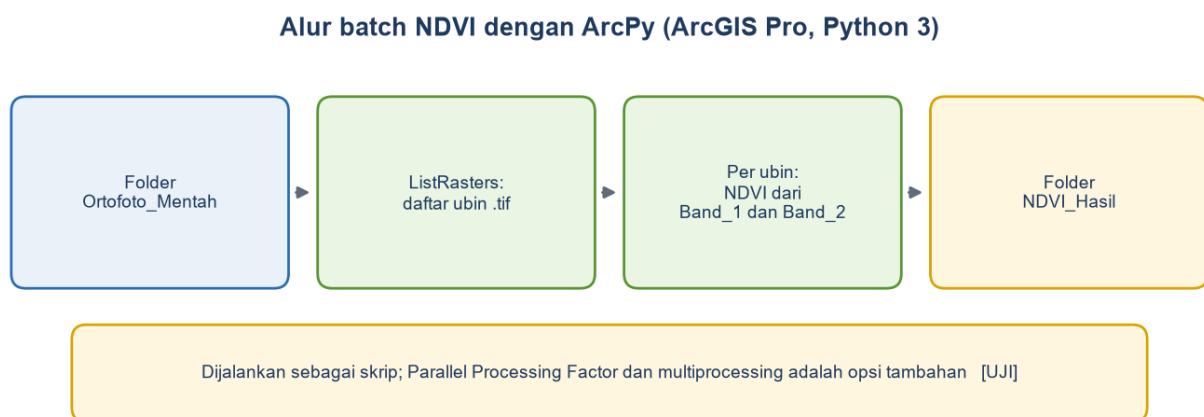
- Ubah hasilnya menjadi titik dengan **Raster to Point**. Simpan sebagai Pucuk_Pohon. Tinggi pohon tersimpan di kolom **Grid_code**.

KRUSIAL: Radius menentukan hasil. Bila jarak rata-rata antar pohon (misalnya sawit) 8 meter, pakai setengahnya agar cabang tidak terhitung sebagai pohon. Alat setara *Find Local Peaks* dalam satu langkah belum ditemukan di dokumentasi, sehingga alurnya disusun dari alat yang terbukti ada.

3.3. Eksekusi Tingkat Lanjut: Pemrosesan Batch dengan ArcPy

ArcGIS Pro tidak punya padanan langsung *QgsTask*. Pemrosesan banyak ubin dilakukan sebagai skrip batch. Esri menyediakan opsi paralel berupa **Parallel Processing Factor** dan modul Python **multiprocessing**; keduanya perlu Anda coba sendiri. [CEK]

Ilustrasi 3.5: Alur batch NDVI dengan ArcPy di ArcGIS Pro



Buka jendela Python di ArcGIS Pro dan jalankan skrip berikut:

PYTHON (ARCPY, PYTHON 3)

```

# [SKRIP 3.1-Pro: Batch Processing Citra Drone dengan ArcPy]
# Penulis: Badar Mubarak Yogaswara
# [CEK] Cocokkan dengan versi Anda. Butuh ekstensi Spatial Analyst atau Image Analyst.
import os
import arcpy
from arcpy.sa import Raster, Float

arcpy.CheckOutExtension("Spatial")
arcpy.env.overwriteOutput = True

folder_input = r"D:\Data_Drone\Ortofoto_Mentah"
folder_output = r"D:\Data_Drone\NDVI_Hasil"
arcpy.env.workspace = folder_input

daftar = arcpy.ListRasters("*.tif")
total = len(daftar)
print(f"Memulai batch {total} ubin...")

for i, nama in enumerate(daftar, start=1):
    # Urutan band mengikuti Skrip 3.1 QGIS: pastikan Band_1 = NIR dan Band_2 = Red
    band1 = Float(Raster(os.path.join(folder_input, nama, "Band_1")))
    band2 = Float(Raster(os.path.join(folder_input, nama, "Band_2")))
  
```

```
ndvi = (band1 - band2) / (band1 + band2)
ndvi.save(os.path.join(folder_output, "NDVI_" + nama))
print(f"{i}/{total} selesai: {nama}")

arcpy.CheckInExtension("Spatial")
print("Batch selesai.")
```

BEDAH KODE (Bagi Praktisi Python): ListRasters mendaftarkan semua ubin .tif. Raster multi-band dibaca per band dengan nama seperti Band_1 dan Band_2. **[CEK]** Rumus mengikuti Skrip 3.1 di Bagian A: $(\text{band 1} - \text{band 2}) / (\text{band 1} + \text{band 2})$. NDVI baku adalah $(\text{NIR} - \text{Red}) / (\text{NIR} + \text{Red})$, jadi pastikan urutan band pada ortofoto Anda.

BAGIAN C | ArcMap 10.8

PERANGKAT DAN LISENSI: ArcMap memakai **Python 2.7**. **Raster Calculator**, **Focal Statistics**, dan **Con** memerlukan ekstensi **Spatial Analyst**. **Raster to Point** tersedia pada semua level lisensi. Raster Calculator hanya tersedia di antarmuka ArcMap dan ModelBuilder, tidak di ArcPy.

3.1. Mengekstrak Canopy Height Model (ArcMap)

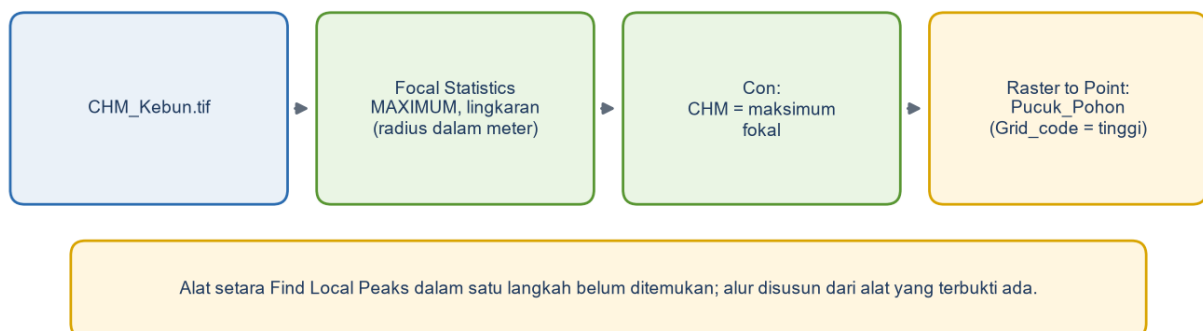
Langkah Eksekusi (Cara Visual):

1. Buka alat **Raster Calculator** (kelompok *Spatial Analyst Tools*). **[CEK]**
2. Masukkan ekspresi Map Algebra: "DSM_Drone" - "DTM_Drone".
3. Simpan hasilnya sebagai CHM_Kebun.tif.
4. Atur simbologi dengan gradasi hijau ke merah. **[CEK]**

3.2. Menghitung Jumlah Tegakan Pohon (ArcMap)

Ilustrasi 3.6: Alur menghitung pucuk pohon di ArcMap

Alur menghitung pucuk pohon di ArcMap



1. Jalankan **Focal Statistics** pada CHM_Kebun.tif dengan statistik **MAXIMUM** dan lingkungan lingkaran (*Circle*) berradius 4 dalam satuan peta (meter). Focal Statistics mendukung MAXIMUM untuk raster desimal.
2. Pilih sel yang sama dengan maksimum di sekitarnya dengan **Con** (dari Raster Calculator):
Con("CHM_Kebun.tif" == "Fokal_Maks.tif", "CHM_Kebun.tif"). **[CEK]**
3. Ubah hasilnya menjadi titik dengan **Raster to Point**; tinggi pohon ada di kolom **Grid_code**.

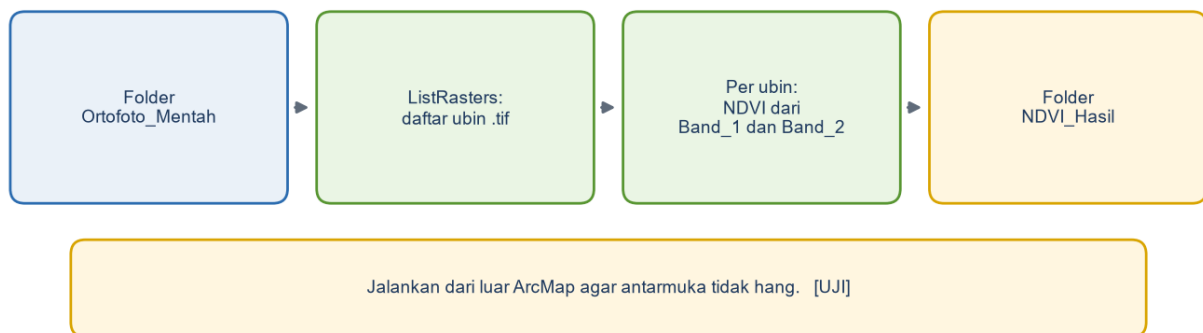
KRUSIAL: Aturan radius sama seperti pada Bagian B: setengah jarak rata-rata antar pohon.

3.3. Eksekusi Tingkat Lanjut: Pemrosesan Batch dengan ArcPy (Python 2.7)

Tidak ada padanan langsung *QgsTask*. Jalankan skrip dari luar ArcMap (misalnya lewat IDLE atau command line) agar antarmuka ArcMap tidak hang. **[CEK]** Parallel Processing Factor ada di ArcMap, tetapi tidak didukung di Background Geoprocessing 64-bit dan hanya dipakai sebagian alat.

Ilustrasi 3.7: Alur batch NDVI dengan ArcPy di ArcMap

Alur batch NDVI dengan ArcPy (ArcMap, Python 2.7)



Jalankan skrip berikut dari luar ArcMap:

PYTHON (ARCPY, PYTHON 2.7)

```

# [SKRIP 3.1-ArcMap: Batch Processing Citra Drone dengan ArcPy]
# Penulis: Badar Mubarak Yogaswara
# [CEK] Cocokkan dengan versi Anda. Butuh ekstensi Spatial Analyst.
from __future__ import print_function # agar print() berlaku di Python 2.7
import os
import arcpy
from arcpy.sa import Raster, Float

arcpy.CheckOutExtension("Spatial")
arcpy.env.overwriteOutput = True

folder_input = r"D:\Data_Drone\Ortofoto_Mentah"
folder_output = r"D:\Data_Drone\NDVI_Hasil"
arcpy.env.workspace = folder_input

daftar = arcpy.ListRasters("*", "TIF")
total = len(daftar)
print("Memulai batch {} ubin...".format(total))

for i, nama in enumerate(daftar, 1):
    # Urutan band mengikuti Skrip 3.1 QGIS: pastikan Band_1 = NIR dan Band_2 = Red
    band1 = Float(Raster(os.path.join(folder_input, nama, "Band_1")))
    band2 = Float(Raster(os.path.join(folder_input, nama, "Band_2")))
    ndvi = (band1 - band2) / (band1 + band2)
    ndvi.save(os.path.join(folder_output, "NDVI_" + nama))
    print("{} / {} selesai: {}".format(i, total, nama))

arcpy.CheckInExtension("Spatial")
print("Batch selesai.")
  
```

BEDAH KODE (Bagi Praktisi Python): Raster Calculator tidak tersedia di ArcPy, jadi perhitungan memakai operator Map Algebra pada objek raster dari **arcpy.sa**. Float() memastikan perhitungan memakai bilangan desimal. **[CEK]**

Ringkasan, Latihan, dan Bantuan Bab 3

Ringkasan

- Canopy Height Model (CHM) adalah DSM dikurangi DTM.
- Pohon dihitung dari titik tertinggi pada tiap gundukan kanopi, dengan radius pencarian sebagai pengaturan kuncinya.
- QgsTask menjalankan pemrosesan banyak ubin di latar belakang agar QGIS tetap bisa dipakai.
- **[ArcGIS Pro / ArcMap]** CHM dengan Raster Calculator; pucuk pohon dengan Focal Statistics (MAXIMUM), Con, dan Raster to Point; batch lewat skrip ArcPy.

Latihan

1. Hitung CHM untuk satu petak contoh, lalu bandingkan tinggi satu pohon yang Anda kenal dengan nilai pada CHM.
2. Hitung jumlah pohon pada petak 1 hektar dengan radius 3, 4, dan 5 meter. Bandingkan dengan hitungan manual pada citra: radius mana yang paling mendekati?
3. Jalankan skrip 3.1 pada 3 ubin contoh sebelum memakainya untuk seluruh 50 ubin. **[CEK]**
4. **[ArcGIS Pro / ArcMap]** Ulangi hitungan pohon dengan Focal Statistics pada radius yang sama dan bandingkan jumlahnya dengan hasil QGIS.

Jika terjadi masalah

- Jumlah pohon terlalu banyak: radius terlalu kecil sehingga cabang ikut terhitung sebagai pohon. Perbesar radius.
- Jumlah pohon terlalu sedikit: radius terlalu besar sehingga pohon yang berdekatan tergabung.
- QGIS tidak merespons saat memproses banyak ubin: jalankan lewat QgsTask seperti pada skrip 3.1.
- **[ArcGIS Pro / ArcMap]** Skrip batch gagal membaca band: periksa nama band raster multi-band (misalnya Band_1 dan Band_2) dan urutannya. **[CEK]**
- **[ArcMap]** ArcMap tidak merespons saat batch: jalankan skrip dari luar ArcMap. **[CEK]**

Tabel Perbandingan Bab 3: QGIS, ArcGIS Pro, dan ArcMap 10.8

Kolom *Status* menjelaskan seberapa pasti padanan itu. **Terbukti**: dokumentasi resmi sudah dibaca penulis. **Kemungkinan**: berdasarkan informasi umum. **[CEK]**: perlu dicoba di perangkat lunaknya.

Langkah	QGIS	ArcGIS Pro	ArcMap 10.8	Status
3.1 Hitung CHM	Raster Calculator	Raster Calculator	Raster Calculator (tidak ada di ArcPy)	Terbukti: Pro dan ArcMap butuh Spatial Analyst (Pro: atau Image Analyst)
3.2 Pencari puncak	Find Local Peaks	Alur: Focal Statistics (MAXIMUM) + Con + Raster to Point	Alur yang sama	Alat tunggal setara belum ditemukan; Focal Statistics dan Raster to Point terbukti

Langkah	QGIS	ArcGIS Pro	ArcMap 10.8	Status
3.2 Pengaturan jarak	Distance/Radius (meter)	Lingkungan Circle, radius dalam satuan peta	Lingkungan Circle, radius dalam satuan peta	Kemungkinan [CEK]
3.3 Latar belakang	QgsTask	Skrip batch ArcPy; opsi paralel	Skrip batch ArcPy 2.7 dari luar ArcMap	Tidak ada padanan langsung QgsTask; opsi paralel [CEK]
3.3 Daftar berkas	os.listdir	arcpy.ListRasters	arcpy.ListRasters	Kemungkinan [CEK]

Ilustrasi 3.8: Ringkasan ketersediaan padanan pada Bab 3

Ringkasan ketersediaan padanan pada Bab 3

	QGIS	ArcGIS Pro	ArcMap 10.8
Hitung CHM	Raster Calculator	Raster Calculator	Raster Calculator
Puncak pohon	Find Local Peaks	Alur: Focal Statistics + Con + Raster to Point	Alur: Focal Statistics + Con + Raster to Point
Pemrosesan batch	QgsTask (latar belakang)	Skrip ArcPy; opsi paralel [UJI]	Skrip ArcPy 2.7; tanpa padanan QgsTask
Ekstensi	Tanpa ekstensi	Spatial/Image Analyst	Spatial Analyst

BAB 4: Kehutanan Lanjut – Analisis Point Cloud (LiDAR) & DTM Ekstraksi

Konteks Riset/Bisnis: Anda menerima data *Point Cloud* (format .las atau .laz) hasil pemindaian LiDAR atau pemetaan fotogrametri dari *drone* di atas hutan yang sangat lebat. Anda diminta membuat Peta Topografi Murni (DTM - *Digital Terrain Model*) untuk merencanakan jalur pembuatan jalan rintisan (*logging road*), namun tanah tertutup rapat oleh kanopi pohon.

BAGIAN A | QGIS

4.1. Memuat Point Cloud Secara Native (Tanpa Plugin Pihak Ketiga)

Dahulu, Anda harus menginstal *LAStools* atau perangkat lunak mahal lainnya. Di QGIS 4.x, *Point Cloud* adalah "warga kelas satu" yang didukung penuh secara *native*.

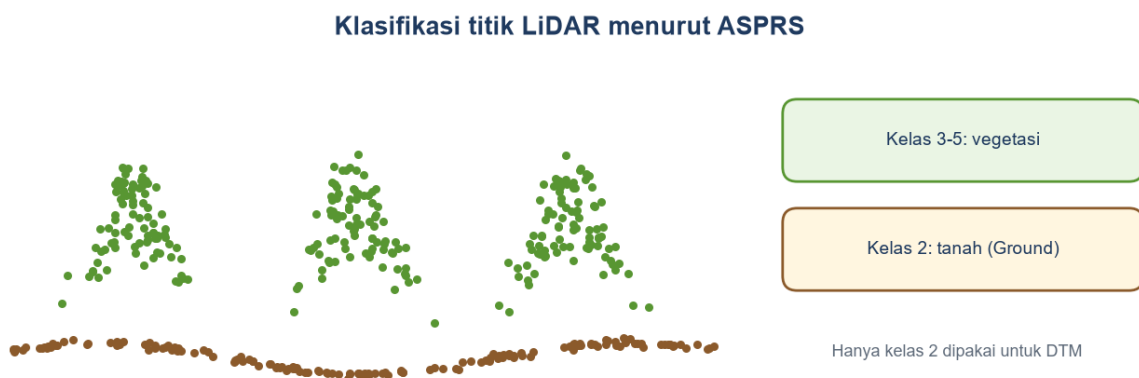
Langkah Eksekusi:

1. Cukup seret dan lepas (*drag and drop*) file .las atau .laz Anda ke panel **Layers** QGIS.
2. QGIS akan langsung merendernya dalam bentuk 2D.
3. Untuk melihatnya dalam 3D, buka menu **View > 3D Map Views > New 3D Map View**. Tahan tombol Shift + *Klik Kiri Mouse* untuk memutar dan melihat struktur hutan dari berbagai sudut.

4.2. Memfilter Kanopi untuk Mendapatkan Bare Earth (Tanah Murni)

Data LiDAR umumnya sudah diklasifikasikan oleh alat pemindainya. Standar global (ASPRS) menetapkan bahwa Kelas 2 adalah Tanah (*Ground*) dan Kelas 3-5 adalah Vegetasi (*Vegetation*).

Ilustrasi 4.1: Klasifikasi titik LiDAR menurut ASPRS



Langkah Eksekusi:

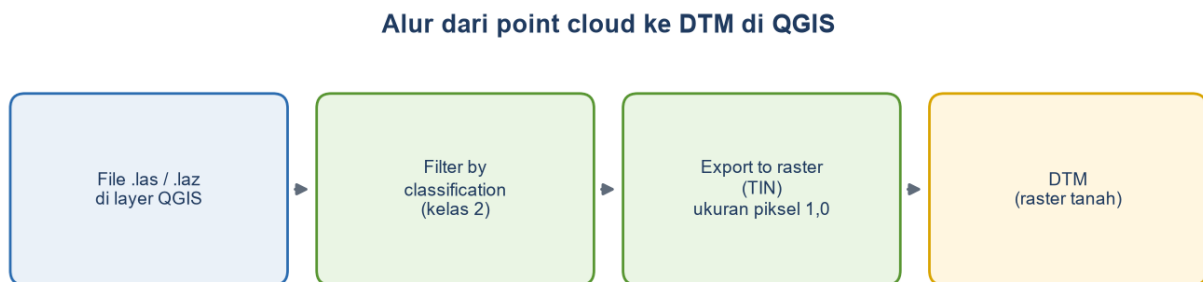
1. Buka **Processing Toolbox > Point Cloud Data Management > Filter by classification**.
2. **Input layer:** Pilih layer .laz Anda.

3. **Classes to keep:** Centang hanya 2 (**Ground**).
4. Klik **Run**. Hasilnya adalah kumpulan titik yang hanya berada di permukaan tanah, menembus kanopi hutan.

4.3. Ekspor Point Cloud Tanah menjadi DTM (Raster)

Titik-titik tanah tadi harus diubah menjadi piksel raster agar bisa dipakai untuk analisis kemiringan lereng (*slope*).

Ilustrasi 4.2: Alur dari point cloud ke DTM di QGIS



Langkah Eksekusi:

1. Buka **Processing Toolbox > Point Cloud Conversion > Export to raster (TIN)**.
2. **Input layer:** Titik Tanah (*Ground*).
3. **Attribute:** Pilih Z (Elevasi/Ketinggian).
4. **Pixel size:** 1.0 (Satu meter per piksel).
5. Klik **Run**. Anda kini memiliki DTM resolusi sangat tinggi yang menembus kanopi hutan, siap digunakan untuk merancang kemiringan jalan rintisan hutan.

NAMA ALAT DI QGIS 4.x (HASIL UJI): Nama alat pada langkah 4.2 dan 4.3 di atas tidak ditemukan di QGIS 4.0.2 yang diuji. Padanannya: **Filter** (kelompok *Point Cloud Extraction*; isi *Filter expression*, misalnya *Classification = 2*) dan **Export raster (using triangulation)** (kelompok *Point Cloud Conversion*; hanya parameter *Resolution*, tanpa pilihan *Attribute*). Untuk memilih atribut Z, pakai **Export to raster**. Ketiganya diuji pada data sintetis: Filter menghasilkan 5.999 titik tanah dan DTM dari triangulasi berselisih rata-rata 0,03 m dari medan sebenarnya. Alat PDAL memerlukan PDAL 2.5.0 atau lebih baru.

KLASIFIKASI TITIK TANAH: Bila data belum terklasifikasi, QGIS 4.0 ke atas memiliki **Classify ground points** (kelompok *Point Cloud Data Management*) yang berbasis algoritma SMRF (Pingel dan kawan-kawan, 2013). Pada data sintetis (tanah terbuka di sela pohon), alat ini mengklasifikasikan 5.999 dari 6.000 titik tanah dengan benar. Hasil itu belum mewakili hutan lebat. SMRF dirancang untuk LiDAR udara dan parameternya (*Cell Size*, *Scalar*, *Slope*, *Threshold*, *Window Size*) perlu disetel sesuai medan. Pada fotogrametri drone di hutan lebat, kamera tidak menembus kanopi sehingga titik tanah yang sebenarnya jarang tertangkap, dan hasil klasifikasi terbatas. Alat ini tidak ada di QGIS 3.x.

BAGIAN B | ArcGIS Pro

PERANGKAT DAN LISENSI: **Classify LAS Ground** dan **Extract LAS** memerlukan ekstensi **3D Analyst**. **LAS Dataset To Raster** memerlukan **Spatial Analyst** atau **3D Analyst**. **Create LAS Dataset** tersedia pada lisensi Standard dan Advanced; pada Basic memerlukan 3D Analyst atau Spatial Analyst.

4.1. Memuat Point Cloud (ArcGIS Pro)

Di ArcGIS Pro, point cloud dikelola lewat **LAS dataset**: satu rujukan ke satu atau banyak berkas .las, .zlas, atau .laz, yang dapat ditampilkan di peta 2D maupun 3D.

Langkah Eksekusi:

1. Buat LAS dataset dengan alat **Create LAS Dataset**, lalu pilih berkas .las, .zlas, atau .laz Anda.
2. Tambahkan LAS dataset ke peta. Titik tampil dengan pewarnaan elevasi, baik di tampilan 2D maupun 3D.
3. Untuk melihatnya dalam 3D, tambahkan LAS dataset ke scene (tampilan 3D), lalu putar untuk melihat struktur hutan dari berbagai sudut. **[CEK]**

PERHATIAN: Berkas .laz tunggal tidak mendapat dukungan penuh seperti LAS dataset. Karena itu, buat LAS dataset lebih dulu, bukan menambahkan berkas .laz langsung ke peta.

4.2. Memfilter Kanopi untuk Mendapatkan Bare Earth (ArcGIS Pro)

Menurut standar ASPRS, kelas 2 adalah tanah (*Ground*) dan kelas 3 sampai 5 adalah vegetasi. Bila data Anda belum terklasifikasi, titik tanah perlu diklasifikasikan lebih dulu.

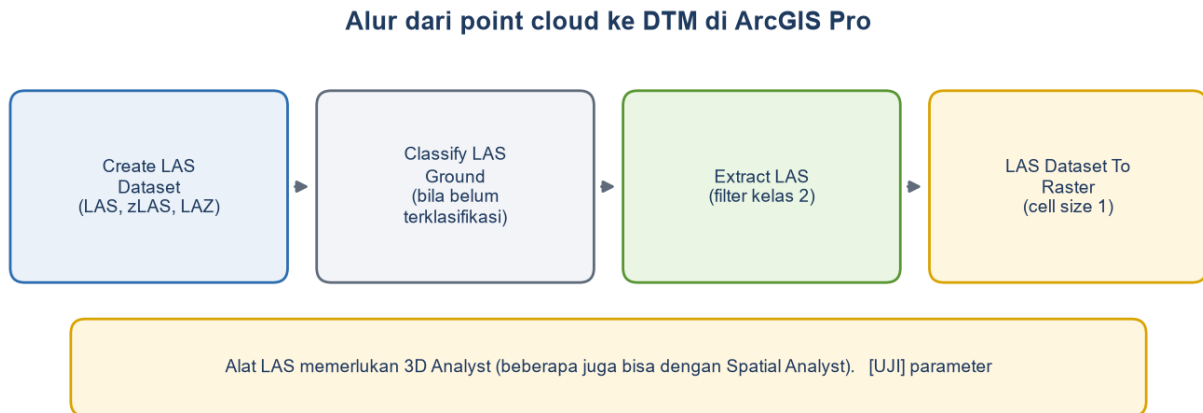
Langkah Eksekusi:

1. Bila data belum terklasifikasi, jalankan **Classify LAS Ground**. Alat ini menyediakan metode *Standard*, *Conservative*, dan *Aggressive*, dan memerlukan LAS dataset dengan sistem koordinat terproyeksi.
2. Jalankan **Extract LAS** pada LAS dataset. Pada filter, pilih kode klasifikasi **2** (Ground).
3. Hasilnya adalah kumpulan titik yang hanya berada di permukaan tanah, menembus kanopi hutan.

PERHATIAN: Seperti SMRF di QGIS, klasifikasi tanah paling cocok untuk LiDAR. Pada fotogrametri drone di hutan lebat, kamera tidak menembus kanopi sehingga titik tanah yang sebenarnya jarang tertangkap, dan hasilnya terbatas. **[CEK]**

4.3. Ekspor Point Cloud Tanah menjadi DTM (ArcGIS Pro)

Ilustrasi 4.3: Alur dari point cloud ke DTM di ArcGIS Pro



Langkah Eksekusi:

1. Buat LAS dataset dari titik tanah hasil Extract LAS, atau pakai LAS dataset yang sudah difilter ke kelas 2.
2. Jalankan **LAS Dataset To Raster** (kelompok *Conversion Tools*).
3. Pilih nilai yang dirasterkan: **elevasi** (alat ini juga dapat memakai intensitas atau RGB). **[CEK]**
4. Atur ukuran sel (*cell size*) menjadi 1,0 (satu meter per piksel), lalu jalankan. Metode interpolasinya dapat berupa *binning* atau triangulasi. **[CEK]**

TIPS: Anda kini memiliki DTM beresolusi tinggi yang menembus kanopi hutan, siap dipakai merancang kemiringan jalan rintisan.

BAGIAN C | ArcMap 10.8

PERANGKAT DAN LISENSI: ArcMap memakai **Python 2.7. Classify LAS Ground, Extract LAS, dan LAS Dataset To Raster** memerlukan ekstensi **3D Analyst. Create LAS Dataset** tersedia pada lisensi Standard dan Advanced; pada Basic memerlukan 3D Analyst atau Spatial Analyst.

4.1. Memuat Point Cloud (ArcMap)

Konsepnya sama dengan Bagian B: point cloud dikelola lewat **LAS dataset**.

Langkah Eksekusi:

1. Buat LAS dataset dengan alat **Create LAS Dataset**. Dokumentasi alat ini menyebut berkas **.las** dan **.zlas**.
2. Tambahkan LAS dataset ke peta. Untuk tampilan 3D, ArcMap memakai aplikasi 3D terpisah dari ekstensi 3D Analyst (misalnya ArcScene). **[CEK]**

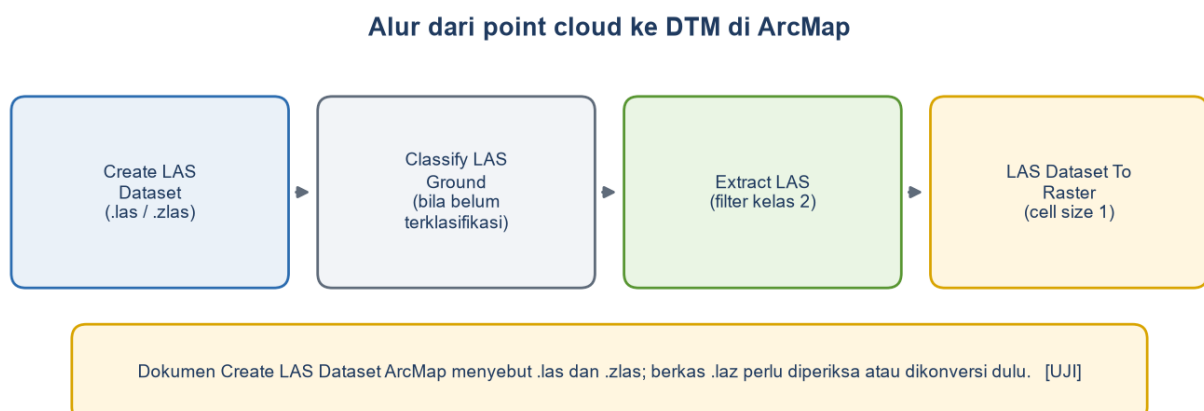
PERHATIAN [CEK]: Dokumentasi Create LAS Dataset di ArcMap tidak menyebut format **.laz**. Bila data Anda berformat **.laz**, periksa dukungannya di perangkat Anda atau ubah dulu ke **.las** atau **.zlas**.

4.2. Memfilter Kanopi untuk Mendapatkan Bare Earth (ArcMap)

1. Bila data belum terklasifikasi, jalankan **Classify LAS Ground** (metode *Standard*, *Conservative*, atau *Aggressive*; LAS dataset harus berkoordinat terproyeksi).
2. Jalankan **Extract LAS** pada LAS dataset dan pilih kode klasifikasi **2** (Ground). Penyaringan dilakukan lewat pengaturan filter layer atau alat Make LAS Dataset Layer.
3. Hasilnya adalah titik yang hanya berada di permukaan tanah.

4.3. Ekspor Point Cloud Tanah menjadi DTM (ArcMap)

Ilustrasi 4.4: Alur dari point cloud ke DTM di ArcMap



1. Jalankan **LAS Dataset To Raster** (kelompok *Conversion Toolbox*), dengan LAS dataset yang sudah difilter ke kelas 2.
2. Pilih elevasi sebagai nilai yang dirasterkan dan atur ukuran sel menjadi 1,0. **[CEK]**

Ringkasan, Latihan, dan Bantuan Bab 4

Ringkasan

- Point cloud dapat dimuat langsung di QGIS dan dilihat dalam 2D maupun 3D.
- Menurut standar ASPRS, kelas 2 adalah tanah (Ground) dan kelas 3 sampai 5 adalah vegetasi.
- Titik tanah diubah menjadi DTM dengan Export to raster.
- **[ArcGIS Pro / ArcMap]** LAS dataset, Classify LAS Ground, Extract LAS (kelas 2), dan LAS Dataset To Raster; semua alat LAS memerlukan 3D Analyst (kecuali bila disebut lain).

Latihan

1. Muat point cloud, buka tampilan 3D, lalu putar untuk melihat struktur tegakan.
2. Filter kelas 2 dan bandingkan jumlah titik sebelum dan sesudah.
3. Ekspor DTM dengan ukuran piksel 0,5; 1; dan 2 meter. Bandingkan tingkat detail dan lama prosesnya.
4. **[ArcGIS Pro / ArcMap]** Buat LAS dataset dari data yang sama, ekstrak kelas 2, lalu buat DTM berukuran sel 1 meter dan bandingkan dengan hasil QGIS.

Jika terjadi masalah

- Hasil filter kosong: data mungkin belum terklasifikasi, yang umum pada hasil fotogrametri drone. Periksa kelas pada properti layer; bila semua titik berkelas 0 atau 1, titik tanah perlu diklasifikasikan lebih dulu. **[CEK]**
- DTM tampak berlubang di bawah kanopi yang sangat rapat: titik tanah terlalu sedikit. Coba ukuran piksel yang lebih besar.
- **[QGIS]** Titik tanah belum terklasifikasi: pakai Classify ground points (SMRF), tersedia mulai QGIS 4.0. Parameternya perlu disetel dan hasilnya terbatas pada fotogrametri di hutan lebat.
- **[ArcGIS Pro / ArcMap]** Classify LAS Ground gagal: periksa bahwa LAS dataset memakai sistem koordinat terproyeksi. Keterbatasannya pada fotogrametri hutan lebat sama seperti SMRF.
- **[ArcMap]** Berkas .laz tidak terbaca: dokumentasi hanya menyebut .las dan .zlas; ubah format lebih dulu. **[CEK]**

Tabel Perbandingan Bab 4: QGIS, ArcGIS Pro, dan ArcMap 10.8

Kolom *Status* menjelaskan seberapa pasti padanan itu. **Terbukti**: dokumentasi resmi sudah dibaca penulis. **Kemungkinan**: berdasarkan informasi umum. **[CEK]**: perlu dicoba di perangkat lunaknya.

Langkah	QGIS	ArcGIS Pro	ArcMap 10.8	Status
4.1 Muat data	Seret berkas ke panel Layers	Create LAS Dataset lalu tambahkan ke peta	Create LAS Dataset lalu tambahkan ke peta	Terbukti: alat ada di Pro dan ArcMap
4.1 Format	.las dan .laz	.las, .zlas, .laz	.las dan .zlas (laz tidak disebut)	Terbukti dari dokumentasi masing-masing; laz di ArcMap [CEK]

Langkah	QGIS	ArcGIS Pro	ArcMap 10.8	Status
4.1 Tampilan 3D	New 3D Map View	Scene (tampilan 3D)	Aplikasi 3D terpisah (mis. ArcScene)	[CEK]
4.2 Klasifikasi tanah	(tidak ada langkah di naskah)	Classify LAS Ground	Classify LAS Ground	Terbukti ada di Pro dan ArcMap; butuh 3D Analyst
4.2 Filter kelas	Filter by classification	Extract LAS (kode klasifikasi 2)	Extract LAS (kode klasifikasi 2)	Terbukti ada; butuh 3D Analyst
4.3 Ekspor DTM	Export to raster (TIN)	LAS Dataset To Raster	LAS Dataset To Raster	Terbukti ada; Pro: Spatial Analyst atau 3D Analyst

Ilustrasi 4.5: Ringkasan ketersediaan padanan pada Bab 4

Ringkasan ketersediaan padanan pada Bab 4

	QGIS	ArcGIS Pro	ArcMap 10.8
Muat point cloud	Layer langsung (.las, .laz)	LAS dataset layer (LAS, zLAS, LAZ)	LAS dataset (.las, .zlas)
Filter tanah	Filter by classification	Extract LAS / Classify LAS Ground	Extract LAS / Classify LAS Ground
Ekspor DTM	Export to raster (TIN)	LAS Dataset To Raster	LAS Dataset To Raster
Ekstensi	Tanpa ekstensi	3D Analyst	3D Analyst

BAB 5: Klasifikasi Berbasis Objek (OBIA) & Machine Learning

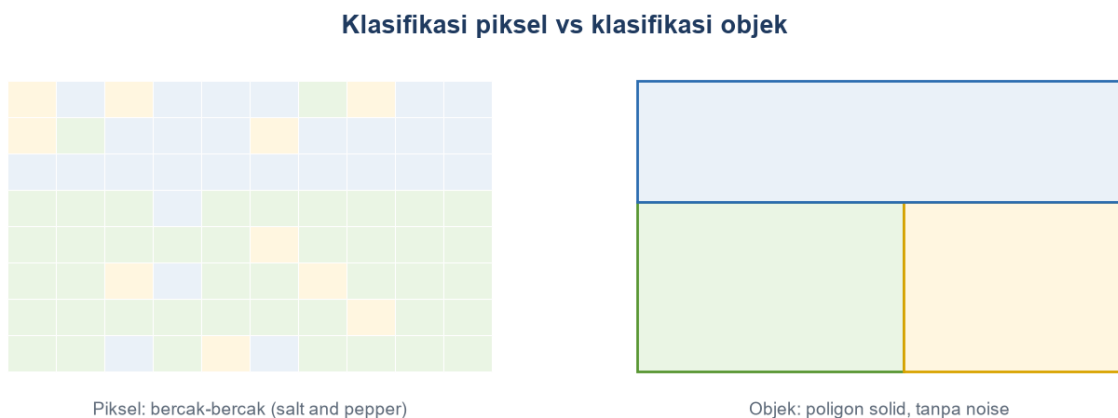
Konteks Riset/Bisnis: Klasifikasi piksel tradisional sering menghasilkan peta tutupan lahan yang "bercak-bercak" (*salt and pepper noise*). Untuk inventarisasi perkebunan kelapa sawit atau klasifikasi jenis tegakan hutan, Anda dapat beralih ke *Object-Based Image Analysis* (OBIA). Metode ini dikombinasikan dengan *Machine Learning* (*Random Forest*).

BAGIAN A | QGIS

5.1. Segmentasi Objek (Membuat Poligon Alami)

Pertama, pecah citra menjadi objek-objek poligon. Pemecahan ini berdasarkan kesamaan warna dan teksturnya.

Ilustrasi 5.1: Klasifikasi piksel dan klasifikasi objek



Langkah Eksekusi:

1. Buka **Processing Toolbox > OTB (Orfeo ToolBox) > Segmentation > Large-Scale Mean-Shift (LSMS)**.
2. **Spatial Radius:** 15 (Berapa jarak spasial piksel untuk digabungkan).
3. **Range Radius:** 10 (Seberapa mirip warna piksel agar bisa digabungkan).
4. **Min Region Size:** 50 (Ukuran terkecil objek).
5. **Output:** Vektor_Segmen.shp.

5.2. Ekstraksi Fitur Objek (Zonal Statistics)

Machine Learning butuh data untuk belajar. Hitung rata-rata nilai Merah, Hijau, Biru, dan NDVI untuk setiap poligon.

Langkah Eksekusi:

1. Buka **Processing Toolbox > Raster Analysis > Zonal Statistics**.
2. **Input Vector:** Vektor_Segmen.shp.
3. **Raster Layer:** Citra Komposit/NDVI Anda.
4. **Statistics to calculate:** Centang Mean, StDev (Standar Deviasi untuk tekstur), dan Variance.

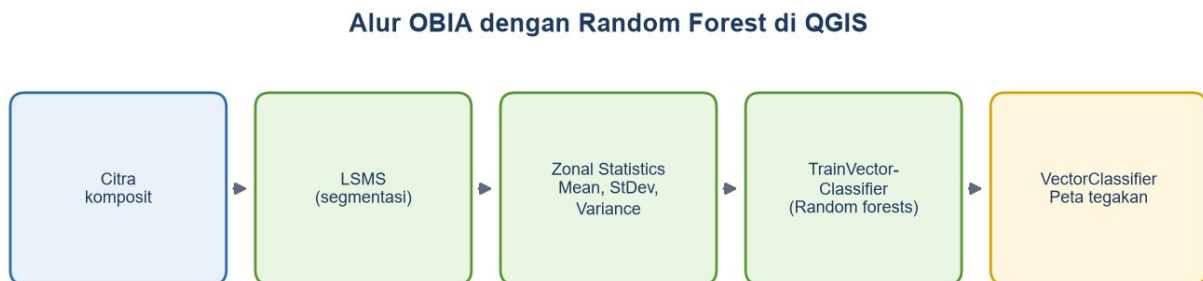
5.3. Melatih Machine Learning (Random Forest)

Anda akan mengajari komputer: "Jika poligon ini memiliki rata-rata NDVI sekian dan tekstur sekian, maka ini adalah Kelapa Sawit."

Langkah Eksekusi:

1. Buka Tabel Atribut Vektor_Segmen.shp. Tambahkan field ID_Kelas (1 = Sawit, 2 = Akasia, 3 = Belukar). Beri label secara manual pada sekitar 30-50 poligon sebagai data latih (*Training Data*).
2. Buka **Processing Toolbox > OTB > Learning > TrainVectorClassifier**.
3. **Input Vector Data:** Vektor_Segmen.shp.
4. **Classifier:** Pilih Random forests.
5. **Output Model:** Model_RF_Kebun.rf.
6. Terapkan model ini ke seluruh gambar dengan alat **VectorClassifier** di OTB. Hasilnya adalah peta klasifikasi bentuk lahan/tegakan hutan yang sangat rapi. Peta ini berbentuk poligon solid, tanpa ada *noise*.

Ilustrasi 5.2: Alur OBIA dengan Random Forest di QGIS



BAGIAN B | ArcGIS Pro

PERANGKAT DAN LISENSI: **Segment Mean Shift**, **Train Random Trees Classifier**, dan **Classify Raster** memerlukan ekstensi **Image Analyst** atau **Spatial Analyst**. Orfeo Toolbox (OTB) tidak diperlukan.

Di ArcGIS, segmentasi dan klasifikasi berjalan pada **raster tersegmentasi**: atribut tiap segmen (warna, rata-rata, deviasi standar, jumlah piksel, kekompakan, dan kepersegian) dapat dipakai langsung saat melatih model. Karena itu langkah *Zonal Statistics* pada Bagian A tidak diperlukan sebagai langkah terpisah.

5.1. Segmentasi Objek (ArcGIS Pro)

Langkah Eksekusi:

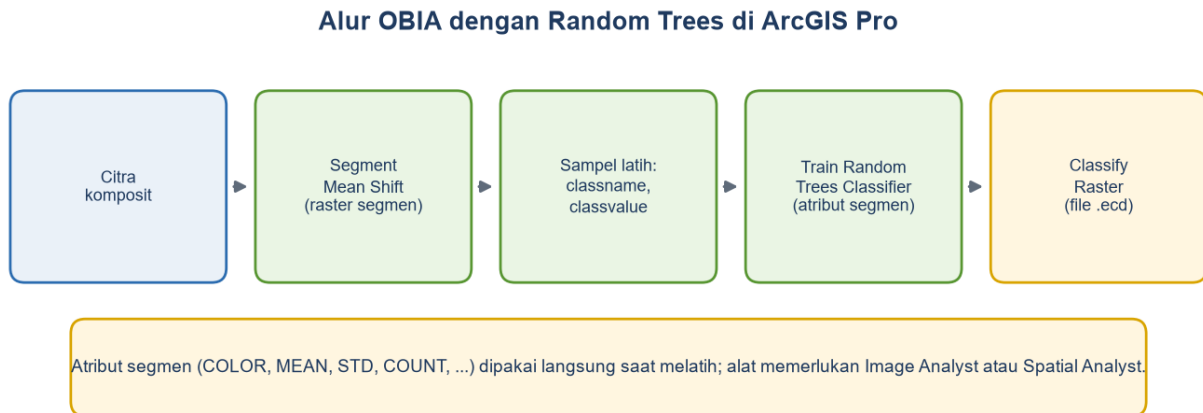
1. Jalankan **Segment Mean Shift** (kelompok *Classification and Pattern Recognition*). Input: citra komposit Anda.
2. **Spectral Detail** (1,0 sampai 20): seberapa penting perbedaan warna. Nilai tinggi membedakan fitur yang mirip. Padanan kasar *Range Radius* di OTB.
3. **Spatial Detail** (1,0 sampai 20): seberapa penting kedekatan piksel. Nilai tinggi menjaga fitur kecil yang bergerombol. Padanan kasar *Spatial Radius* di OTB.
4. **Minimum Segment Size In Pixels**: ukuran segmen terkecil. Segmen yang lebih kecil digabung dengan tetangganya.
5. Hasilnya berupa raster tersegmentasi 3-band 8-bit. **[CEK]** Padanan angka dengan nilai OTB (15, 10, 50) tidak sama persis; sesuaikan lewat percobaan.

5.2. Ekstraksi Fitur Objek (ArcGIS Pro)

Pada ArcGIS, fitur objek dihitung otomatis sebagai **atribut segmen** saat melatih model, sehingga tidak perlu langkah terpisah. Bila Anda tetap ingin statistik per poligon untuk keperluan lain, pakai **Zonal Statistics as Table**: raster desimal mendukung Mean, Maximum, Median, Minimum, Percentile, Range, Standard deviation, dan Sum. Varians tidak tersedia, tetapi varians sama dengan kuadrat deviasi standar.

5.3. Melatih Machine Learning (ArcGIS Pro)

Ilustrasi 5.3: Alur OBIA dengan Random Trees di ArcGIS Pro



Langkah Eksekusi:

1. Buat poligon sampel latih di atas raster tersegmentasi. Tabel sampel wajib memiliki field **classname** dan **classvalue** (misalnya 1 = Sawit, 2 = Akasia, 3 = Belukar). Beri label pada sekitar 30 sampai 50 poligon.
2. Jalankan **Train Random Trees Classifier**. Input raster: raster tersegmentasi. Atur **Max Number of Trees** dan **Max Tree Depth**. Pilih atribut segmen yang dipakai (misalnya MEAN dan STD). Keluarannya berupa file model **.ecd**.
3. Jalankan **Classify Raster** dengan raster yang sama dan file **.ecd** tadi. Masukan harus sesuai dengan yang dipakai saat melatih.
4. Hasilnya adalah peta klasifikasi tegakan berbentuk objek yang rapi. Bila perlu poligon, ubah dengan **Raster to Polygon**. [CEK]

PERHATIAN [CEK]: Hasil tiap versi dapat berbeda dari OTB karena algoritma segmentasi dan pelatihannya tidak sama. Bandingkan hasilnya dengan data uji, bukan dengan angka parameter.

BAGIAN C | ArcMap 10.8

PERANGKAT DAN LISENSI: ArcMap memakai **Python 2.7**. **Segment Mean Shift**, **Train Random Trees Classifier**, dan **Classify Raster** memerlukan ekstensi **Spatial Analyst**. OTB tidak diperlukan.

5.1. Segmentasi Objek (ArcMap)

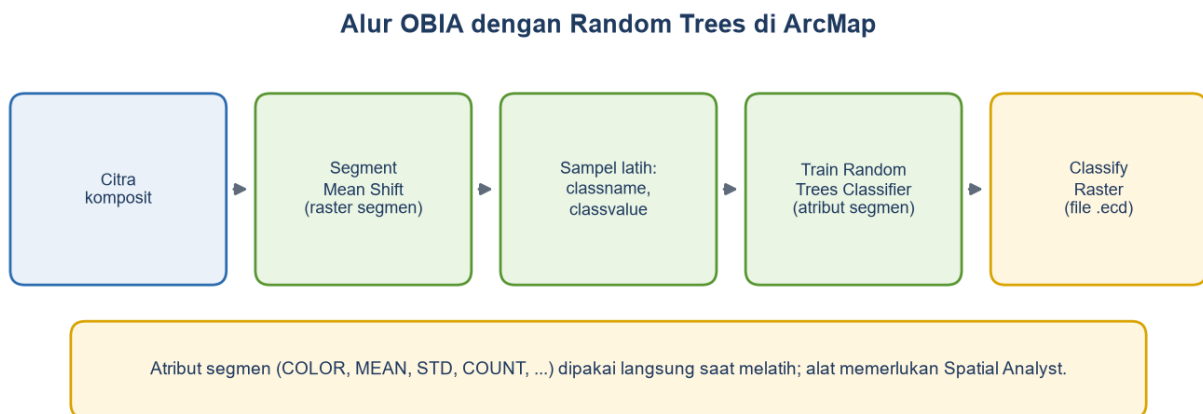
1. Jalankan **Segment Mean Shift** (Spatial Analyst Tools) pada citra komposit.
2. Atur **Spectral Detail** (1,0 sampai 20), **Spatial Detail** (1,0 sampai 20), dan **Minimum Segment Size In Pixels**. Maknanya sama seperti pada Bagian B.
3. Hasilnya berupa citra tersegmentasi 3-band 8-bit. Nilai OTB (15, 10, 50) tidak dapat dipindahkan langsung; sesuaikan lewat percobaan. **[CEK]**

5.2. Ekstraksi Fitur Objek (ArcMap)

Sama seperti Bagian B: atribut segmen (**COLOR, MEAN, STD, COUNT, COMPACTNESS, RECTANGULARITY**) tersedia pada alat pelatihan, sehingga **Zonal Statistics as Table** tidak wajib. Untuk statistik per poligon, alat ini mendukung Mean, STD, Min, Max, Range, dan Sum; Majority, Minority, Median, dan Variety hanya dihitung untuk raster bilangan bulat.

5.3. Melatih Machine Learning (ArcMap)

Ilustrasi 5.4: Alur OBIA dengan Random Trees di ArcMap



1. Buat poligon sampel latih di atas citra tersegmentasi, dengan field **classname** dan **classvalue**. Dokumentasi menyarankan 20 sampel atau lebih per kelas.
2. Jalankan **Train Random Trees Classifier**: tentukan raster tersegmentasi, sampel latih, file keluaran **.ecd**, **Max Number of Trees**, **Max Tree Depth**, dan atribut segmen yang dipakai.
3. Jalankan **Classify Raster** dengan file **.ecd** tadi.
4. Ubah ke poligon dengan **Raster to Polygon** bila perlu. **[CEK]**

Ringkasan, Latihan, dan Bantuan Bab 5

Ringkasan

- OBIA memecah citra menjadi objek sehingga hasil klasifikasi tidak "bercak-bercak".
- Anda menghitung fitur tiap objek (rata-rata dan sebaran nilai) dengan Zonal Statistics.
- Random Forest dilatih dari poligon yang sudah diberi label, lalu diterapkan ke seluruh poligon.
- **[ArcGIS Pro / ArcMap]** Segment Mean Shift, sampel latih (classname dan classvalue), Train Random Trees Classifier dengan atribut segmen, dan Classify Raster dengan file .ecd.

Latihan

1. Segmentasi satu citra kecil dengan parameter di buku, lalu ubah Spatial Radius dan Range Radius. Amati perubahan ukuran objek.
2. Beri label 40 poligon, lalu sisihkan 10 poligon sebagai data uji (jangan dipakai untuk melatih).
3. Terapkan model, bandingkan prediksi dengan label pada 10 poligon uji, dan hitung persentase yang benar.
4. **[ArcGIS Pro / ArcMap]** Ulangi dengan Segment Mean Shift: ubah Spectral Detail dan Spatial Detail, lalu amati perubahan ukuran segmen.

Jika terjadi masalah

- Menu OTB tidak muncul: OTB belum terpasang atau belum diatur.
- Satu kelas hasilnya buruk: tambah poligon latih untuk kelas itu dan usahakan jumlah tiap kelas seimbang.
- **[ArcGIS Pro / ArcMap]** Classify Raster menolak masukan: masukan harus sesuai dengan yang dipakai saat melatih (file .ecd).
- **[ArcGIS Pro / ArcMap]** Pelatihan gagal karena tabel sampel: pastikan ada field classname dan classvalue.

Tabel Perbandingan Bab 5: QGIS, ArcGIS Pro, dan ArcMap 10.8

Kolom *Status* menjelaskan seberapa pasti padanan itu. **Terbukti**: dokumentasi resmi sudah dibaca penulis. **Kemungkinan**: berdasarkan informasi umum. **[CEK]**: perlu dicoba di perangkat lunaknya.

Langkah	QGIS	ArcGIS Pro	ArcMap 10.8	Status
5.1 Segmentasi	OTB LSMS (Spatial Radius 15, Range Radius 10, Min Region Size 50)	Segment Mean Shift (Spectral Detail, Spatial Detail, Minimum Segment Size)	Segment Mean Shift (parameter sama)	Terbukti ada; Pro: Image Analyst atau Spatial Analyst. Padanan angka [CEK]
5.2 Fitur objek	Zonal Statistics (Mean, StDev, Variance)	Atribut segmen (COLOR, MEAN, STD, ...); atau Zonal Statistics as Table	Atribut segmen; atau Zonal Statistics as Table	Terbukti. Variance tidak tersedia di Zonal Statistics as Table
5.3 Melatih	TrainVectorClassifier (Random forests) menghasilkan .rf	Train Random Trees Classifier menghasilkan .ecd	Train Random Trees Classifier menghasilkan .ecd	Terbukti ada

Langkah	QGIS	ArcGIS Pro	ArcMap 10.8	Status
5.3 Menerapkan	VectorClassifier	Classify Raster	Classify Raster	Terbukti ada
5.3 Poligon hasil	Hasil poligon langsung	Raster to Polygon (bila perlu)	Raster to Polygon (bila perlu)	Terbukti ada; semua level lisensi

Ilustrasi 5.5: Ringkasan ketersediaan padanan pada Bab 5

Ringkasan ketersediaan padanan pada Bab 5

	QGIS	ArcGIS Pro	ArcMap 10.8
Segmentasi	OTB LSMS	Segment Mean Shift	Segment Mean Shift
Fitur objek	Zonal Statistics	Atribut segmen (MEAN, STD, ...)	Atribut segmen (MEAN, STD, ...)
Melatih model	TrainVector-Classifier (RF)	Train Random Trees Classifier	Train Random Trees Classifier
Menerapkan model	VectorClassifier	Classify Raster	Classify Raster

BAB 6: Otomatisasi Alur Kerja & Atlas Dinamis Tingkat Lanjut

Konteks Riset/Bisnis: Setiap bulan, Anda menerima puluhan data deforestasi, kebakaran hutan, atau panen dari berbagai Kesatuan Pengelolaan Hutan (KPH). Pemimpin ingin laporan komprehensif berformat PDF untuk setiap KPH tanpa Anda harus menata teks dan peta secara manual.

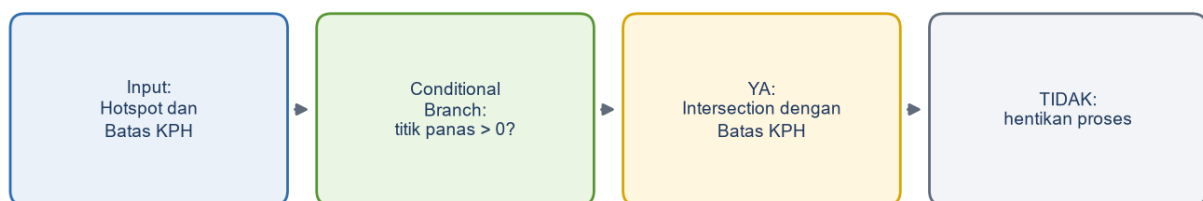
BAGIAN A | QGIS

6.1. Graphical Modeler dengan Percabangan Logika

Anda akan membangun mesin pemrosesan yang bisa berpikir.

Ilustrasi 6.1: Model dengan percabangan di Graphical Modeler

Model dengan percabangan di Graphical Modeler



Langkah Eksekusi:

1. Buka **Processing > Graphical Modeler**.
2. Tambahkan **Vector Layer** (*Input: Data Titik Panas/Hotspot*) dan **Vector Layer** (*Input: Batas KPH*).
3. **Tambahkan Logika:** Cari algoritma Conditional Branch. Atur: "JIKA jumlah titik panas > 0, MAKA jalankan algoritma Intersection dengan Batas KPH. JIKA TIDAK, hentikan proses."
4. Simpan model sebagai "Kalkulator_Karhutla".

6.2. Print Layout & Ekspor Laporan Otomatis (Atlas)

Langkah Eksekusi (Desain & Pengaturan):

1. Buka **Project > New Print Layout**. Beri nama "Laporan Kebakaran Hutan".
2. Tambahkan **Add Map**.
3. Di panel kanan, buka **Atlas**, centang **Generate an Atlas**. Pilih **Coverage Layer: Batas_KPH**.

4. Klik peta di kanvas. Pada **Item Properties**, centang **Controlled by atlas** dan atur margin 15% agar peta menyesuaikan bentuk setiap KPH.

Ilustrasi 6.2: Alur Atlas: satu halaman per KPH

Alur Atlas: satu halaman per KPH



6.3. Teks Dinamis Berbasis Expression Builder (Inti Laporan)

Tambahkan **Add Label**. Masukkan kode ekspresi ini. Kode ini menghitung secara dinamis *Hotspot* (titik api) yang jatuh di dalam poligon KPH pada halaman yang sedang aktif.

EKSPRESI QGIS

```

'LAPORAN KEBAKARAN HUTAN KPH ' || upper("NAMA_KPH") || '\n' ||
'-----' || '\n' ||
'Total Luas KPH: ' || round($area / 10000, 2) || ' Ha \n' ||
'Jumlah Titik Api Aktif: ' ||
aggregate(
  layer:= 'Hotspot_Nasional',
  aggregate:= 'count',
  expression:= "ID_Titik",
  filter:= intersects($geometry, geometry(@parent))
) || ' Titik.'
  
```

BAGAIMANA INI BEKERJA? Fungsi `aggregate()` melakukan *Spatial Join* secara *virtual*. QGIS akan mengambil layer `Hotspot_Nasional`, menghitung (count) jumlahnya, tetapi memfilternya (`filter:= intersects`) HANYA yang menyentuh batas `geometry(@parent)` alias batas KPH di halaman atlas saat ini.

6.4. Ekspor Massal Laporan Sektor

1. Di panel *Atlas generation*, atur **Output filename expression** menjadi `"NAMA_KPH" || '_Laporan_Bulan_Ini'`.
2. Klik ikon **Export Atlas as PDF** di atas layar.
3. Pergilah ke kantin dan minum kopi. Saat Anda kembali, QGIS sudah membuat 50 halaman PDF berisi peta yang di-zoom ke tiap KPH, lengkap dengan angka luas dan deteksi yang akurat di setiap halaman.

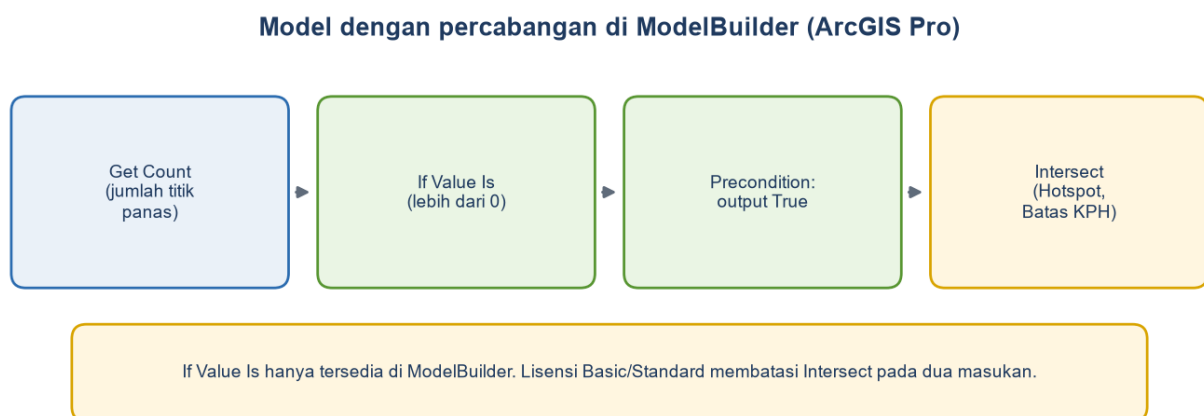
BAGIAN B | ArcGIS Pro

PERANGKAT DAN LISENSI: **Get Count** dan **Spatial Join** tersedia pada semua level lisensi. **Intersect** pada lisensi Basic dan Standard dibatasi dua masukan (di sini cukup dua). **If Value Is** hanya tersedia di ModelBuilder, tidak di panel Geoprocessing atau Python. Pro memakai **Python 3**.

6.1. Model dengan Percabangan Logika (ArcGIS Pro)

Padanan Graphical Modeler adalah **ModelBuilder**. Percabangan dibuat dengan alat yang menghasilkan nilai Boolean, lalu Boolean itu dipasang sebagai *precondition* bagi alat berikutnya.

Ilustrasi 6.3: Model dengan percabangan di ModelBuilder (ArcGIS Pro)



Langkah Eksekusi:

1. Buka ModelBuilder dan buat model baru. **[CEK]**
2. Tambahkan dua masukan: layer titik panas (Hotspot_Nasional) dan batas KPH (Batas_KPH). Jadikan keduanya parameter model.
3. Tambahkan **Get Count** pada layer titik panas. Keluarannya adalah jumlah baris (bilangan bulat).
4. Tambahkan **If Value Is** dengan syarat lebih besar dari 0. Alat ini menghasilkan dua keluaran Boolean: **True** dan **False**.
5. Jadikan keluaran **True** sebagai *precondition* untuk **Intersect** antara titik panas dan Batas_KPH. Bila titik panas nol, Intersect tidak berjalan.
6. Simpan model sebagai "Kalkulator_Karhutla".

6.2. Print Layout dan Ekspor Laporan Otomatis (ArcGIS Pro)

Padanan Atlas adalah **Map Series**: satu halaman layout untuk setiap fitur pada layer indeks.

Langkah Eksekusi (Desain dan Pengaturan):

1. Buat layout baru dengan nama "Laporan Kebakaran Hutan". **[CEK]**
2. Tambahkan bingkai peta (*map frame*).

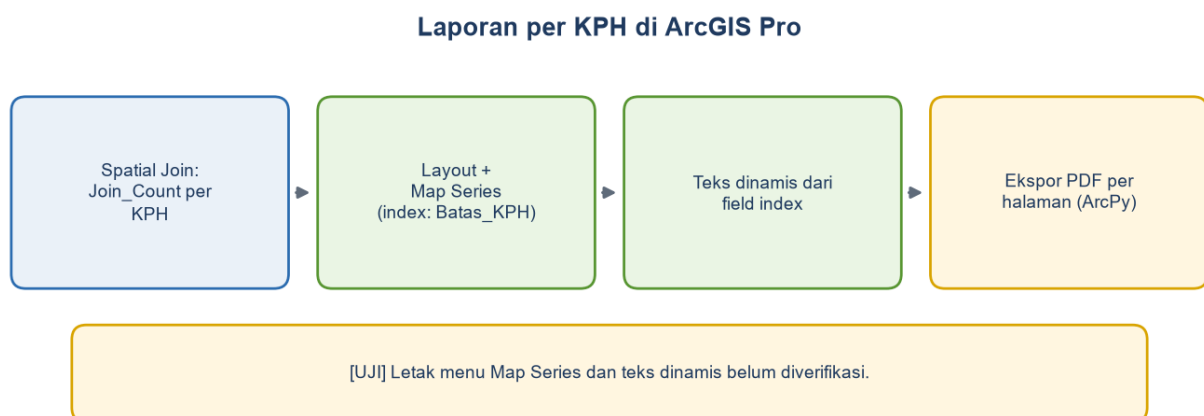
3. Aktifkan **Map Series** bertipe spasial dan pilih layer indeks **Batas_KPH**. [CEK]
4. Atur agar peta menyesuaikan bentuk tiap KPH dengan margin sekitar 15 persen. [CEK]

6.3. Teks Dinamis (ArcGIS Pro)

Pada QGIS, jumlah titik api dihitung di dalam ekspresi teks dengan fungsi `aggregate()`. Pada ArcGIS, jumlah itu dihitung lebih dulu, lalu dipakai sebagai atribut KPH.

1. Jalankan **Spatial Join** dengan target **Batas_KPH** dan join **Hotspot_Nasional** (`JOIN_ONE_TO_ONE`, `KEEP_ALL`, `INTERSECT`). Jumlah titik per KPH tersimpan di field **Join_Count**. [CEK] untuk nama field di Pro.
2. Tambahkan field **Luas_Ha** berisi luas KPH dalam hektar, misalnya dengan **Calculate Geometry**. [CEK]
3. Pakai **Batas_KPH** hasil **Spatial Join** sebagai layer indeks Map Series.
4. Tambahkan elemen **Dynamic Text** pada layout dan pilih atribut layer indeks (**NAMA_KPH**, **Luas_Ha**, dan **Join_Count**). [CEK]

Ilustrasi 6.4: Laporan per KPH di ArcGIS Pro



TIPS: Pro juga mendukung ekspresi Arcade untuk teks dinamis kustom. Cara kerja dan sintaksnya perlu Anda coba sendiri, sehingga bagian ini memakai atribut yang dihitung lebih dulu. [CEK]

6.4. Ekspor Massal Laporan Sektoral (ArcGIS Pro)

Pro mengekspor Map Series lewat dialog ekspor layout. Untuk satu berkas PDF per KPH dengan nama dari field **NAMA_KPH**, pakai skrip berikut:

```

PYTHON (ARCPY, PYTHON 3)
# [SKRIP 6.1-Pro: Ekspor PDF per KPH dari Map Series]
# Penulis: Badar Mubarak Yogaswara
# [CEK] Cocokkan dengan versi Anda.
import os
import arcpy

aprx = arcpy.mp.ArcGISProject("CURRENT")
  
```

```
lyt = aprx.listLayouts("Laporan Kebakaran Hutan")[0]
ms = lyt.mapSeries
folder_out = r"D:\Laporan"

for no in range(1, ms.pageCount + 1):
    ms.currentPageNumber = no
    nama = ms.pageRow.NAMA_KPH
    lyt.exportToPDF(os.path.join(folder_out, f"{nama}_Laporan_Bulan_Ini.pdf"))
    print(f"Selesai: {nama}")
```

BEDAH KODE (Bagi Praktisi Python): MapSeries menyediakan pageCount, currentPageNumber, dan pageRow (atribut fitur indeks saat ini). Metode exportToPDF masih bisa dipakai tetapi digantikan metode export sejak ArcGIS Pro 3.4. **[CEK]**

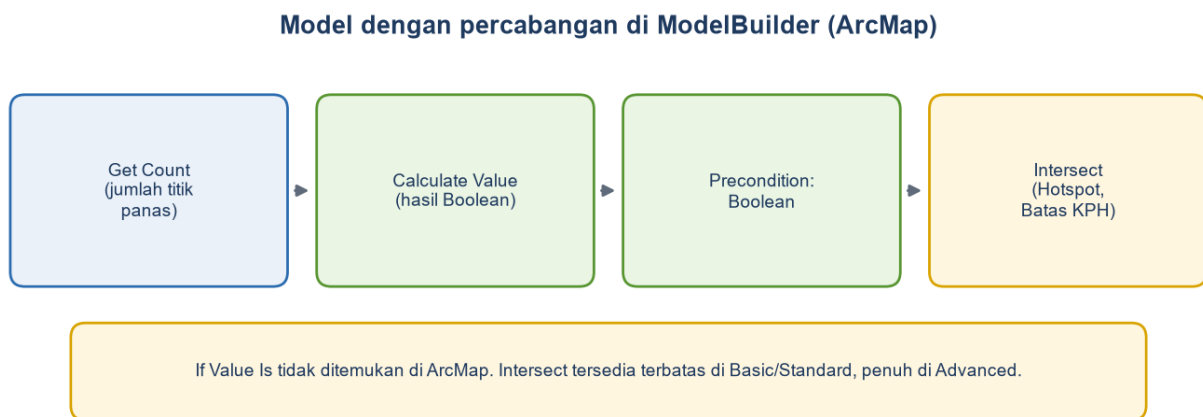
BAGIAN C | ArcMap 10.8

PERANGKAT DAN LISENSI: ArcMap memakai **Python 2.7**. **Get Count** dan **Spatial Join** tersedia pada semua level lisensi. **Intersect** tersedia terbatas pada Basic dan Standard, dan penuh pada Advanced.

6.1. Model dengan Percabangan Logika (ArcMap)

ModelBuilder di ArcMap juga mendukung percabangan lewat nilai Boolean sebagai *precondition*. Alat **If Value Is** tidak ditemukan di ArcMap, sehingga Boolean dihasilkan dengan **Calculate Value**.

Ilustrasi 6.5: Model dengan percabangan di ModelBuilder (ArcMap)



1. Buat model baru dengan dua masukan: Hotspot_Nasional dan Batas_KPH.
2. Tambahkan **Get Count** pada layer titik panas.
3. Tambahkan **Calculate Value** yang menguji apakah jumlahnya lebih dari 0, dengan tipe keluaran Boolean. **[CEK]** untuk bentuk ekspresinya.
4. Jadikan Boolean itu *precondition* untuk **Intersect** antara titik panas dan Batas_KPH.
5. Simpan model sebagai "Kalkulator_Karhutla".

6.2. Print Layout dan Ekspor Laporan Otomatis (ArcMap)

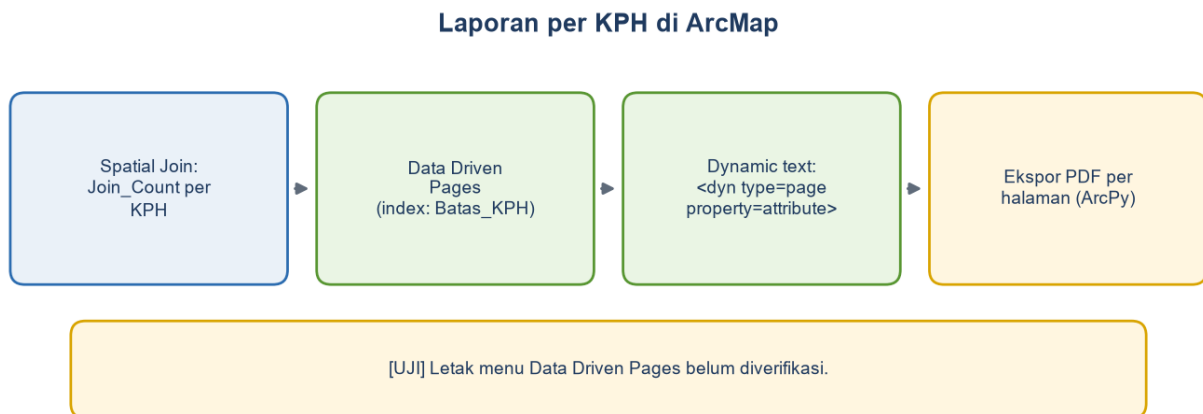
Padanan Atlas adalah **Data Driven Pages**. Fitur ini hanya dapat diekspor dari tampilan layout.

1. Buka tampilan layout dan siapkan peta.
2. Atur **Data Driven Pages**: pilih layer indeks **Batas_KPH** dan field nama **NAMA_KPH**. **[CEK]**
3. Atur agar peta menyesuaikan bentuk tiap KPH dengan margin sekitar 15 persen. **[CEK]**

6.3. Teks Dinamis (ArcMap)

ArcMap memakai **dynamic text** berbentuk tag mirip HTML. Atribut layer indeks dipanggil dengan tag di bawah ini. Seperti pada Bagian B, jumlah titik api dihitung lebih dulu dengan **Spatial Join** (field **Join_Count**), dan luas disimpan di field **Luas_Ha** (misalnya dengan Calculate Geometry). **[CEK]**

Ilustrasi 6.6: Laporan per KPH di ArcMap



Tambahkan elemen teks pada layout dan isi dengan teks berikut:

TEKS DINAMIS ARCMAP

```

LAPORAN KEBAKARAN HUTAN KPH <dyn type="page" property="attribute" field="NAMA_KPH"/>
-----
Total Luas KPH: <dyn type="page" property="attribute" field="Luas_Ha"/> Ha
Jumlah Titik Api Aktif: <dyn type="page" property="attribute" field="Join_Count"/> Titik.
  
```

PERHATIAN: Bentuk tag atribut dikutip dari dokumentasi ArcMap tentang dynamic text Data Driven Pages; nama field di atas adalah nama contoh. ArcMap tidak punya fungsi huruf besar seperti upper(), jadi simpan NAMA_KPH dalam huruf besar bila ingin tampil begitu. **[CEK]**

6.4. Ekspor Massal Laporan Sektoral (ArcMap)

Lewat menu **File > Export Map** dan format PDF, tab *Pages* memilih halaman yang diekspor, tetapi hasilnya satu berkas. Untuk satu PDF per KPH, pakai skrip berikut:

PYTHON (ARCPY, PYTHON 2.7)

```

# [SKRIP 6.1-ArcMap: Ekspor PDF per KPH dari Data Driven Pages]
# Penulis: Badar Mubarak Yogaswara
# [CEK] Cocokkan dengan versi Anda.
from __future__ import print_function # agar print() berlaku di Python 2.7
import os
import arcpy

mxd = arcpy.mapping.MapDocument("CURRENT")
ddp = mxd.dataDrivenPages
folder_out = r"D:\Laporan"

for no in range(1, ddp.pageCount + 1):
    ddp.currentPageID = no
    nama = ddp.pageRow.getValue("NAMA_KPH")
    arcpy.mapping.ExportToPDF(mxd, os.path.join(folder_out, nama + "_Laporan_Bulan_Ini.pdf"))
    print("Selesai: {}".format(nama))

del mxd
  
```

BEDAH KODE (Bagi Praktisi Python): DataDrivenPages menyediakan pageCount, currentPageID, dan pageRow (baris layer indeks untuk halaman aktif). Pola mengekspor satu halaman per berkas dikutip dari contoh di dokumentasi ArcMap. **[CEK]**

Ringkasan, Latihan, dan Bantuan Bab 6

Ringkasan

- Graphical Modeler dapat bercabang: proses berjalan hanya bila syarat terpenuhi.
- Atlas membuat satu halaman peta untuk setiap fitur pada layer cakupan.
- Fungsi aggregate() menghitung titik panas yang jatuh di dalam tiap KPH.
- **[ArcGIS Pro]** ModelBuilder dengan Get Count dan If Value Is; Map Series; jumlah titik dihitung lebih dulu dengan Spatial Join; ekspor lewat skrip ArcPy.
- **[ArcMap]** ModelBuilder dengan Get Count dan Calculate Value; Data Driven Pages dengan tag dynamic text; ekspor lewat skrip ArcPy Python 2.7.

Latihan

1. Buat model sederhana dengan percabangan: bila jumlah titik panas 0, hentikan proses.
2. Buat atlas untuk 3 KPH contoh dan ekspor menjadi PDF.
3. Ubah ekspresi nama berkas keluaran agar memuat tanggal hari ini, misalnya dengan `format_date(now(), 'yyyy-MM-dd')`. [CEK]
4. **[ArcGIS Pro / ArcMap]** Buat Map Series atau Data Driven Pages untuk 3 KPH contoh, lalu jalankan skrip ekspor dan periksa nama berkas PDF.

Jika terjadi masalah

- Label menampilkan 0 titik padahal ada titik api: periksa bahwa nama layer persis Hotspot_Nasional dan kedua layer memakai CRS yang sama.
- PDF hanya berisi satu halaman: Atlas belum diaktifkan atau Coverage Layer belum dipilih.
- **[ArcGIS Pro / ArcMap]** Join_Count bernilai 0 padahal ada titik api: periksa bahwa kedua layer memakai CRS yang sama dan opsi kecocokan Spatial Join adalah INTERSECT. [CEK]
- **[ArcGIS Pro / ArcMap]** Skrip ekspor gagal: pastikan layout atau Data Driven Pages sudah aktif dan nama field indeks sesuai.
- **[ArcMap]** Intersect tidak dapat dijalankan penuh: lisensi Basic dan Standard membatasinya.

Tabel Perbandingan Bab 6: QGIS, ArcGIS Pro, dan ArcMap 10.8

Kolom *Status* menjelaskan seberapa pasti padanan itu. **Terbukti**: dokumentasi resmi sudah dibaca penulis. **Kemungkinan**: berdasarkan informasi umum. **[CEK]**: perlu dicoba di perangkat lunaknya.

Langkah	QGIS	ArcGIS Pro	ArcMap 10.8	Status
6.1 Alat model	Graphical Modeler	ModelBuilder	ModelBuilder	Terbukti ada di Pro dan ArcMap
6.1 Percabangan	Conditional Branch	Get Count + If Value Is + precondition	Get Count + Calculate Value (Boolean) + precondition	Terbukti: Pro (If Value Is hanya di ModelBuilder); ArcMap: tanpa If Value Is [CEK]
6.1 Overlay	Intersection	Intersect (Basic/Standard: maks. dua masukan)	Intersect (terbatas di Basic/Standard)	Terbukti ada

Langkah	QGIS	ArcGIS Pro	ArcMap 10.8	Status
6.2 Laporan per fitur	Print Layout > Atlas	Layout > Map Series	Data Driven Pages	Terbukti dari panduan migrasi Esri; letak menu [CEK]
6.3 Teks dinamis	Ekspresi dengan aggregate()	Dynamic Text dari atribut indeks (Join_Count dihitung dulu)	Tag <dyn type="page" property="attribute" field="..." />	ArcMap: terbukti dari dokumentasi; Pro: kemungkinan [CEK]
6.4 Ekspor massal	Export Atlas as PDF	Dialog ekspor + skrip MapSeries	Export Map + skrip DataDrivenPages	Terbukti: properti kelas ada; skrip perlu Anda coba [CEK]

Ilustrasi 6.7: Ringkasan ketersediaan padanan pada Bab 6

Ringkasan ketersediaan padanan pada Bab 6

	QGIS	ArcGIS Pro	ArcMap 10.8
Model bercabang	Conditional Branch	If Value Is + precondition	Calculate Value + precondition
Layout per fitur	Atlas	Map Series	Data Driven Pages
Teks dinamis	Expression (aggregate)	Dynamic text (field index)	Dynamic text <dyn ...>
Ekspor massal	Export Atlas as PDF	ArcPy MapSeries	ArcPy mapping DataDrivenPages

Lanjut belajar

Seri berikutnya adalah **M2: Basis Data Spasial dan Analisis Lanjut**. Pindai kode QR ini untuk membuka halaman seri tersebut dan unduhannya.



Glosarium

Glosarium ini menjelaskan istilah teknis yang dipakai di buku,urut abjad. Bila Anda menemukan istilah yang belum dikenal, carilah di sini lalu buka bab yang disebut di akhir tiap entri.

Akurasi keseluruhan (overall accuracy): persentase titik lapangan yang kelasnya sama dengan kelas hasil klasifikasi citra. Di buku ini dihitung dari jumlah titik cocok dibagi jumlah seluruh titik. *Bab 2.*

ArcPy: paket Python milik ArcGIS untuk mengotomatiskan pekerjaan di ArcGIS Pro (Python 3) dan ArcMap (Python 2.7). *Sebelum Mulai, Bab 1.*

Atlas: fitur Print Layout QGIS yang membuat satu halaman peta untuk setiap fitur pada layer cakupan, misalnya satu halaman per KPH. Padanannya adalah Map Series di ArcGIS Pro dan Data Driven Pages di ArcMap. *Bab 6.*

Atribusi (attribution): pencantuman sumber data atau citra pada hasil kerja Anda. Citra EOX wajib diberi atribusi yang terlihat di mana pun ia ditampilkan. *Bab 1.*

Attachment (lampiran): jenis isian untuk menyimpan foto pada titik survei. Di QGIS berupa Widget Type Attachment, di ArcGIS lewat Enable Attachments. *Bab 2.*

Bare earth (tanah murni): permukaan tanah tanpa pohon dan bangunan. Dalam point cloud, ini berupa titik berkelas 2 (Ground). *Bab 4.*

Canopy Height Model (CHM): raster tinggi pohon, diperoleh dengan mengurangkan DTM dari DSM. Tanah kosong bernilai 0. *Bab 3.*

Citra komposit (composite image): citra gabungan beberapa band yang dipakai sebagai bahan segmentasi. *Bab 5.*

Classify ground points (klasifikasi titik tanah): alat QGIS 4.0 ke atas yang menandai titik tanah pada point cloud yang belum terklasifikasi, berbasis algoritma SMRF. Hasilnya terbatas pada fotogrametri drone di hutan lebat. *Bab 4.*

Clipping (pemotongan): memotong citra atau layer sesuai batas poligon. Di Bab 1 pemotongan dilakukan dengan alat GDAL atau Extract by Mask. *Pendahuluan, Bab 1.*

COG (Cloud-Optimized GeoTIFF): format citra yang tersusun agar bisa dibaca sebagian dari server cloud. Anda hanya menarik piksel yang tampil di layar. *Pendahuluan, Bab 1.*

Conditional Branch (percabangan bersyarat): algoritma di Graphical Modeler QGIS yang menjalankan langkah berikutnya hanya bila syarat terpenuhi, misalnya bila jumlah titik panas lebih dari 0. *Bab 6.*

CRS (Coordinate Reference System): sistem koordinat suatu peta atau layer. Perhitungan luas dan jarak yang akurat memerlukan CRS berbasis meter, seperti UTM, bukan derajat. *Bab 1.*

Data Driven Pages: fitur ArcMap yang membuat satu halaman layout untuk setiap fitur pada layer indeks; padanan Atlas di QGIS. *Bab 6.*

Data latih (training data): poligon yang sudah Anda beri label kelas, dipakai untuk mengajari model machine learning. Buku menyarankan sekitar 30 sampai 50 poligon. *Bab 5.*

Data sintetis (synthetic data): data contoh buatan untuk latihan di buku ini. Hasil analisisnya bukan kondisi lapangan nyata. *Sebelum Mulai.*

Domain: daftar nilai sah untuk sebuah field di geodatabase ArcGIS, sehingga surveyor hanya memilih isian yang diizinkan. Padanan Value Map di QGIS. *Bab 2.*

DSM (Digital Surface Model): model elevasi yang merekam permukaan teratas, yaitu pucuk pohon sekaligus permukaan tanah. *Bab 3.*

DTM (Digital Terrain Model): model elevasi yang hanya merekam tanah murni. Bisa dibuat dari titik tanah point cloud. *Bab 3, 4.*

Ekstensi (extension): tambahan berlisensi di ArcGIS, misalnya Spatial Analyst, 3D Analyst, atau Image Analyst. Sebagian alat hanya jalan bila ekstensinya tersedia. *Pendahuluan.*

EPSG: kode angka yang menamai sebuah CRS. Contohnya EPSG:4326 untuk WGS 84 dalam derajat dan EPSG:32749 untuk WGS 84 / UTM zone 49S. *Bab 1.*

Expression Builder (pembuat ekspresi): jendela QGIS untuk menulis ekspresi, misalnya teks dinamis pada label laporan atlas. *Bab 6.*

Field Maps (ArcGIS Field Maps): aplikasi lapangan Esri untuk pengumpulan data di ponsel. Ia hanya mengedit layer yang di-host di ArcGIS Online atau Enterprise. *Bab 2.*

Focal Statistics: alat ArcGIS yang menghitung statistik pada lingkungan di sekitar tiap sel. Dengan statistik MAXIMUM, alat ini dipakai untuk mencari pucuk pohon. *Bab 3.*

GDAL: pustaka pengolah data spasial yang menyediakan algoritma native untuk memotong dan mengolah raster di QGIS. *Pendahuluan, Bab 1.*

GeoPackage (.gpkg): format berkas data spasial yang ringan dan menjadi pengganti modern Shapefile. Dipakai untuk menyimpan titik survei. *Bab 2.*

Graphical Modeler: alat QGIS untuk menyusun langkah pemrosesan menjadi model berupa diagram, termasuk dengan percabangan. Padanannya di ArcGIS adalah ModelBuilder. *Bab 6.*

Ground-truthing (validasi lapangan): mencocokkan hasil analisis citra dengan kondisi nyata di lapangan. *Bab 2.*

Hosted feature layer: layer yang diterbitkan ke ArcGIS Online atau Enterprise, sehingga dapat diedit di Field Maps. *Bab 2.*

Image service: citra yang disajikan server dan ditampilkan di peta tanpa menyalin seluruh datanya. Ini alternatif terdekat STAC di ArcMap. *Bab 1.*

KPH (Kesatuan Pengelolaan Hutan): wilayah pengelolaan hutan. Di buku, batas KPH dipakai sebagai area kajian dan sebagai layer cakupan laporan. *Bab 6.*

LAS dataset: rujukan di ArcGIS ke satu atau banyak berkas point cloud (.las, .zlas, atau .laz). Dibuat dengan alat Create LAS Dataset. *Bab 4.*

LiDAR: pemindaian dengan laser yang menghasilkan point cloud. Sebagian titiknya dapat menembus sela kanopi hingga ke tanah. *Bab 4.*

Local maxima (maksimum lokal): titik tertinggi pada tiap gundukan kanopi di CHM. Tiap titik dianggap satu pucuk pohon. *Bab 3.*

Machine learning (pembelajaran mesin): cara komputer belajar dari contoh berlabel agar bisa menentukan kelas objek baru, misalnya sawit atau akasia. *Bab 5.*

Map Series: fitur layout ArcGIS Pro yang membuat satu halaman untuk setiap fitur pada layer indeks; padanan Atlas di QGIS. *Bab 6.*

Mergin Maps: alternatif QFieldSync untuk menyinkronkan proyek QGIS ke perangkat seluler. *Bab 2.*

ModelBuilder: alat ArcGIS untuk menyusun model pemrosesan. Percabangan dibuat dengan nilai Boolean yang dipasang sebagai precondition. *Bab 6.*

Native (bawaan): algoritma yang ada di dalam QGIS atau GDAL tanpa plugin pihak ketiga. Algoritma native umumnya lebih cepat dan lebih stabil. *Pendahuluan.*

NDVI (Normalized Difference Vegetation Index): indeks kehijauan vegetasi, dihitung dari band inframerah dekat (NIR) dan merah: $(NIR - Red) / (NIR + Red)$. *Bab 1.*

NoData: sel raster yang tidak memiliki nilai. Pada pencarian pucuk pohon, sel yang bukan puncak menjadi NoData. *Bab 3.*

OBIA (Object-Based Image Analysis): klasifikasi yang memecah citra menjadi objek poligon lebih dulu, bukan menggolongkan piksel satu per satu. Hasilnya tidak "bercak-bercak". *Bab 5.*

Ortofoto (orthophoto): foto udara dari drone yang sudah dikoreksi sehingga dapat dipakai sebagai peta. Di Bab 3 dipakai dalam bentuk ubin. *Bab 3.*

OTB (Orfeo ToolBox): perangkat pengolah citra yang dipasang terpisah dari QGIS. Dipakai untuk segmentasi dan Random Forest di Bab 5. *Sebelum Mulai, Bab 5.*

OTF Projection (on-the-fly projection): fitur QGIS yang menyelaraskan tampilan layer ke CRS proyek secara otomatis, tanpa membuat berkas baru. *Bab 1.*

PDAL: pustaka pengolah point cloud yang dipakai alat point cloud QGIS. Alat itu memerlukan PDAL versi 2.5.0 atau lebih baru. *Sebelum Mulai, Bab 4.*

Plugin: tambahan yang dipasang ke QGIS lewat Plugins > Manage and Install Plugins, misalnya STAC API Browser dan QFieldSync. *Sebelum Mulai, Bab 1.*

Point cloud (awan titik): kumpulan titik berkoordinat 3D hasil LiDAR atau fotogrametri, biasanya berformat .las atau .laz. *Bab 4.*

Precondition (prasyarat): syarat berupa nilai Boolean yang harus True agar sebuah alat di ModelBuilder berjalan. *Bab 6.*

Proyeksi ulang (reprojection): mengubah data dari satu CRS ke CRS lain, misalnya dari derajat ke UTM. Di ArcGIS memakai Project atau Project Raster. *Bab 1.*

PyQGIS: antarmuka Python untuk QGIS. Python Console bawaan QGIS dipakai untuk mengotomatiskan tugas berulang. *Pendahuluan.*

QField: aplikasi lapangan untuk ponsel yang membuka proyek QGIS. Tim lapangan dapat mengisi data tanpa sinyal, lalu menyinkronkannya saat ada internet. *Bab 2.*

QFieldCloud: layanan cloud untuk menyinkronkan proyek QField. Ia punya paket gratis dengan batas penyimpanan dan paket berbayar. *Sebelum Mulai, Bab 2.*

QFieldSync: plugin QGIS yang mengemas proyek menjadi paket untuk QField lewat Package for QField. *Bab 2.*

QgsTask: kelas PyQGIS untuk menjalankan pemrosesan di latar belakang, sehingga QGIS tidak hang saat memproses banyak ubin. *Bab 3.*

Random Forest (hutan acak): algoritma machine learning yang memakai banyak pohon keputusan. Di ArcGIS padanannya Train Random Trees Classifier. *Bab 5.*

Raster Calculator (kalkulator raster): alat untuk menghitung nilai raster dengan ekspresi, misalnya NDVI atau CHM (DSM dikurangi DTM). *Bab 1, 3.*

Resampling (pengambilan sampel ulang): metode menentukan nilai piksel baru saat raster diproyeksikan ulang. Untuk data kategorikal pilih Nearest atau Majority; untuk data kontinu pilih Bilinear atau Cubic. *Bab 1.*

Salt and pepper noise (noise bercak-bercak): hasil klasifikasi piksel tradisional yang tampak berbintik. OBIA mengurangi masalah ini. *Bab 5.*

Segmentasi (segmentation): memecah citra menjadi objek poligon berdasarkan kemiripan warna dan tekstur, misalnya dengan Large-Scale Mean-Shift di OTB atau Segment Mean Shift di ArcGIS. *Bab 5.*

Shapefile: format data vektor lama yang digantikan GeoPackage sebagai wadah data lapangan. *Bab 2.*

SMRF (Simple Morphological Filter): algoritma klasifikasi titik tanah yang dipakai Classify ground points. Dirancang untuk LiDAR udara, dan parameternya perlu disetel sesuai medan. *Bab 4.*

Spatial Join (gabung spasial): menarik atribut poligon ke titik yang berada di dalamnya. Dipakai untuk memvalidasi titik lapangan dan, di ArcGIS, menghitung titik panas per KPH. *Bab 2, 6.*

STAC (Spatiotemporal Asset Catalog): standar modern pencarian data spasial. Dengan STAC Anda mencari dan memanggil citra dari katalog di cloud tanpa mengunduhnya. *Pendahuluan, Bab 1.*

Tegakan (stand): kumpulan pohon di suatu area. Jumlah tegakan dihitung dari pucuk pohon pada CHM. *Bab 3.*

Teks dinamis (dynamic text): teks pada layout yang isinya berubah mengikuti fitur pada halaman aktif, misalnya nama dan luas KPH. *Bab 6.*

Triangulasi (TIN): metode membuat permukaan dari titik-titik dengan menghubungkannya menjadi segitiga. Dipakai untuk mengekspor titik tanah menjadi DTM. *Bab 4.*

Ubin (tile): potongan kecil dari citra besar. Citra drone sering terbagi menjadi banyak ubin yang diproses satu per satu. *Bab 3.*

UTM (Universal Transverse Mercator): sistem proyeksi berbasis meter yang dibagi menjadi zona. Nomor zona dihitung dari bujur, dengan belahan Utara (N) atau Selatan (S). *Bab 1.*

Value Map: jenis widget formulir QGIS yang membatasi isian pada pilihan baku, misalnya Sehat, Terserang Hama, atau Tebangan Liar. *Bab 2.*

/vsicurl/ (GDAL VSI): awalan pada alamat citra agar GDAL membaca berkas langsung dari server tanpa mengunduhnya utuh. GDAL hanya menarik piksel yang diperlukan. *Bab 1.*

XYZ Tiles: layanan peta dasar berupa ubin gambar yang dimuat lewat panel Browser QGIS tanpa plugin tambahan. *Bab 1.*

Zonal Statistics: alat yang menghitung statistik raster, seperti rata-rata dan deviasi standar, untuk tiap poligon. Hasilnya menjadi fitur objek bagi machine learning. *Bab 5.*

Daftar Pustaka

Daftar ini memuat sumber yang dipakai menyusun buku ini. Semua tautan dibuka dan dicek pada 7 Oktober 2026. Halaman dokumentasi daring tidak mencantumkan tahun terbit yang jelas, sehingga ditulis (t.t.).

Dokumentasi resmi

5. Cloud Optimized GeoTIFF. (t.t.). *Cloud Optimized GeoTIFF*. cogeo.org. <https://cogeo.org/>.
6. Esri. (t.t.). *Classify LAS Ground (3D Analyst)*. ArcMap, Documentation (desktop.arcgis.com). <https://desktop.arcgis.com/en/arcmap/latest/tools/3d-analyst-toolbox/classify-las-ground.htm>.
7. Esri. (t.t.). *Classify LAS Ground (3D Analyst Tools)*. ArcGIS Pro documentation (doc.esri.com). <https://doc.esri.com/en/arcgis-pro/latest/tool-reference/3d-analyst/classify-las-ground.html>.
8. Esri. (t.t.). *Extract LAS (3D Analyst Tools)*. ArcGIS Pro documentation (doc.esri.com). <https://doc.esri.com/en/arcgis-pro/latest/tool-reference/3d-analyst/extract-las.html>.
9. Esri. (t.t.). *Focal Statistics (Spatial Analyst Tools)*. ArcGIS Pro documentation (doc.esri.com). <https://doc.esri.com/en/arcgis-pro/latest/tool-reference/spatial-analyst/focal-statistics.html>.
10. Esri. (t.t.). *Get Count (Data Management Tools)*. ArcGIS Pro documentation (doc.esri.com). <https://doc.esri.com/en/arcgis-pro/latest/tool-reference/data-management/get-count.html>.
11. Esri. (t.t.). *Introduction to the SpatioTemporal Asset Catalog (STAC)*. ArcGIS Pro documentation (doc.esri.com). <https://doc.esri.com/en/arcgis-pro/latest/help/data/imagery/introduction-to-stac.html>.
12. Esri. (t.t.). *LAS Dataset To Raster (Conversion Tools)*. ArcGIS Pro documentation (doc.esri.com). <https://doc.esri.com/en/arcgis-pro/latest/tool-reference/conversion/las-dataset-to-raster.html>.
13. Esri. (t.t.). *Lidar point classification*. ArcMap, Documentation (desktop.arcgis.com). <https://desktop.arcgis.com/en/arcmap/latest/manage-data/las-dataset/lidar-point-classification.htm>.
14. Esri. (t.t.). *Project Raster (Data Management)*. ArcMap, Documentation (desktop.arcgis.com). <https://desktop.arcgis.com/en/arcmap/latest/tools/data-management-toolbox/project-raster.htm>.
15. Esri. (t.t.). *Project Raster (Data Management Tools)*. ArcGIS Pro documentation (doc.esri.com). <https://doc.esri.com/en/arcgis-pro/latest/tool-reference/data-management/project-raster.html>.
16. Esri. (t.t.). *Segment Mean Shift (Image Analyst Tools)*. ArcGIS Pro documentation (doc.esri.com). <https://doc.esri.com/en/arcgis-pro/latest/tool-reference/image-analyst/segment-mean-shift.html>.
17. Esri. (t.t.). *Segment Mean Shift (Spatial Analyst)*. ArcMap, Documentation (desktop.arcgis.com). <https://desktop.arcgis.com/en/arcmap/latest/tools/spatial-analyst-toolbox/segment-mean-shift.htm>.
18. Esri. (t.t.). *Train Random Trees Classifier (Image Analyst Tools)*. ArcGIS Pro documentation (doc.esri.com). <https://doc.esri.com/en/arcgis-pro/latest/tool-reference/image-analyst/train-random-trees-classifier.html>.
19. Esri. (t.t.). *Train Random Trees Classifier (Spatial Analyst)*. ArcMap, Documentation (desktop.arcgis.com). <https://desktop.arcgis.com/en/arcmap/latest/tools/spatial-analyst-toolbox/train-random-trees-classifier.htm>.

20. Esri. (t.t.). *What are Data Driven Pages?* ArcMap, Documentation (desktop.arcgis.com). <https://desktop.arcgis.com/en/arcmap/latest/map/page-layouts/what-are-data-driven-pages-.htm>.
21. Kartoza. (t.t.). *STAC API Browser*. QGIS Python Plugins Repository. https://plugins.qgis.org/plugins/qgis_stac/.
22. Orfeo ToolBox. (t.t.). *LargeScaleMeanShift*. Orfeo ToolBox 10.0.0 documentation. https://www.orfeo-toolbox.org/CookBook/Applications/app_LargeScaleMeanShift.html.
23. Orfeo ToolBox. (t.t.). *Orfeo ToolBox is not a black box*. orfeo-toolbox.org. <https://www.orfeo-toolbox.org/>.
24. Orfeo ToolBox. (t.t.). *TrainVectorClassifier*. Orfeo ToolBox 10.0.0 documentation. https://www.orfeo-toolbox.org/CookBook/Applications/app_TrainVectorClassifier.html.
25. Orfeo ToolBox. (t.t.). *VectorClassifier*. Orfeo ToolBox 10.0.0 documentation. https://www.orfeo-toolbox.org/CookBook/Applications/app_VectorClassifier.html.
26. PDAL. (t.t.). *filters.smrf*. Point Data Abstraction Library (PDAL) documentation. <https://pdal.io/en/latest/stages/filters.smrf.html>.
27. QField. (t.t.). *QField Ecosystem Documentation*. docs.qfield.org. <https://docs.qfield.org/>.
28. QGIS Project. (t.t.). *Creating an Output* (bagian 18.3). QGIS Documentation 4.2. https://docs.qgis.org/4.2/en/docs/user_manual/print_layout/create_output.html.
29. QGIS Project. (t.t.). *Modeler tools* (bagian 24.1.12). QGIS Documentation 4.2. https://docs.qgis.org/4.2/en/docs/user_manual/processing_algs/qgis/modelertools.html.
30. QGIS Project. (t.t.). *Opening Data* (bagian 11.1). QGIS Documentation 4.2. https://docs.qgis.org/4.2/en/docs/user_manual/managing_data_source/opening_data.html.
31. QGIS Project. (t.t.). *Point Cloud Conversion* (bagian 24.1.15). QGIS Documentation 4.2. https://docs.qgis.org/4.2/en/docs/user_manual/processing_algs/qgis/pointcloudconversion.html.
32. QGIS Project. (t.t.). *Point Cloud Data Management* (bagian 24.1.16). QGIS Documentation 4.2. https://docs.qgis.org/4.2/en/docs/user_manual/processing_algs/qgis/pointclouddatamanagement.html.
33. QGIS Project. (t.t.). *Point Cloud Extraction* (bagian 24.1.17). QGIS Documentation 4.2. https://docs.qgis.org/4.2/en/docs/user_manual/processing_algs/qgis/pointcloudextraction.html.
34. QGIS Project. (t.t.). *Raster analysis* (bagian 24.1.18). QGIS Documentation 4.2. https://docs.qgis.org/4.2/en/docs/user_manual/processing_algs/qgis/rasteranalysis.html.
35. stac-utils. (t.t.). *qgis-stac-plugin: QGIS plugin for reading STAC APIs*. GitHub. <https://github.com/stac-utils/qgis-stac-plugin>.
36. STAC. (t.t.). *STAC: SpatioTemporal Asset Catalogs*. stacspec.org. <https://stacspec.org/en>.

Makalah dan buku

37. Pingel, T. J., Clarke, K. C., dan McBride, W. A. (2013). *An improved simple morphological filter for the terrain classification of airborne LIDAR data*. ISPRS Journal of Photogrammetry and Remote Sensing, 77, 21-30. DOI: 10.1016/j.isprsjprs.2012.12.002. <https://doi.org/10.1016/j.isprsjprs.2012.12.002>.

Data dan layanan

38. EOX IT Services GmbH. (t.t.). *EOXCloudless*. eox.at. <https://cloudless.eox.at/>.

- 39.** EOX IT Services GmbH. (t.t.). *License Non-Commercial*. EOxCloudless.
<https://cloudless.eox.at/license-non-commercial>.

Aplikasi Live Penulis

Berikut aplikasi web milik penulis yang bisa Anda buka langsung. Daftar ini dicek pada 7 Oktober 2026. Sumbernya bagian Aplikasi Live di <https://badarmubarok.id/#aplikasi-live>.

No.	Aplikasi	Tautan
1	SIM RHL — Sistem Informasi Manajemen Rehabilitasi Hutan dan Lahan (Fakultas Kehutanan UGM), Versi PC	https://simrhl.badarmubarok.id
2	SIM RHL, Versi HP	https://m.simrhl.badarmubarok.id
3	ScienceThink — RAD Tool (Rapid Application Development)	https://sciencethink.badarmubarok.id
4	Peta Landsystem Indonesia — peta landsystem dan peta tematik lainnya	https://landsystem.badarmubarok.id
5	Analisis Kebutuhan Unsur Hara — aplikasi Web GIS analisis unsur hara	https://unsurhara.badarmubarok.id
6	Analisa Karbon — Web GIS estimasi cadangan karbon dari citra satelit	https://karbon.badarmubarok.id
7	Aplikasi Deforestasi — deteksi perubahan tutupan hutan (2001–2025)	https://deforestasi.badarmubarok.id
8	Deteksi Area Terbakar — pemetaan tingkat keparahan area terbakar (NBR/dNBR)	https://kebakaran.badarmubarok.id
9	Mata Elang — platform deteksi konservasi berbasis AI	https://mataelang.badarmubarok.id
10	SIM Konflik — Sistem Informasi Manajemen Konflik	https://simkonflik.badarmubarok.id
11	Seri Buku GIS Kehutanan dan Pertanian — 12 buku praktik GIS (dalam bahasa Indonesia)	https://badarmubarok.id/buku-gis/

MASTER CLASS · SERI M1

MASTERCLASS GIS MODERN

Alur kerja modern untuk analis kehutanan dan pertanian: citra cloud, survei lapangan, drone, LiDAR, klasifikasi, dan atlas otomatis..

Yang akan Anda pelajari

- 1 Fondasi Modern & Manajemen Data Cerdas
- 2 Sinkronisasi Lapangan (Ground-Truthing Cloud)
- 3 Pertanian Presisi & Analisis Citra Drone (UAV)
- 4 Kehutanan Lanjut: Analisis Point Cloud (LiDAR) & DTM
- 5 Klasifikasi Berbasis Objek (OBIA) & Machine Learning
- 6 Otomatisasi Alur Kerja & Atlas Dinamis



**Pindai untuk membuka
seluruh seri buku GIS**

Buku PDF, paket data, dan skrip
<https://badarmubarok.id/buku-gis/>