



GEODÉZIAI FELADATOK AUTOMATIZÁLÁSA KÉPFELDOLGOZÁSSAL

Siki Zoltán, Takács Bence



BUDAPESTI MŰSZAKI
ÉS GAZDASÁGTUDOMÁNYI EGYETEM
Építőmérnöki Kar - építőmérnöki képzés 1782 óta

Általános és Felsőgeodézia Tanszék

2021.05.07.

Rédey szeminárium

Vázlat

- Mozcásvizsgálatok, pontosság vizsgálat
- Illesztőpontok automatikus azonosítása a képeken
- A nyílt forráskód „ereje”



BUDAPESTI MŰSZAKI
ÉS GAZDASÁGTUDOMÁNYI EGYETEM

Építőmérnöki Kar - építőmérnöki képzés 1782 óta

Általános és Felsőgeodézia Tanszék

Mozgásvizsgálatok



Előzmények:

- Rákóczi híd dinamikus próbaterhelése
- Kecskeméti Máté diplomaterve 2019
- Rédey szeminárium előadás 2020
- Mérnökgeodézia Konferencia 2020
- Cikk a Baltic Journal of Modern Computing-ban 2021
- GK cikk megjelenés alatt

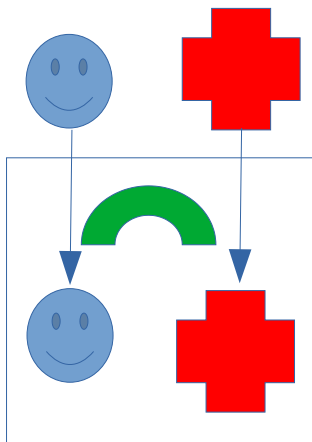
Mintakeresési módszerek



Mintaillesztés (template matching)

+

Mindig van találat
Egyszerű algoritmus
Nem kell speciális jelet feltenni



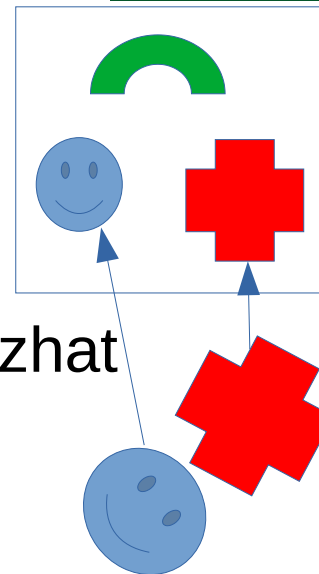
-

Hamis találat esélye nagyobb
Csak minimális elfordulás
Csak minimális méret változás

Alakfelismerés (pattern recognition)

+

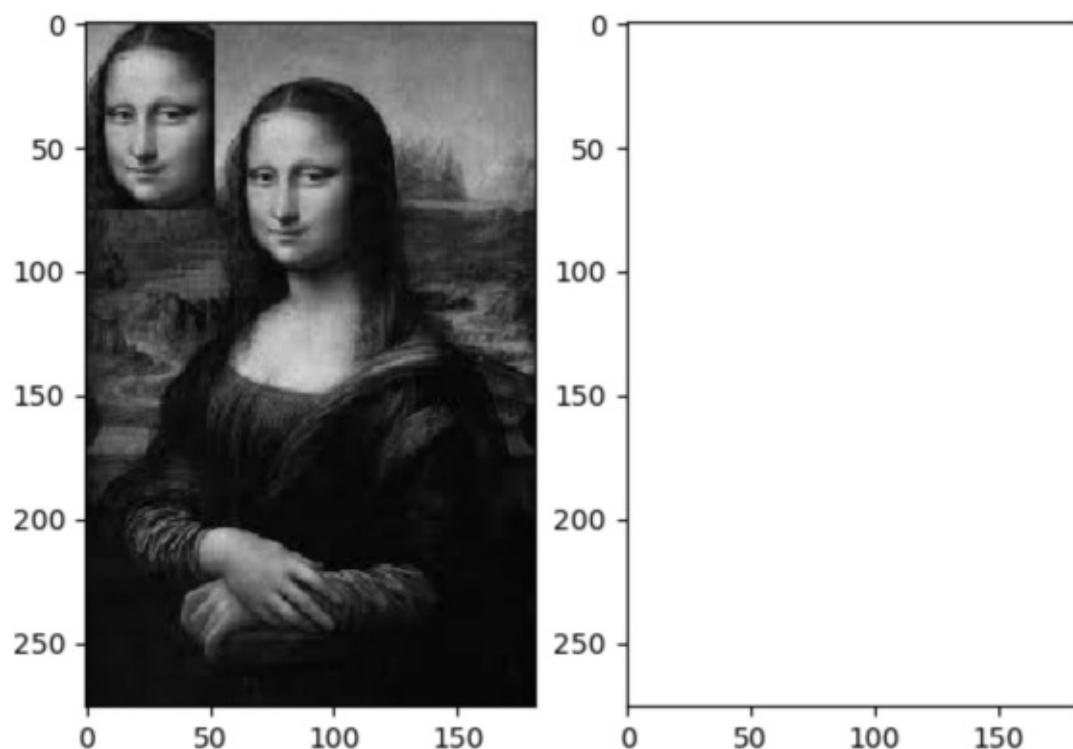
Minta elfordulhat
Minta mérete megváltozhat
Írány becsülhető (3D)



-

Fényviszonyokra érzékenyebb
Speciális jelet kell elhelyezni

Mintaillesztés (template matching)



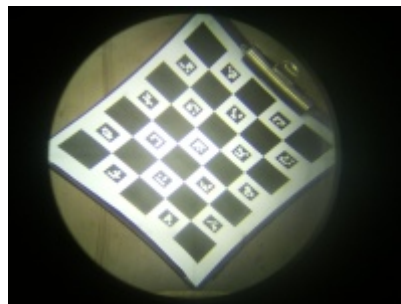
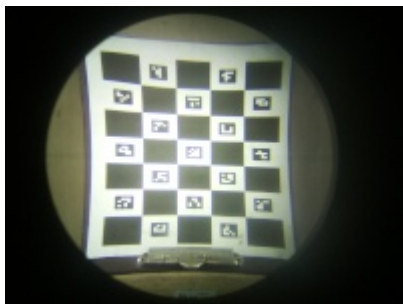
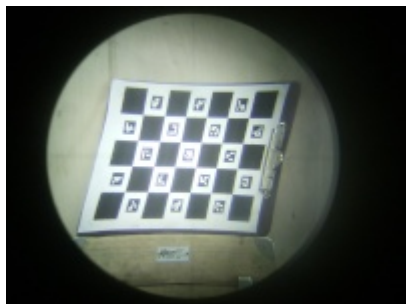
$$R(x, y) = \sum_{x', y'} (T(x', y') - I(x + x', y + y'))^2 \quad \min$$

$$R(x, y) = \frac{\sum_{x', y'} (T(x', y') - I(x + x', y + y'))^2}{\sqrt{\sum_{x', y'} T(x', y')^2 \cdot \sum_{x', y'} I(x + x', y + y')^2}} \quad \min$$

$$R(x, y) = \sum_{x', y'} (T(x', y') \cdot I(x + x', y + y')) \quad \max$$

$$R(x, y) = \frac{\sum_{x', y'} (T(x', y') \cdot I(x + x', y + y'))}{\sqrt{\sum_{x', y'} T(x', y')^2 \cdot \sum_{x', y'} I(x + x', y + y')^2}} \quad \max$$

Kamera kalibráció

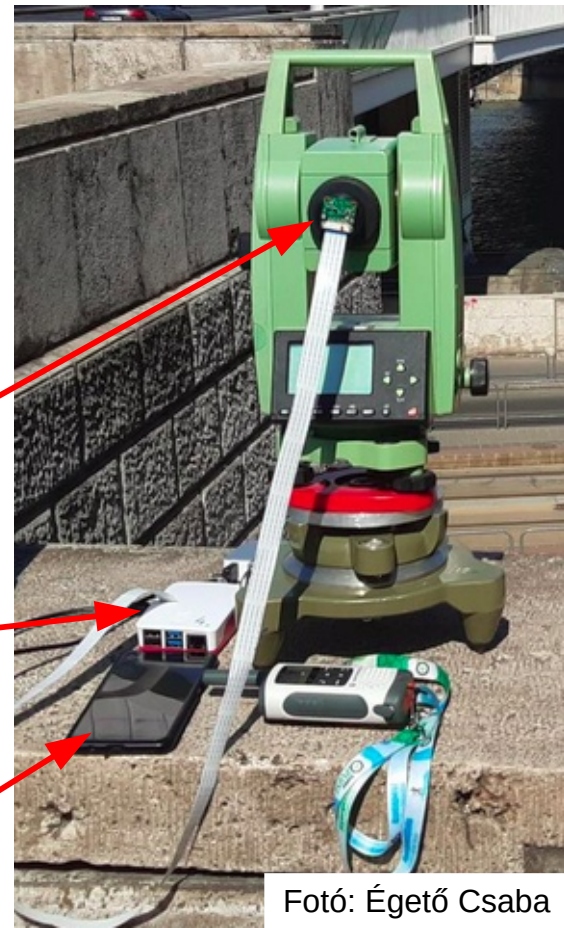


Távcsőre szerelt kamera képe erősen torzul.
Kamera kalibráció szükséges.

Pi kamera modul
8 MP

Raspberry Pi

Mobiltelefon

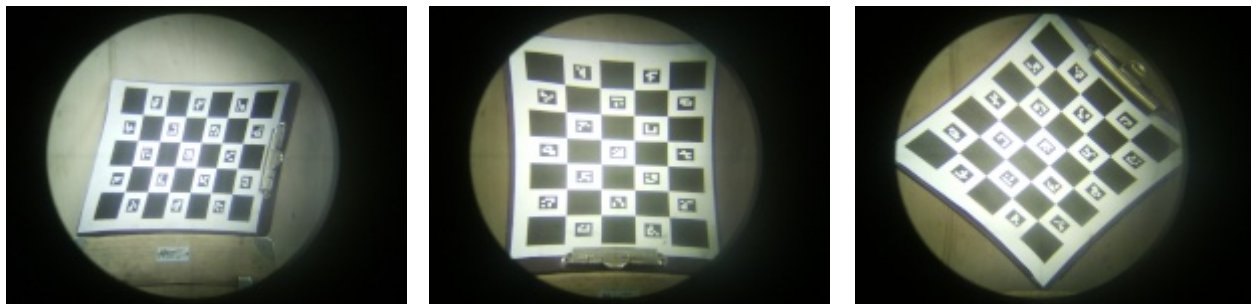


Fotó: Égető Csaba

Kamera kalibráció



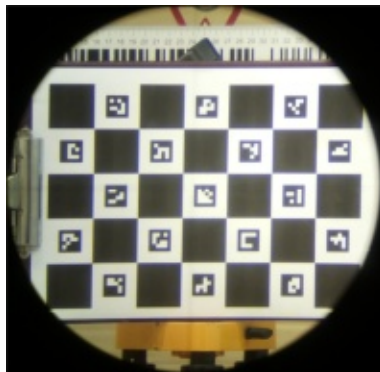
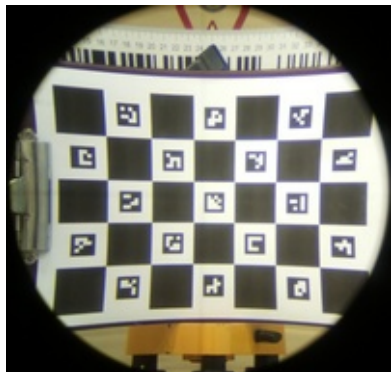
ChArUco tábla (25 felvétel)



$$\begin{bmatrix} 27785.301 & 0.0 & 1397.28 \\ 0.0 & 26618.89 & 1458.82 \\ 0.0 & 0.0 & 1 \end{bmatrix}$$

$$k = [75.00 \quad -605.21 \quad 5.79]$$

$$p = [0.61 \quad -0.64]$$



Radiális torzítás

$$x_{\text{torzított}} = x \cdot (1 + k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6)$$

$$y_{\text{torzított}} = y \cdot (1 + k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6)$$

Tangenciális torzítás

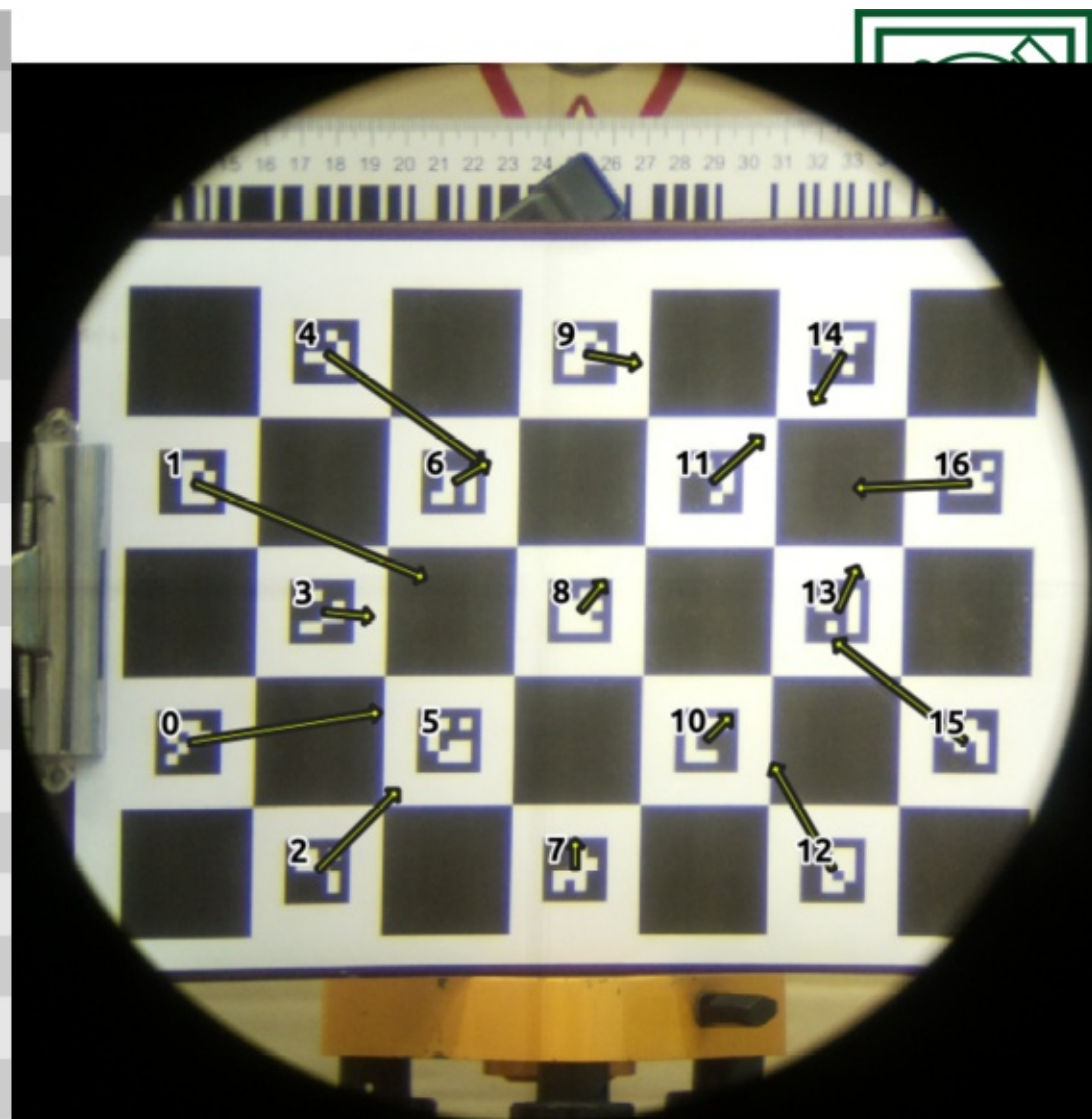
$$x_{\text{torzított}} = x + (2 \cdot p_1 \cdot x \cdot y + p_2 \cdot (r^2 + 2 \cdot x^2))$$

$$y_{\text{torzított}} = y + (p_1 \cdot (r^2 + 2 \cdot y^2) + 2 \cdot p_2 \cdot x \cdot y)$$

Kamera mátrix

$$\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

	előtt		után				
ID	X	Y	X	Y	DX	DY	Táv.
0	808	1466	846	1460	38	-6	38.5
1	808	928	854	947	46	19	49.8
2	1084	1728	1100	1712	16	-16	22.6
3	1100	1201	1110	1202	10	1	10.0
4	1086	669	1117	691	31	22	38.0
5	1360	1456	1360	1456	0	0	0.0
6	1362	950	1369	946	7	-4	8.1
7	1608	1715	1608	1709	0	-6	6.0
8	1613	1206	1618	1200	5	-6	7.8
9	1617	691	1628	693	11	2	11.2
10	1862	1460	1867	1455	5	-5	7.1
11	1868	955	1878	946	10	-9	13.5
12	2129	1732	2117	1711	-12	-21	24.2
13	2122	1210	2126	1201	4	-9	9.8
14	2143	682	2137	692	-6	10	11.7
15	2404	1478	2378	1458	-26	-20	32.8
16	2411	947	2388	948	-23	1	23.0



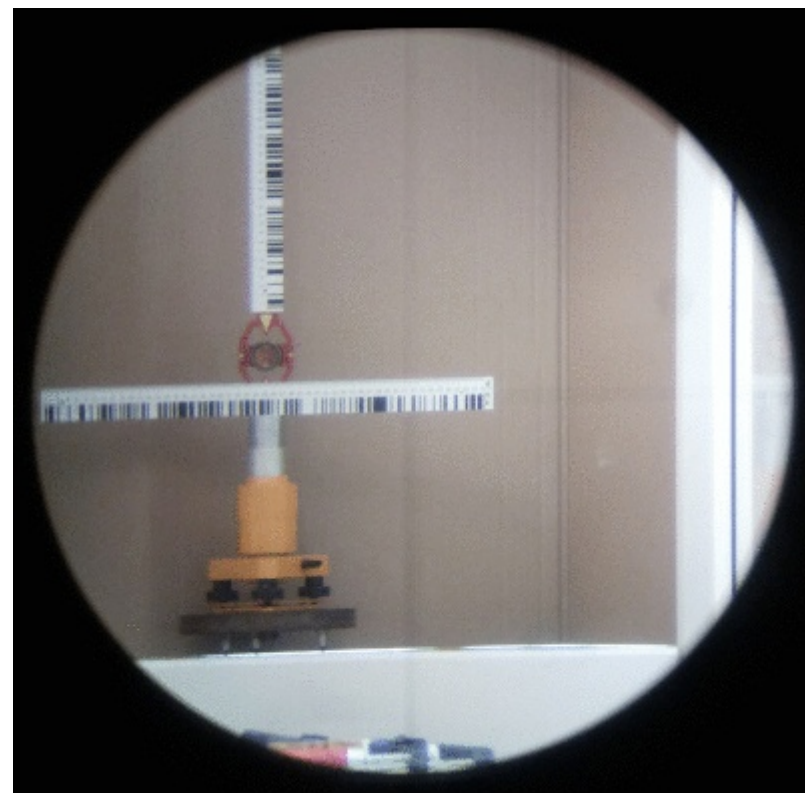
Eltolás vektorok ötszörös nagyítással

Pontosságvizsgálat



6-31 m távolságon 5 cm-enkénti leolvasások
55, 83 m vonalzó elején és végén leolvasás (51 cm)
kalibrációval javított képeken

táv [m]	Pixel / cm				minta	mm
	min	max	átlag	szórás	szám	szórás
6	121.4	124.2	122.7	0.7	27	0.06
11	66.4	69.6	67.8	0.6	78	0.09
20	38	39.7	38.5	0.3	75	0.08
22	33.6	35.5	34.1	0.3	<u>421</u>	0.09
31	23.8	25.8	24.7	0.2	<u>440</u>	0.08
55	13.7	14	13.9	0.1	15	0.07
83	9.2	9.5	9.3	0.1	18	0.11

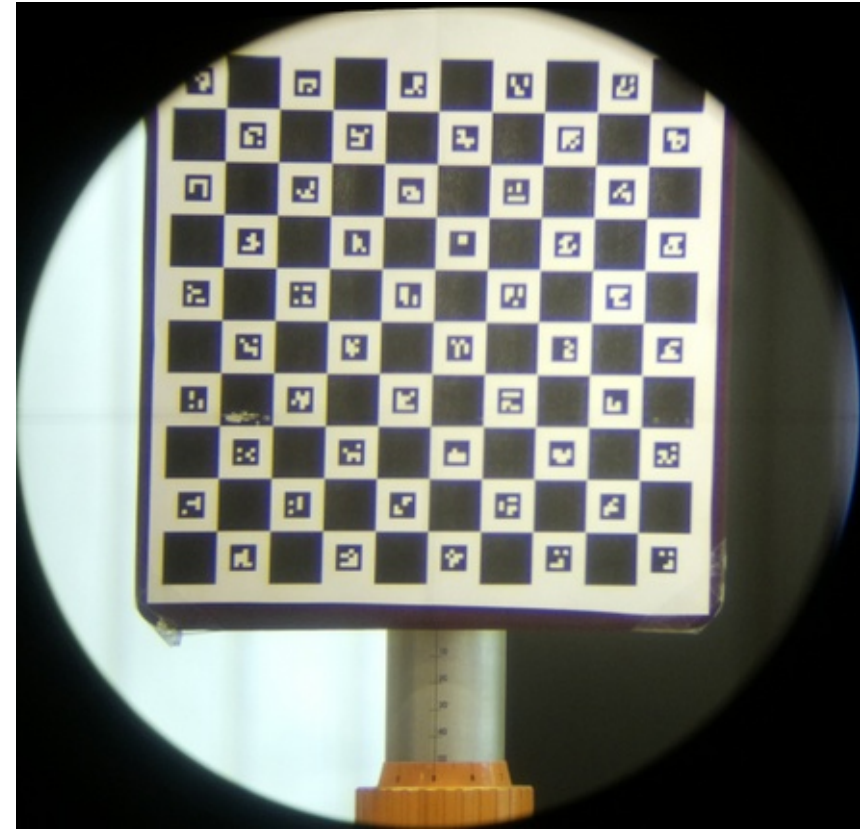


Pontosságvizsgálat



10 m – 50 pont / tábla

	Pixel / 0.1 mm				minta
Táv.	min	max	átlag	szórás	szám
10	6.0	8.1	6.8	0.1	1574



Pontosságvizsgálat



10 m – 50 pont / tábla

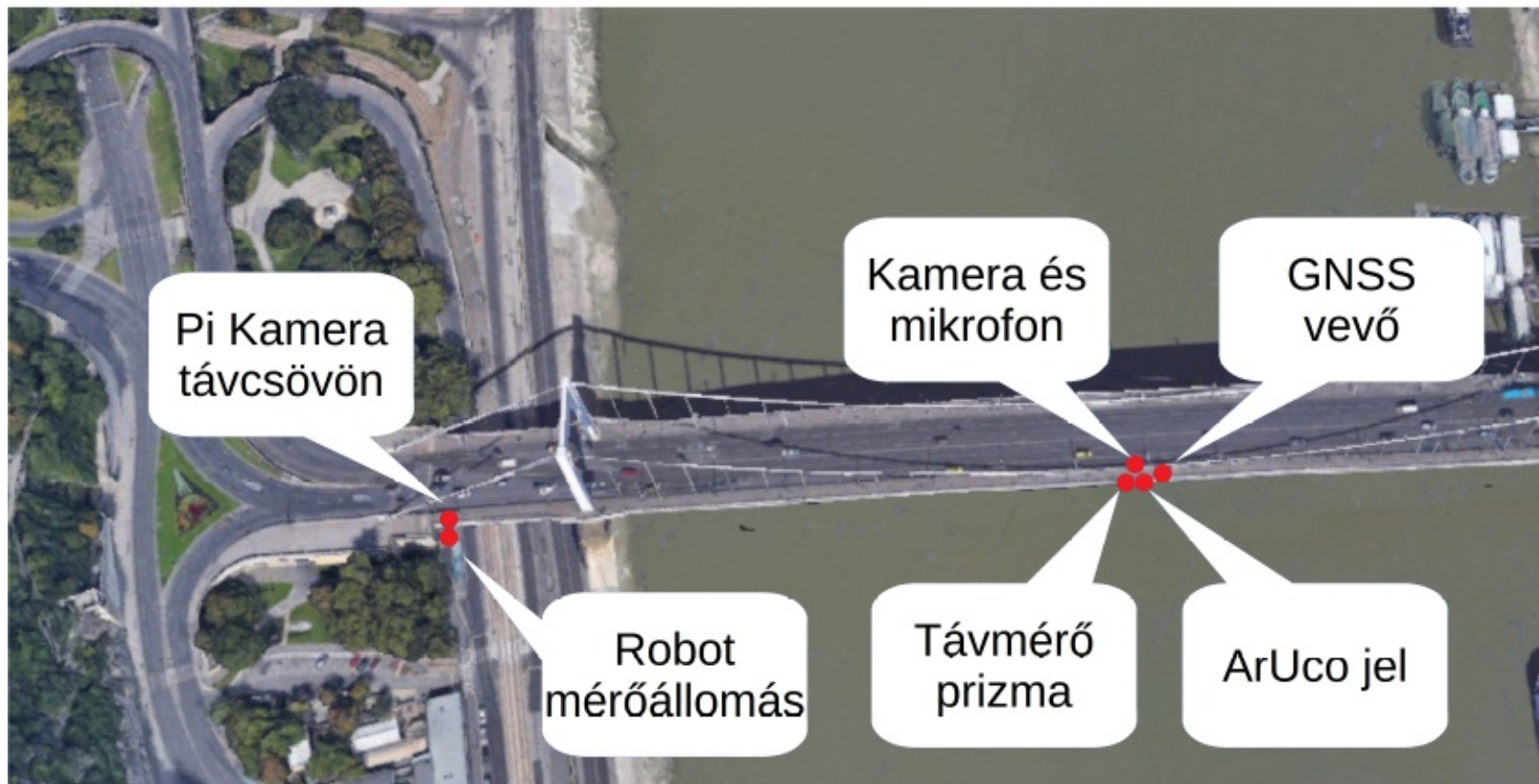
	Pixel / 0.1 mm				minta
Táv.	min	max	átlag	szórás	szám
10	6.0	8.1	6.8	0.1	1574

55 m – 17 pont / tábla

	Pixel / mm				minta
Táv.	min	max	átlag	szórás	szám
55	13	15	13.8	0.4	323



Alkalmazási példa

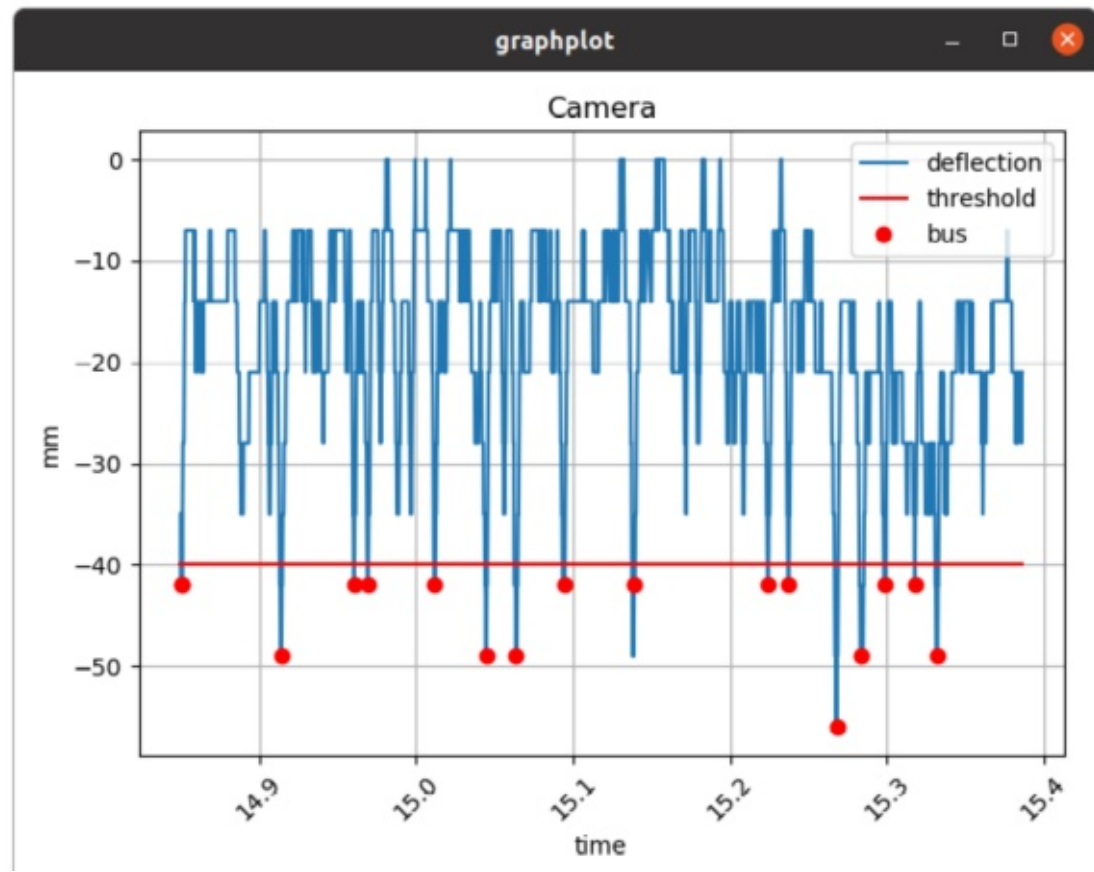
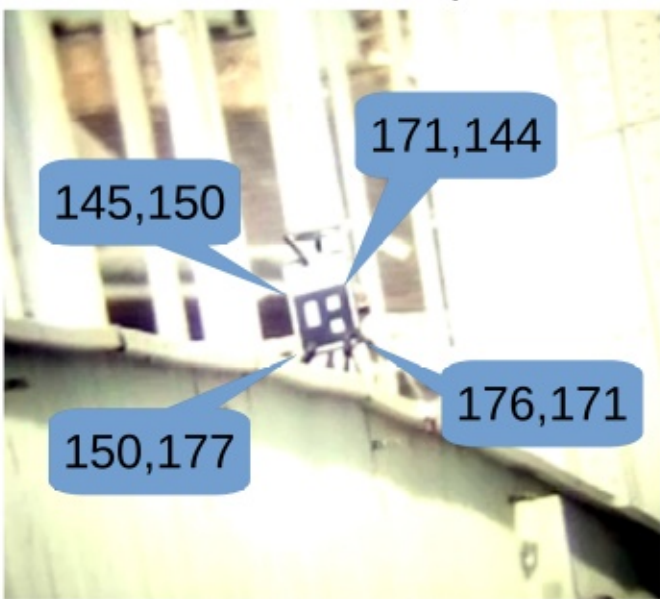


Alkalmazási példa



Kamera – jel távolság ~200 m
1 fps videó felvétel

$GSD = 190 \text{ mm} / 27 \text{ px} = 7 \text{ mm}$

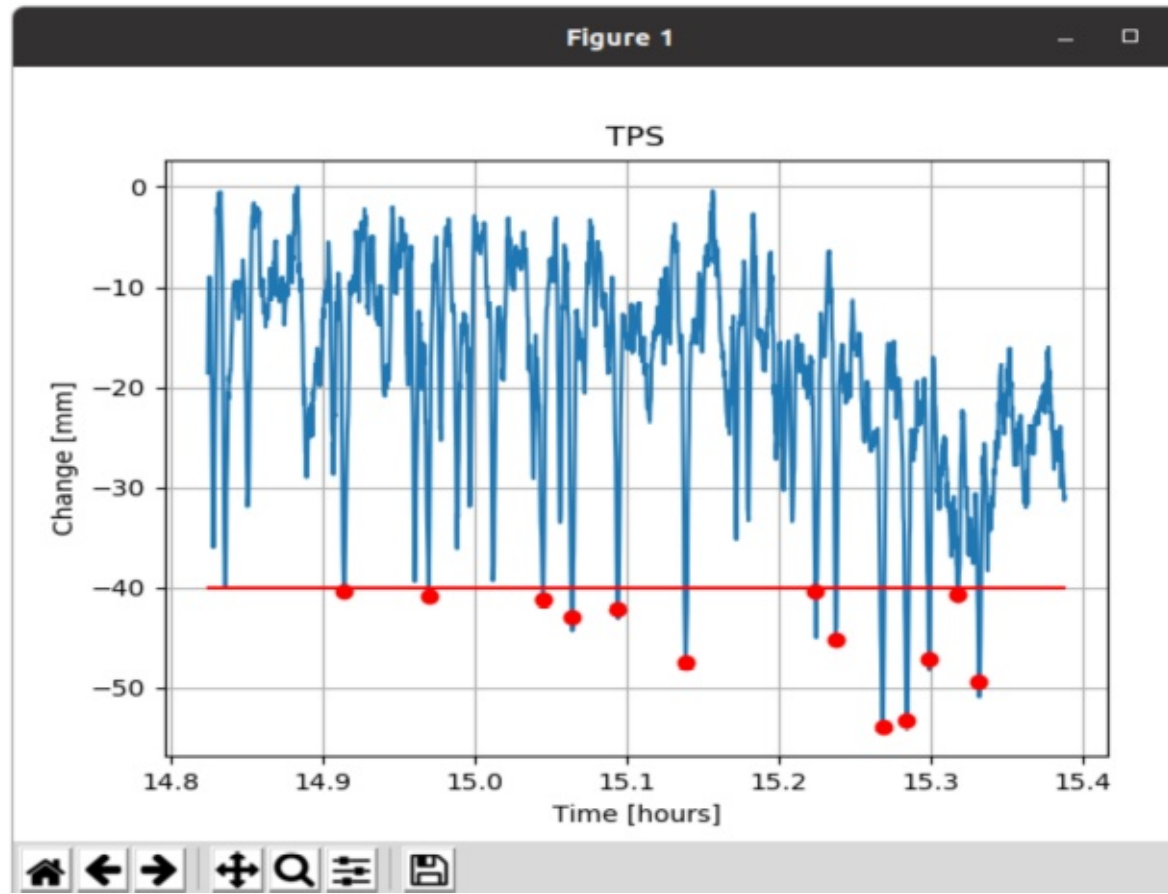
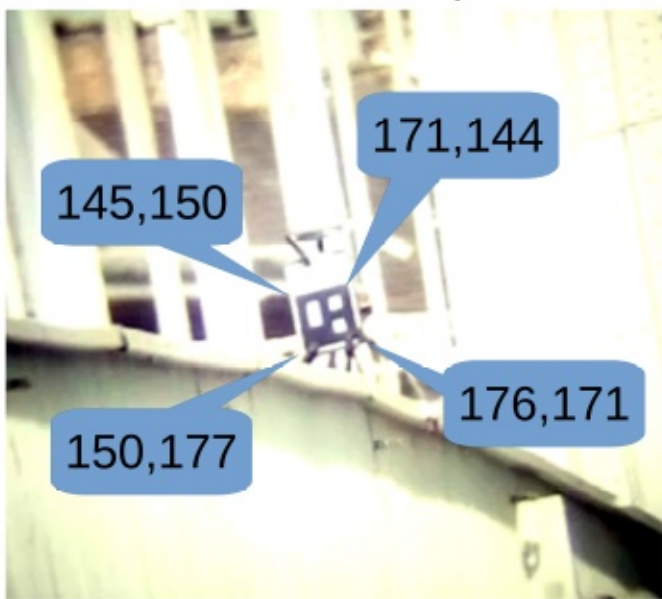


Alkalmazási példa



Kamera – jel távolság ~200 m
1 fps videó felvétel

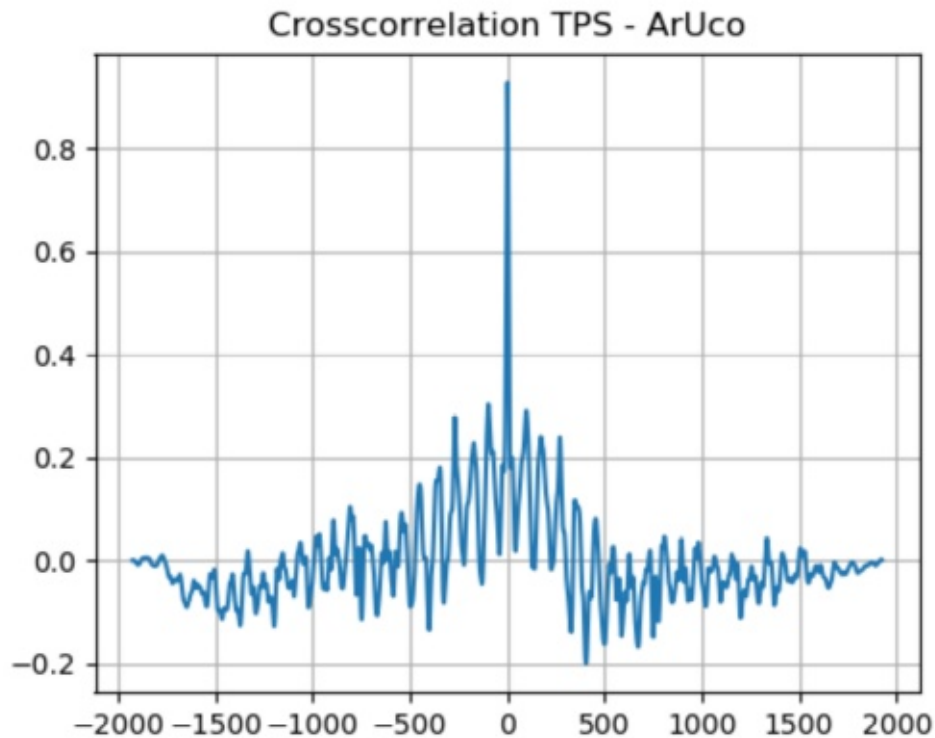
$GSD = 190 \text{ mm} / 27 \text{ px} = 7 \text{ mm}$



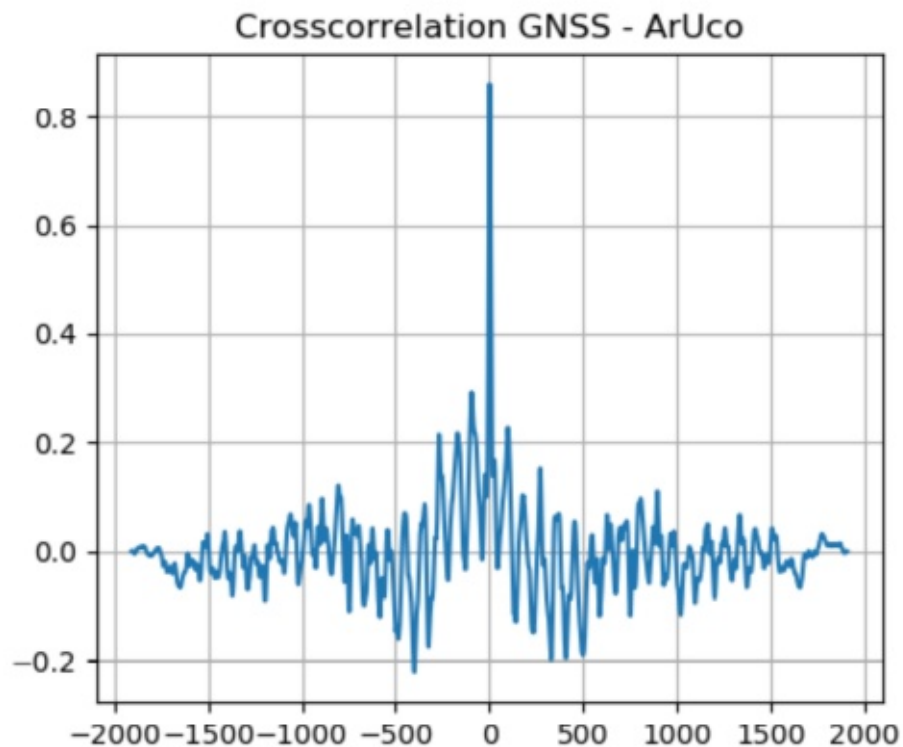
Alkalmazási példa

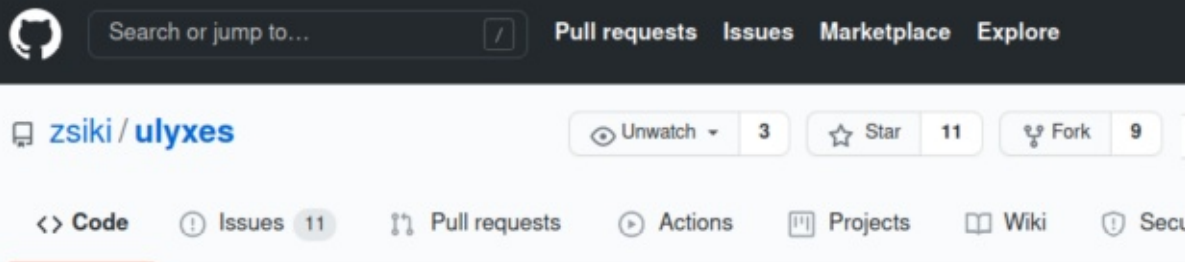


Eltérések átlaga: -0.5 mm Eltérés medián: -0.6 mm
Eltérések szórása: 3.8 mm Eltérés max: 12 mm



Eltérések átlaga: 17 mm Eltérés medián: 14 mm
Eltérések szórása: 9.7 mm Eltérés max: 54 mm





Forráskód



Ulyxes projekt része

Több kisebb segédprogram

Nem csak Pi kamerával
használható

7 résztvevő

Illesztőpontok automatikus azonosítása



Kétféle megoldás

Egyforma jól azonosítható jelek
relatív tájékozások után
legalább három illesztőpontot két képen azonosítani kell
3DSurvey



Kódolt jelek, eltérő mintázat minden illesztőpontra
feldolgozás elején megkereshetők az illesztőpontok
teljesen automatizálható
Photomodeler



Problémák és megoldások



Első kísérletek

2019 egyetem kert

Problémák

Felvételek az ArUco jelek „beégése”
Képterülethez képes kis méretű jelek



Megoldások

Szürke/fekete jelek (adaptív küszöbérték)
3 x 3 ArUco jelkészlet használata



Segédprogramok (Web)ODM-hez



gcp_find.py
exif_pos.py
gcp_check.py

dict_gen_3x3.py
aruco_make.py
gsd_calc

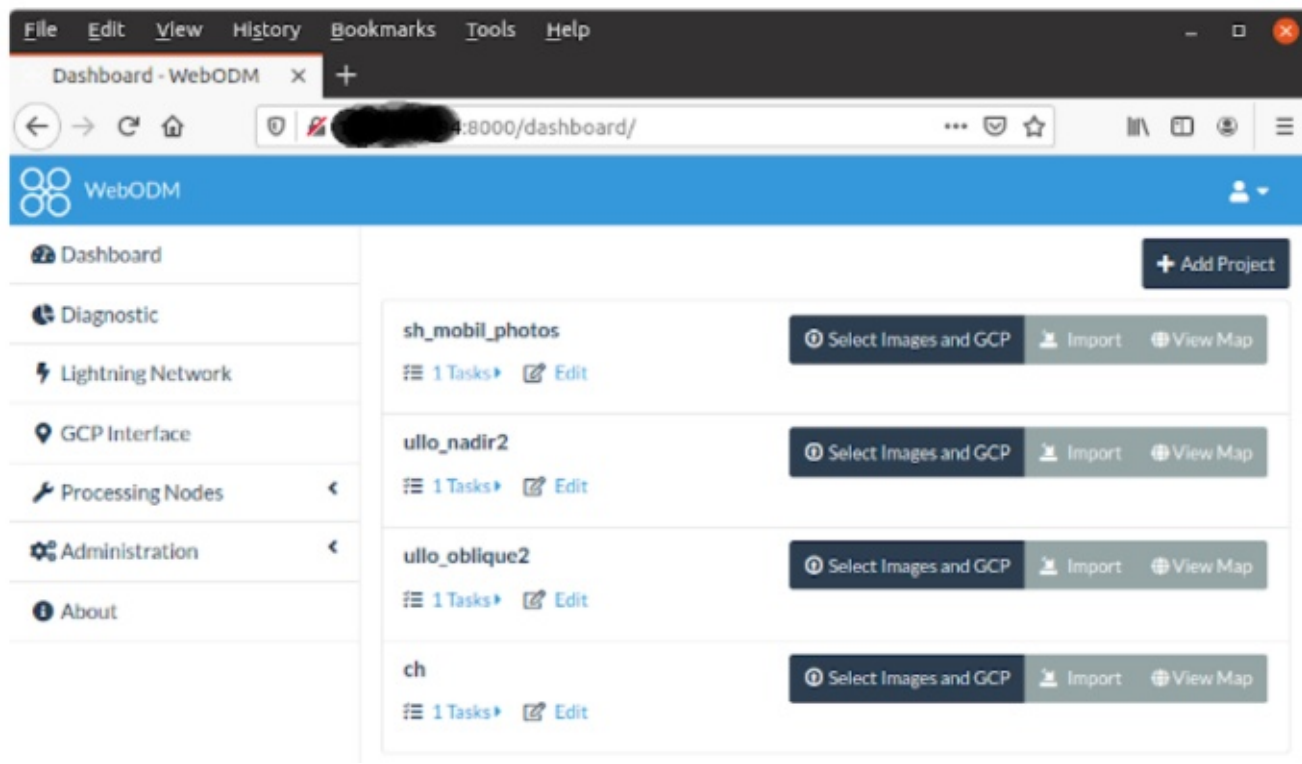


OpenDroneMap



ODM – Docker (Linux/Windows) parancssorból
WebODM – böngészőből

Saját szerverünkön is!



Eredmények



Repülési magasság	Jelméret	Képek száma	Felismert jelek száma	Nem felismert	Hamis felismerés
25 m	20 x 20 cm	8	22	7	0
30 m	30 x 30 cm	7	36	0	2
50 m	30 x 30 cm	4	31	0	0
65 m	30 x 30 cm	7	50	9	2

Szürke-fekete
jelek
90%

Repülési magasság	Jelméret	Képek száma	Felismert jelek száma	Nem felismert	Hamis felismerés
20 m	20 x 20 cm	3	5	1	0
50 m	20 x 20 cm	2	3	5	0
50 m	30 x 30 cm	2	7	1	0
75 m	30 x 30 cm	3	1	14	0

Fehér-fekete
jelek
43%

Alkalmazás éles feladatban



Repülési magasság ~50 m

Repülési mód	Jelméret (3 x 3)	Képek száma	Felismert jelek száma	Hamis felismerés
nadír	30 x 30 cm	225	29	
ferde (25°)	30 x 30 cm	499	58	
ferde (25°)	30 x 30 cm	805	199	299 (271)

Automatizáltan
kizárható pont ID
alapján



ArUco méret számítás



On-line
szolgáltatás

← → ↺ 🏠 🔒 www.agt.bme.hu/on_line/gsd_calc/gsd_calc.html ... 🛡️ ☆ 📄 📱 🌐 ☰

GSD and ArUco marker size calculator

Sensor type

Phantom_4_Pro

Sensor data

Sensor width Focal length Image width Image height Flight height ArUco size

13.2 8.8 5472 3648 50 3 x 3

Calculate

Reset

Results

GSD Width Height ArUco width
cm/pixel m m cm

1.4 75.0 50.0 25

Usage:

Fill the fields in the "GSD and ArUco marker size calculator" group and click on the Calculate button. All input fields are mandatory. Selecting a sensor type some input fields are filled from our small database.

Search or jump to... Pull requests Issues Marketplace Explore

zsiki / Find-GCP Unwatch

Code Issues 1 Pull requests Actions Projects Wiki Security

master 1 branch 0 tags Go to file Add file Code

zsiki	ABC order corrected	f428c42	10 days ago	74 commits
gsd_calc	ABC order corrected		10 days ago	
samples	coordinates for GCPs added		2 months ago	
LICENSE	Initial commit		2 years ago	
README.md	format of sensordb.json documented		20 days ago	
aruco_make.py	documentation updated		8 months ago	
dict_gen_3x3.py	documentation updated		8 months ago	
exif_pos.py	creation date added to output		27 days ago	
gcp_check.py	docs updated		2 months ago	
gcp_find.py	new commandline parameters to display GCPs		2 months ago	

☰ README.md ✎

Find-GCP

Forráskód



Önálló projekt

Segítőket szívesen látunk!

Tesztelés, dokumentálás, ...

FileEditViewHistoryBookmarksToolsHelp

Dashboard - WebODM x Blog - OpenDroneMap x +

←→↻🏠

🔒 https://www.opendronemap.org/blog/ 90% ... 📄 ⭐

📄 📄 📄 ☰

 OpenDroneMap

Projects ▾Community ▾DocumentationStoriesCode ▾

Blog

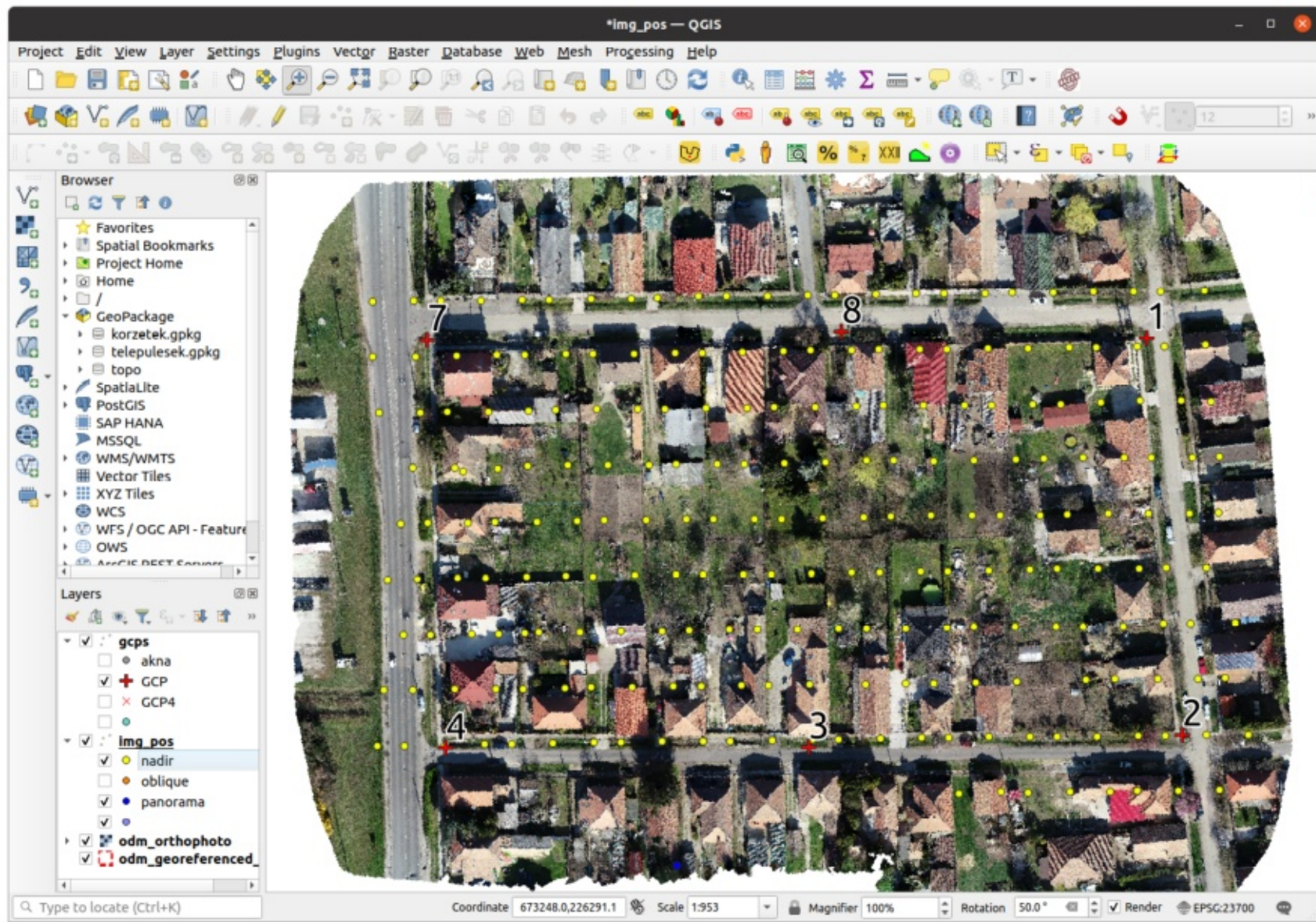


Automatize GCP image coordinate collection

This is a guest post authored by Zoltán Siki, the creator of the awesome Find-GCP project. Have you tried to create the `gcp_list.txt` file for a project with hundreds of images? It is a time-consuming and boring task to find each Ground Control Point (GCP) in 6-10 images and the chance of misidentification are high. [...]

QGIS 3.18

Pontfelhők a
GIS adatokkal
együtt





Nyílt forráskód ereje



Felhasznált fontosabb nyílt forráskódú komponensek
Python, NumPy, PIL, matplotlib, OpenCV, ODM, WebODM, Potree,
QGIS, pudb3, GeoEasy, ...

Python vagy C++?

Futási idők másodp.	Octave	Python	C++*	Python OpenCV	C++ OpenCV
Mona 182 x 276	1.7	0.7	0.2	0.4	0.1
Lab1 ~2k x 3k	466	235	28	1.2	0.8
Lab ~5.5k x 4k	2880	703	109	2.8	1.8

* Horváth Viktor Győző munkája



And the best part is, it's open source!

