

BAB V

PENGUJIAN DAN PEMBAHASAN

2.1 Tujuan Pengujian

Pengujian dilakukan untuk memastikan bahwa Add-on PTracker yang dikembangkan dapat berjalan sesuai dengan tujuan penelitian. Secara khusus, pengujian ini bertujuan untuk:

1. Memastikan *Penginstall-an Add-on Photogrammetry Importer* berjalan lancar
2. Memastikan Add-on PTracker dapat terpasang dan dijalankan pada Blender versi 5.0.1 tanpa error.
3. Menguji proses setup *Workspace* secara otomatis sesuai struktur *folder* yang telah dirancang.
4. Menguji proses *Camera Tracking* berbasis *Photogrammetry* menggunakan COLMAP, GLOMAP, dan FFMPEG.
5. Menilai keberhasilan proses *Camera Tracking* berdasarkan *output* yang dihasilkan.
6. Menganalisis kemudahan penggunaan Add-on dibandingkan *workflow* manual.
7. Membandingkan akurasi hasil *camera tracking* dengan metode manual bawaan blender.

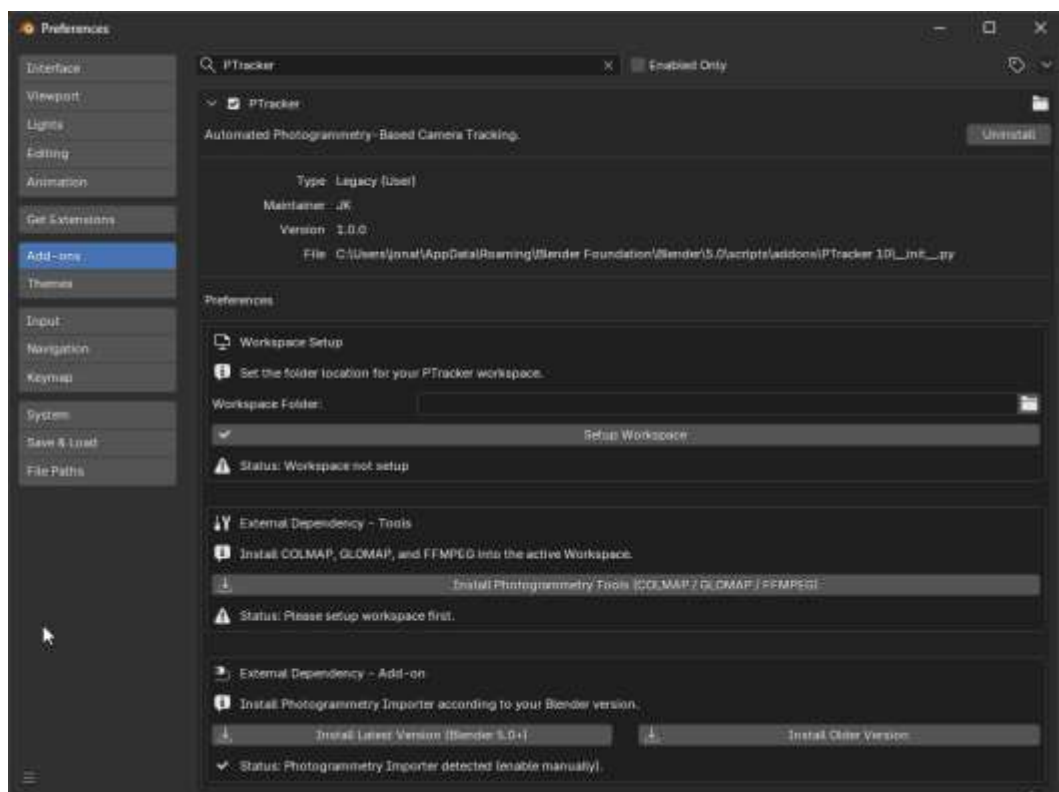
2.2 Lingkungan Pengujian

Pengujian sistem dilakukan dengan spesifikasi lingkungan sebagai berikut:

- Sistem Operasi: Windows 11
- Blender: Versi 5.0.1
- Bahasa Pemrograman: Python (Blender API)
- *Tools* Pendukung: FFMPEG (v8.0.1), COLMAP (v3.12.3), GLOMAP (v1.1.0), *Photogrammetry Importer* (v3.2.1)
- Jenis Data Uji: *Video real-world* (Video test 1.mp4)

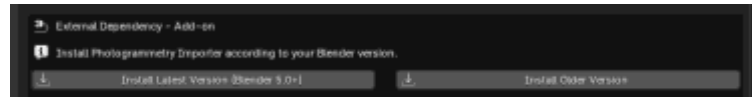
Lingkungan pengujian ini dipilih karena sesuai dengan target pengguna add on, yaitu pengguna Blender pada sistem operasi Windows.

2.3 Pengujian Instalasi Add-on

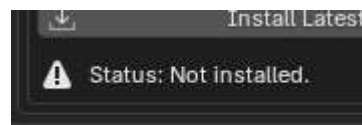


Gambar 5.1 Tampilan Preferences Add-on PTracker

Pada gambar 5.2 menampilkan tampilan preferences yang dimana skenario pengguna menginstall Add-on PTracker melalui menu *Preferences > Add-on* di Blender dengan kondisi awal belum ada Add-on *Photogrammetry Importer* terinstall seperti yang ditunjukkan pada gambar 5.2 dan 5.3

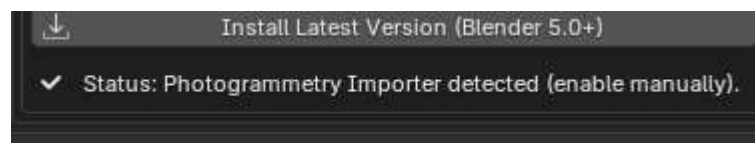


Gambar 5.2 Tampilan Install Add-on *Photogrammetry Importer*



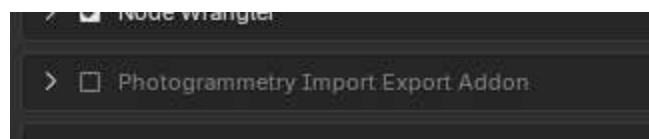
Gambar 5.3 Status Add-on *Photogrammetry Importer*

Hasil yang Diharapkan adalah Add-on PTracker berhasil terinstal dan dengan mengklik tombol *Install* untuk menginstall Add-on eksternal (sesuai dengan versi Blender yang digunakan) langsung otomatis terinstall, pengguna hanya perlu melakukan *enable*.



Gambar 5.4 Status Installed Add-on *Photogrammetry Importer*

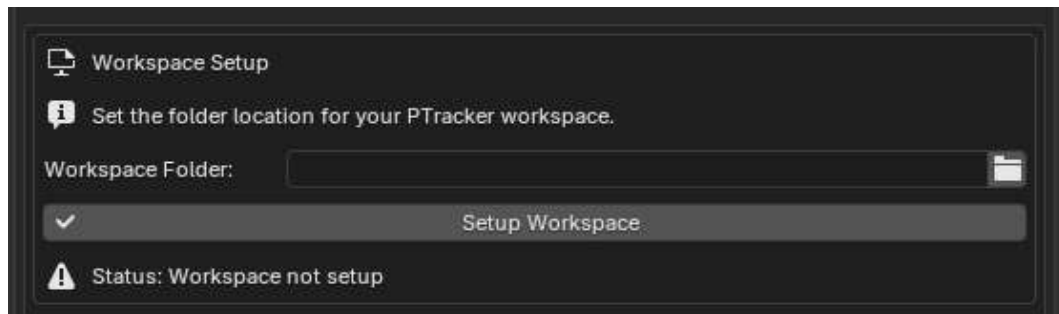
Hasil pengujiannya adalah Add-on eksternal berhasil terinstal dan muncul di library Preferences Blender tanpa error ditunjukkan pada gambar 5.4.



Gambar 5.5 Add-on *Photogrammetry Importer* terinstall di library

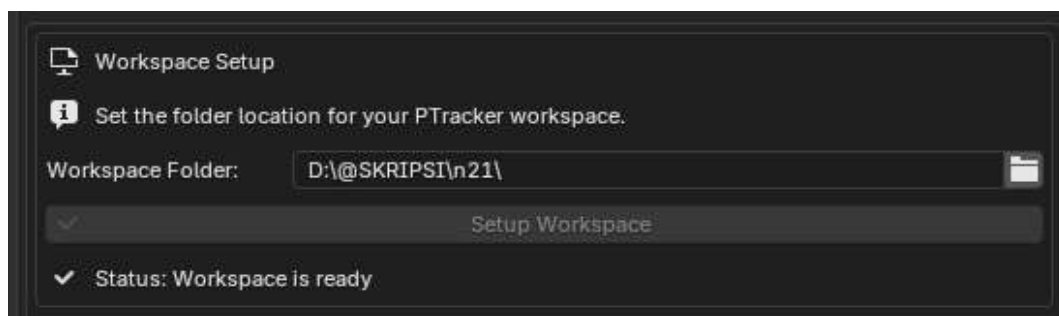
Setelah *terinstall* dan terdeteksi seperti pada gambar 5.5 berjalan dengan baik dihalaman N-Panel pada 3D Viewport. Kesimpulan, pengujian berhasil.

2.4 Pengujian Setup *Workspace*

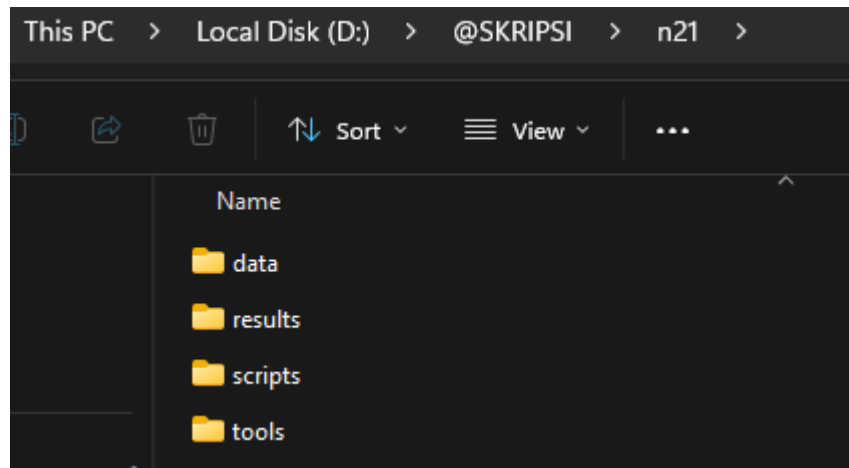


Gambar 5.6 Tampilan Panel *Workspace* Setup

Seperti pada gambar 5.6 skenario pengguna menentukan lokasi *path Workspace* dan menekan tombol *Setup Workspace*. Hasil yang Diharapkan adalah sistem secara otomatis membuat struktur *folder Workspace* dan menyalin *File AutoTracker.bat* ke *folder scripts*. Hasil Pengujian adalah Struktur *folder* berhasil dibuat sesuai rancangan dan Add-on mendeteksi folder *workspace* yang dibuat, mengubah status menjadi "*Workspace is ready*" seperti pada gambar 5.7.

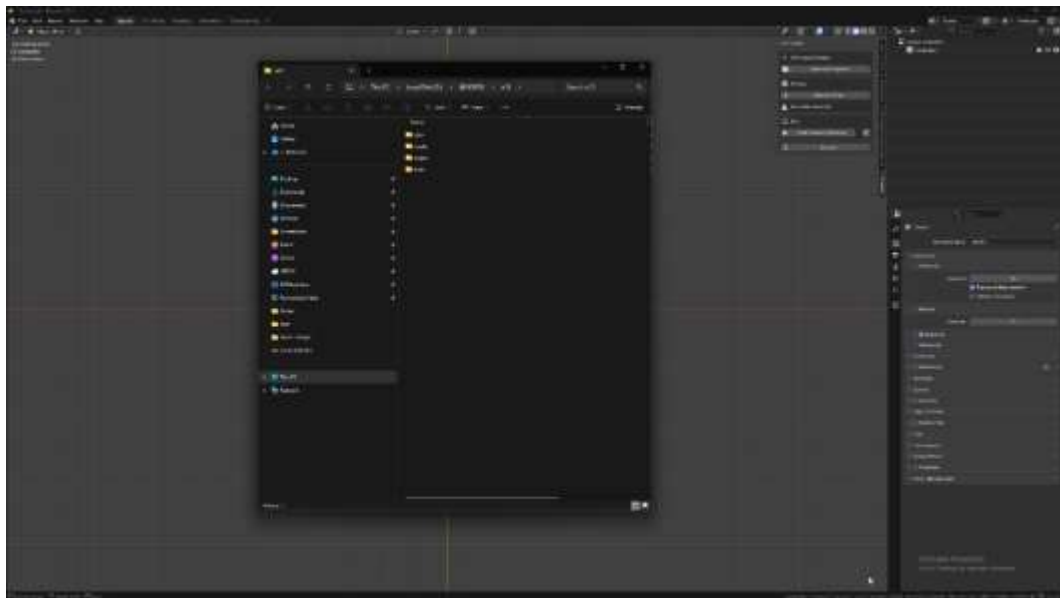


Gambar 5.7 *Workspace* telah disetup Sesuai Struktur



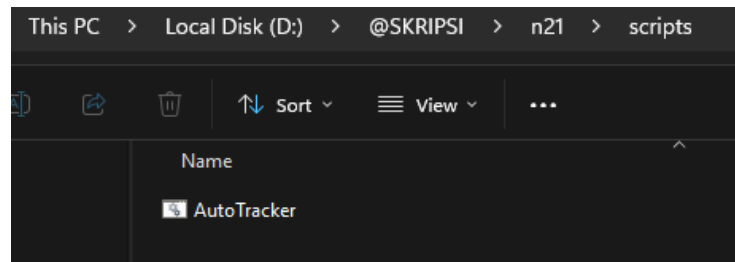
Gambar 5.8 *Folder* Telah dibuat Sesuai Struktur *Workspace*

Window File Explorer terbuka dengan Lokasi yang sama dengan *Workspace* ditunjukkan pada gambar 5.8.



Gambar 5.9 *Folder Open Workspace*

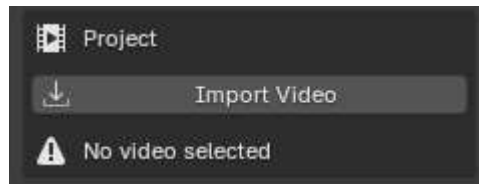
File AutoTracker.bat berhasil disalin ditunjukkan pada gambar 5.9.



Gambar 5.10 AutoTracker Berhasil Dipindahkan Kedalam *Workspace*

File AutoTracker.bat berhasil dipindahkan ke *scripts* ditunjukkan pada gambar 5.10. Kesimpulan, pengujian berhasil.

2.5 Pengujian *Import Video*



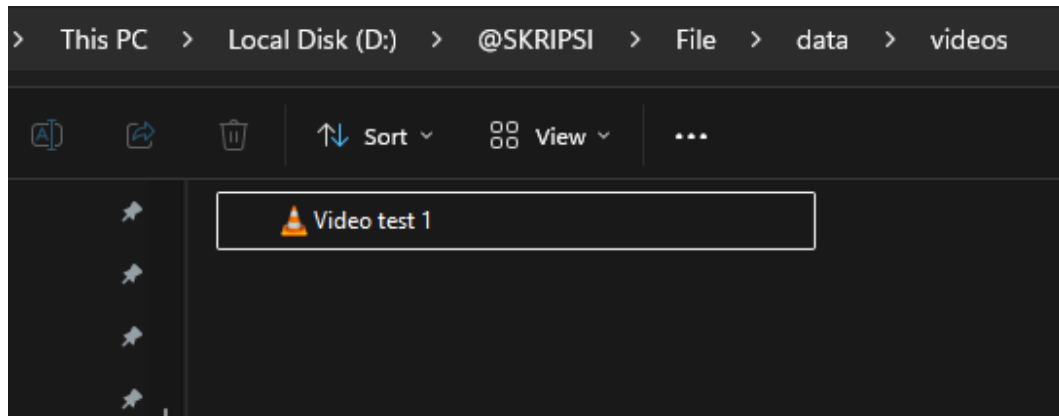
Gambar 5.11 Panel *Import Video*

Tampilan *Import Video* ditunjukkan pada gambar 5.11, *File video* yang digunakan adalah *video* yang berisi merekam mengitari satu titik, *video* direkam pada jam 10.00 WIB lokasi outdoor di *rooftop* seperti pada gambar 5.12.



Gambar 5.12 Lokasi Pengambilan *Video*

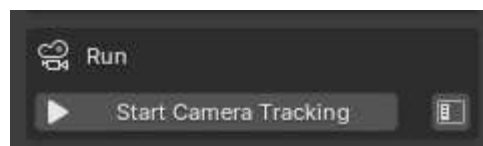
Pada panel Add-on dengan skenario pengguna menekan tombol *Import Video* dan memilih *File video* dengan format yang didukung. Hasil yang diharapkan *Video* berhasil disalin ke *folder data/videos* dan nama *video* tampil pada panel Add-on.



Gambar 5.13 *File Video Di Dalam Workspace*

Hasil Pengujian *Video* berhasil diimport dan informasi *video* tampil pada UI ditunjukan pada gambar 5.13. Kesimpulan, pengujian berhasil.

2.6 Pengujian Proses *Camera Tracking*



Gambar 5.14 *Panel Camera Tracking*

Tampilan *Start Camera Tracking* ditunjukan pada gambar 5.14 dengan Skenario, Pengguna menekan tombol *Start Camera Tracking* setelah *Workspace* dan *video* siap. Hasil yang diharapkan adalah sistem menjalankan *pipeline Photogrammetry* secara otomatis melalui *AutoTracker.bat* dan menghasilkan *output Camera Tracking*.

```

120260128 10:14:02.428271 13976 feature_extraction.cc:262] Name: frame_000180.jpg
120260128 10:14:02.428578 13976 feature_extraction.cc:271] Dimensions: 1920 x 1080
120260128 10:14:02.428622 13976 feature_extraction.cc:274] Camera: #1 - SIMPLE_RADIAL
120260128 10:14:02.428669 13976 feature_extraction.cc:277] Focal Length: 2304.00px
120260128 10:14:02.428717 13976 feature_extraction.cc:281] Features: 11686
120260128 10:14:02.460167 13976 feature_extraction.cc:259] Processed file [181/465]
120260128 10:14:02.460261 13976 feature_extraction.cc:262] Name: frame_000181.jpg
120260128 10:14:02.460318 13976 feature_extraction.cc:271] Dimensions: 1920 x 1080
120260128 10:14:02.460340 13976 feature_extraction.cc:274] Camera: #1 - SIMPLE_RADIAL
120260128 10:14:02.460360 13976 feature_extraction.cc:277] Focal Length: 2304.00px
120260128 10:14:02.460390 13976 feature_extraction.cc:281] Features: 11686
120260128 10:14:02.492267 13976 feature_extraction.cc:259] Processed file [182/465]
120260128 10:14:02.492377 13976 feature_extraction.cc:262] Name: frame_000182.jpg
120260128 10:14:02.492383 13976 feature_extraction.cc:271] Dimensions: 1920 x 1080
120260128 10:14:02.492362 13976 feature_extraction.cc:274] Camera: #1 - SIMPLE_RADIAL
120260128 10:14:02.492384 13976 feature_extraction.cc:277] Focal Length: 2304.00px
120260128 10:14:02.492908 13976 feature_extraction.cc:281] Features: 12002
120260128 10:14:02.524477 13976 feature_extraction.cc:259] Processed file [183/465]
120260128 10:14:02.524580 13976 feature_extraction.cc:262] Name: frame_000183.jpg
120260128 10:14:02.524634 13976 feature_extraction.cc:271] Dimensions: 1920 x 1080
120260128 10:14:02.524682 13976 feature_extraction.cc:274] Camera: #1 - SIMPLE_RADIAL
120260128 10:14:02.524710 13976 feature_extraction.cc:277] Focal Length: 2304.00px
120260128 10:14:02.524734 13976 feature_extraction.cc:281] Features: 11852
120260128 10:14:02.556713 13976 feature_extraction.cc:259] Processed file [184/465]
120260128 10:14:02.556809 13976 feature_extraction.cc:262] Name: frame_000184.jpg
120260128 10:14:02.557218 13976 feature_extraction.cc:271] Dimensions: 1920 x 1080
120260128 10:14:02.557254 13976 feature_extraction.cc:274] Camera: #1 - SIMPLE_RADIAL
120260128 10:14:02.557290 13976 feature_extraction.cc:277] Focal Length: 2304.00px
120260128 10:14:02.557312 13976 feature_extraction.cc:281] Features: 11980

```

Gambar 5.15 Window Toggle System Console

Berdasarkan gambar 5.15 menunjukkan bahwa system berjalan dan sedang memproses data dari *File video* yang diimport,

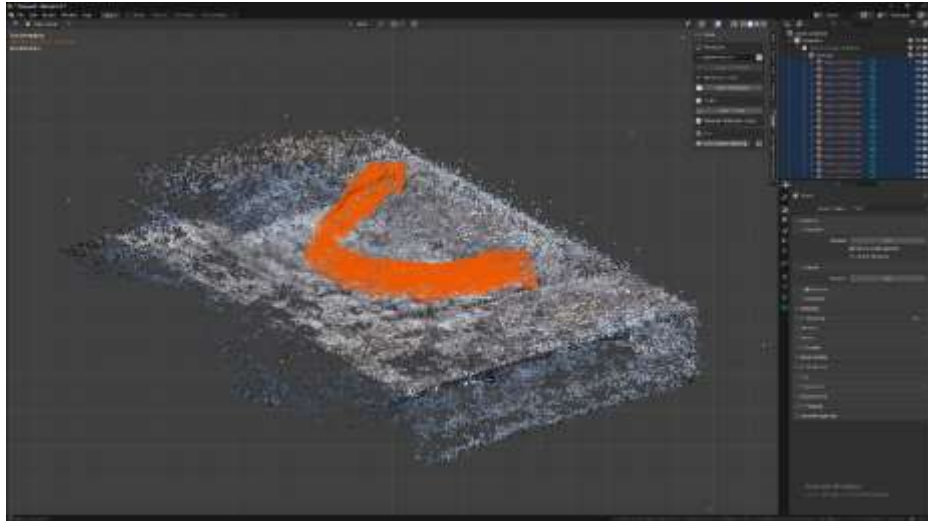
```

120260207 08:52:01.800934 24880 global_mapper.cc:92]
[PTRACKER_STAGE] EXPORT
[SUCCESS] Finished Video test 1
=====
All processing finished
=====

```

Gambar 5.16 Tampilan Jika Proses Camera Tracking Sudah Selesai

Hasil Pengujiannya adalah proses *Camera Tracking* berjalan dengan baik seperti pada gambar 5.16 ditandai dengan status pada *Toggle System Console* yang menunjukkan “*All processing finished*” memakan waktu sekitar 21 menit, menghasilkan *folder images*, *sparse*, serta *file* database hasil rekonstruksi.



Gambar 5.17 Hasil *Camera Tracking*

Setelah selesai dan diimport kedalam Blender. Dapat dilihat object yang berwarna *orange* adalah jejak posisi *camera* per 1 *frame*, yang dimana *camera* berputar 180 derajat mengelilingi satu titik seperti pada gambar 5.17. Kesimpulan, pengujian berhasil.

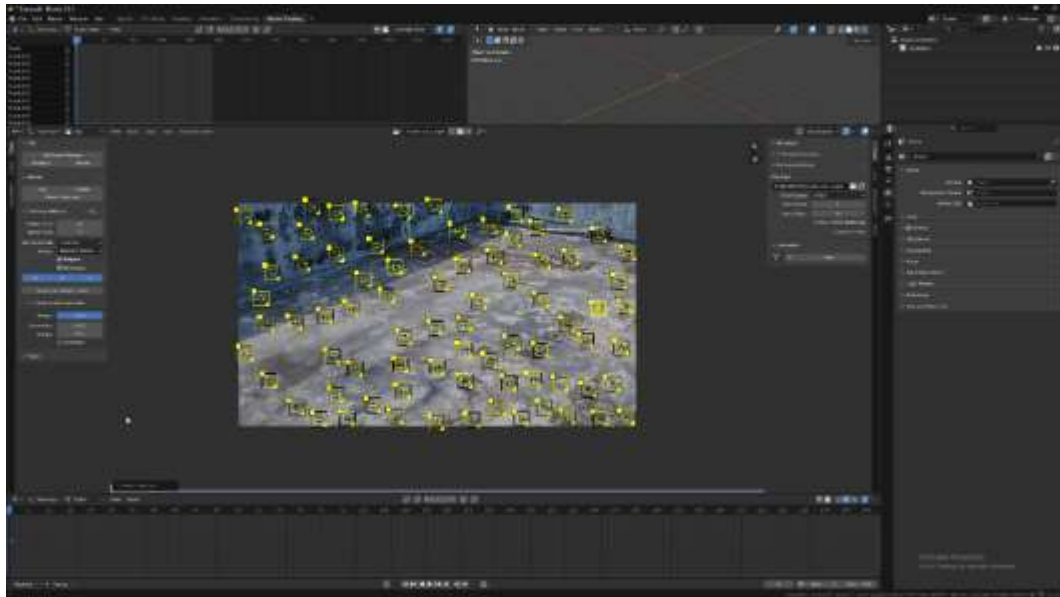
2.7 Perbandingan Akurasi *Camera Tracking*

Di pengujian ini akan membandingkan hasil akhir akurasi dengan bahan *video* yang sama (*Video test 1.mp4*) antara menggunakan add-on PTracker yang telah dibuat dengan metode *photogrammetry* dengan metode tradisional atau manual dengan fitur bawaan dari Blender.

5.7.1 *Camera Tracking Manual* (Fitur Blender)

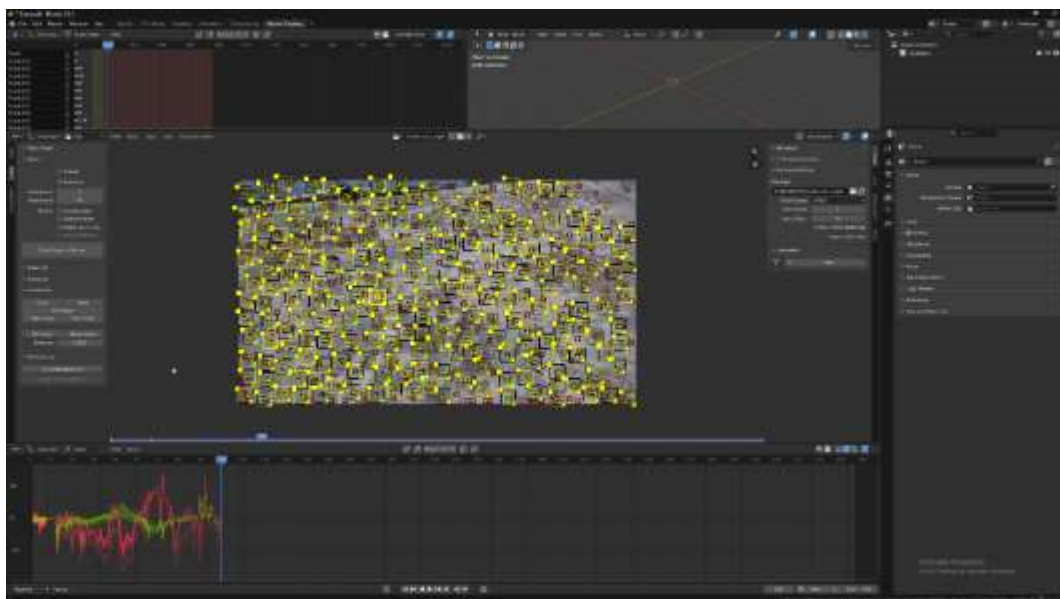
Untuk mengakses fitur *camera tracking* blender kita harus masuk ke *Workspace* VFX > Motion Tracking. *Import video*, lakukan Prefetch agar preview lebih lancar lalu Detect *Features* untuk melakukan deteksi otomatis point-point yang diperkirakan dapat menjadi acuan system tracking oleh blender. Dan disarankan untuk melakukan adjustment manual pada point tracker agar hasil lebih

akurat, pada proses inilah yang biasanya memakan waktu, proses *tracking* manual ditunjukkan pada gambar 5.18.



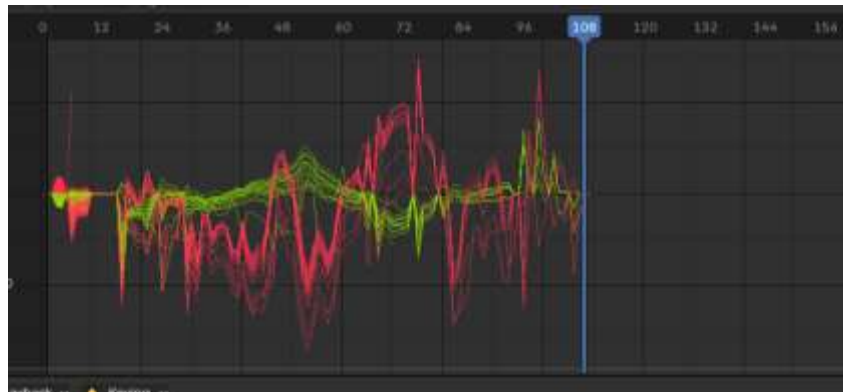
Gambar 5.18 Tampilan Fitur *Tracking* Manual Blender

Tracking dilakukan dengan cara CTRL + T untuk *forward tracking*, hasil dari tracking dapat dilihat pada gambar 5.19.



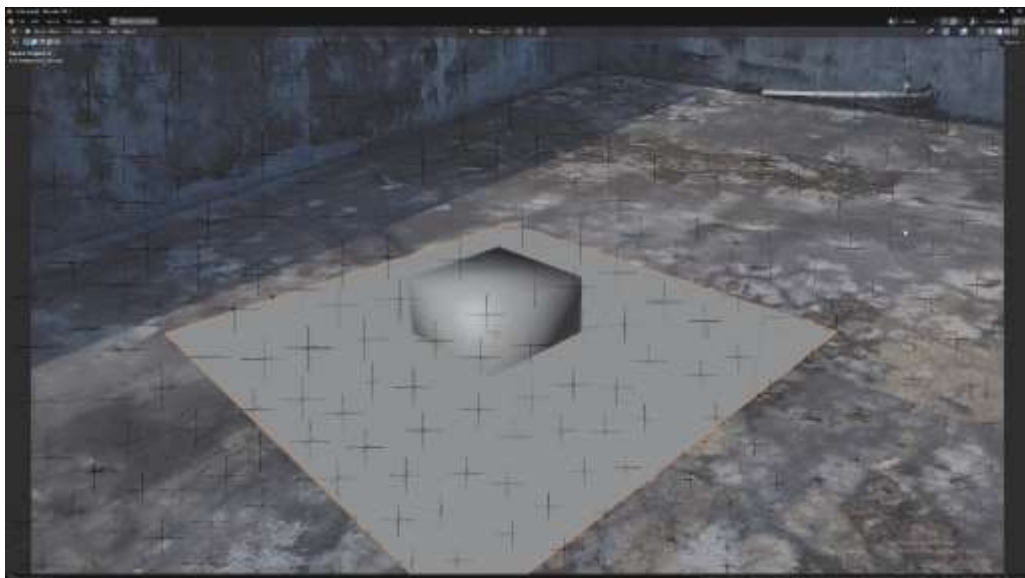
Gambar 5.19 Hasil *Tracking* Manual Blender

Tracking berhenti pada *frame* 108 dari total 465 *frames* pada gambar 5.20.



Gambar 5.20 Data *Keyframe Point Tracker*

progress hanya berjalan sekitar 23,23% dikarenakan kondisi *camera* merekam mengelilingi sekitar sehingga titik awal point tracker keluar dari area *frame video* yang menyebabkan point tracker tidak memiliki data dan juga *video* tidak memiliki *contrast* dan warna yang bervariasi untuk tiap point tracker menjadi acuan. Inilah salah satu kelemahan manual tracking. Hasil akhir dari manual tracking dapat dilihat pada gambar 5.21 dan gambar 5.22.



Gambar 5.21 Timeline Manual Tracking *Frame* 12



Gambar 5.22 Timeline Manual *Tracking Frame* 61

Dengan menambahkan object 3D sederhana yaitu *Plane* dan *Cube* sebagai acuan akurasi *camera tracking*, dapat dilihat pada gambar 5.22 data *camera tracking* sudah tidak akurat yang menyebabkan object 3D terputar di disumbu X tidak mengikuti pergerakan kamera yang seharusnya.

5.7.2 *Camera Tracking Add-on (Automated Photogrammetry Based)*

Menggunakan hasil data *camera tracking* yang sudah dijalankan pada tahap pengujian, dapat dilihat pada gambar 5.23 dan gambar 5.24.



Gambar 5.23 Viewport *Photogrammetry Tracking Frame* 12



Gambar 5.24 Viewport Photogrammetry Tracking Frame 61

Hasil tracking *camera* tidak terputus, berjalan dengan lancar dari awal (*frame 0*) sampai akhir (*frame 465*) seperti pada gambar 5.25



Gambar 5.25 Timeline Keyframe Camera Tracking Add-on

2.8 Pembahasan Hasil Pengujian

Berdasarkan hasil pengujian yang telah dilakukan, Add-on PTracker terbukti mampu mengotomatisasi proses *Camera tracking* berbasis *photogrammetry* di dalam Blender. Proses yang sebelumnya memerlukan banyak langkah manual kini dapat dilakukan melalui satu panel terintegrasi. Penggunaan *File batch* (AutoTracker.bat) terbukti lebih stabil dalam menjalankan *pipeline Photogrammetry* dibandingkan eksekusi langsung melalui Python, terutama dalam menangani beban komputasi tinggi. Selain itu, struktur *Workspace* yang terorganisir memudahkan pengguna dalam mengelola

data input dan *output*. Dan akurasi dari camera tracking yang jauh lebih akurat disbanding dengan metode manual *tracking* yang tersedia didalam Blender.

2.9 Kelebihan dan Keterbatasan Sistem

Kelebihan dari sistem yang telah dibuat adalah berhasil mengotomatisasi *workflow Camera tracking* berbasis *photogrammetry*, mengurangi kesalahan pengguna akibat proses manual. Struktur *Workspace* yang rapi dan konsisten, stabil digunakan untuk pemrosesan data berskala besar. Sementara itu keterbatasannya adalah proses *import* hasil *tracking* ke Blender masih dilakukan secara manual begitu juga dengan data *File* COLMAP, GLOMAP, dan GLOMAP, pengujian masih terbatas pada sistem operasi Windows, belum tersedia progress bar visual secara real-time di dalam panel Add-on.