



**Politechnika
Śląska**

14 października 2024,

Gliwice

Laboratorium ZAOWR

Rok akademicki	Termin	Rodzaj studiów	Kierunek	Prowadzący	Grupa
2024/2025	Czwartek, 10:00-13:15	Stacjonarne	Informatyka	Dr inż. Marcin Paszkuta Mgr inż. Mateusz Płonka	IGT

Ćwiczenie nr 1

Temat: Kalibracja kamer

Informatyka, IGT

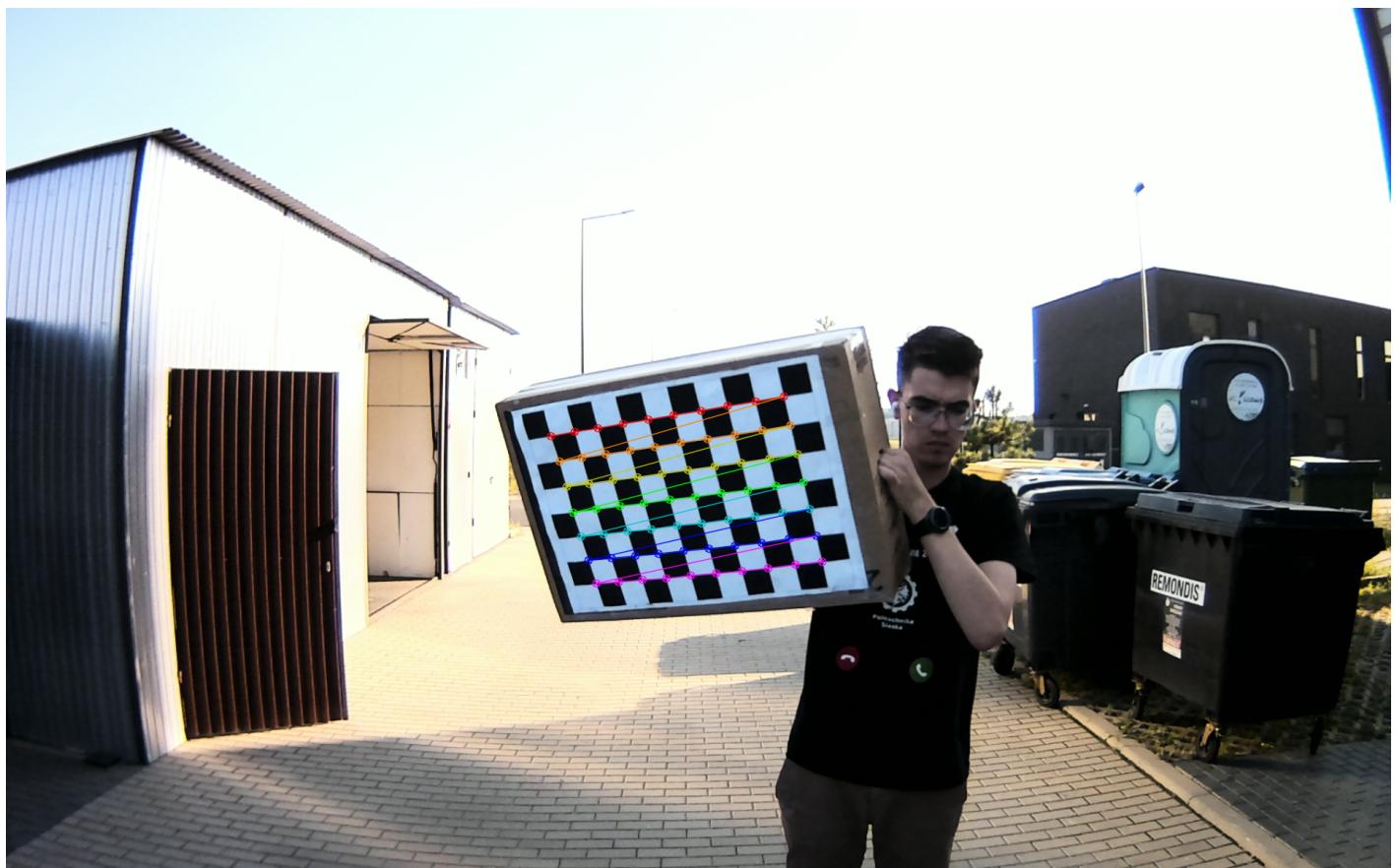
Maksymilian Kisiel

Spis treści

1 Wykrywanie wzorca kalibracyjnego na obrazie	2
2 Wyznaczanie macierzy parametrów wewnętrznych	3
3 Średni błąd reprojekcji	4
4 Zapis parametrów Kalibracji	4
5 Usuwanie dystorsji na obrazie - metoda <i>undistort</i>	5
6 Usuwanie dystorsji na obrazie - metoda <i>remap</i>	5
7 Kod	6

1 Wykrywanie wzorca kalibracyjnego na obrazie

Do wykrywania wzorca wykorzystane zostały zdjęcia dostarczone przez prowadzącego, a dokładniej zdjęcia z katalogu "ZAOWiR Image set - Calibration/Chessboard/Mono 1/cam4/".



Rys. 1: Wykryty wzorzec kalibracyjny

2 Wyznaczanie macierzy parametrów wewnętrznych

Analizując wyniki otrzymane poniżej (Kod 1 i Kod 2) możemy zauważyc, że próba poprawienia dokładności wykrytych narożników zakończyła się sukcesem, ponieważ średni błąd reprojekcji zmniejszył się o ~38% po zastosowaniu metody **cornerSubPix()**.

```
1 {  
2   "mse": 9.541713095159594,  
3   "rms": 0.8841990682295957,  
4   "cameraMatrix": [  
5     [1269.4880376329331, 0.0, 1060.1525393059474],  
6     [0.0, 1264.4300574118486, 617.133473144097],  
7     [0.0, 0.0, 1.0]  
8   ],  
9   "distortionCoefficients": [  
10    [-0.4013507573021153, 0.18285379875140967, 0.002597374870980325,  
11      -0.012461378215704717, -0.046416732648169215]  
12  ],  
13  "rotationVectors": [  
14    [[0.0073743030401264235], [0.16779005536926803],  
15      [0.019893429247728807]],  
16    [[-0.11508093990861244], [0.10590397003910658],  
17      [-0.0396273739169384]]  
18  ],  
19  "translationVectors": [  
20    [[69.61308374642202], [51.217792773732846], [1400.3104947237646]],  
21      [[-477.12600168210923], [-120.12970044612818],  
22        [801.1518606218784]]  
23 ]  
24 }
```

Kod 1: Fragment otrzymanej kalibracji dla metody **findChessboardCorners()**

```
1 {  
2   "mse": 5.984166144997382,  
3   "rms": 0.5399844606283781,  
4   "cameraMatrix": [  
5     [1272.011234078766, 0.0, 1058.4537673810164],  
6     [0.0, 1266.8726860857762, 617.7592332273604],  
7     [0.0, 0.0, 1.0]  
8   ],  
9   "distortionCoefficients": [  
10    [-0.399356477474748337, 0.18200290247627665,  
11      0.0020154085712910707, -0.012190829753206725,  
12      -0.04648398598417859]  
13  ],  
14  "rotationVectors": [  
15    [[0.014376302442723948], [0.1667778841470017],  
16      [0.018832348485715023]],  
17    [[-0.1173456057855872], [0.10779188735105613],  
18      [-0.03962029269228776]]  
19  ],  
20  "translationVectors": [  
21    [[71.27846898868391], [50.76036240921024], [1400.9402673825555]],  
22      [[-476.2081267995082], [-120.35757569213392], [803.862414335442]]  
23 ]  
24 }
```

Kod 2: Fragment otrzymanej kalibracji dla metody **cornerSubPix()**

3 Średni błąd reprojekcji

Otrzymany błąd reprojekcji MSE wyniósł odpowiednio

- 5.984166144997382 dla metody *cornerSubPix()*,
- 9.541713095159594 dla metody *findChessboardCorners()*.

4 Zapis parametrów Kalibracji

Aplikacja została rozszerzona o możliwość zapisu parametrów kalibracji do pliku.

Fragment odpowiedzialny za zapis przedstawiono poniżej (Kod 3).

```
1 from json import dump as jdump
2 from .exceptions import CalibrationParamsPathNotProvided
3
4 def save_calibration(
5     calibrationParams: dict[str, list], calibrationParamsPath: str
6 ) -> None:
7     """
8         Save the provided calibration parameters to specified JSON file
9
10    :raises CalibrationParamsPathNotProvided: Raises an error if the
11        path was not provided or it isn't an instance of a string.
12    """
13
14    if (calibrationParamsPath == "") or (not
15        isinstance(calibrationParamsPath, str)):
16        raise CalibrationParamsPathNotProvided
17
18    with open(calibrationParamsPath, "w", encoding="utf-8") as f:
19        jdump(calibrationParams, f, ensure_ascii=False, indent=4)
20
21 if saveCalibrationParams:
22     try:
23         from .save_calibration import save_calibration
24
25         calibrationParams = {
26             "mse": mse,
27             "rms",
28             "cameraMatrix": cameraMatrix.tolist(),
29             "distortionCoefficients": distortionCoefficients.tolist(),
30             "rotationVectors": [rvec.tolist() for rvec in
31             rotationVectors],
32             "translationVectors": [tvec.tolist() for tvec in
33             translationVectors],
34         }
35
36         save_calibration(calibrationParams, calibrationParamsPath)
37     except CalibrationParamsPathNotProvided:
38         print("\nError occurred while saving the calibration
39             parameters!\n")
40         raise
41
42     except Exception as e:
43         print("\nUnknown error occurred\n")
44         raise
```

Kod 3: Metoda odpowiedzialna za zapis parametrów do pliku

5 Usuwanie dystorsji na obrazie - metoda *undistort*



Obraz przed usunięciem dystorsji



Obraz po usunięciu dystorsji

Rys. 2: Porównanie obrazu oryginalnego i skorygowanego metodą **undistort**

6 Usuwanie dystorsji na obrazie - metoda *remap*



Obraz przed usunięciem dystorsji



Obraz po usunięciu dystorsji

Rys. 3: Porównanie obrazu oryginalnego i skorygowanego metodą **remap**

7 Kod

Na potrzeby tego laboratorium została przygotowana paczka, która zawiera metody potrzebne do wykonania zadań laboratoryjnych. Kod paczki jest dostępny w serwisie [GitHub](#), jednak kopia kodu źródłowego została załączona do sprawozdania w celu łatwej weryfikacji wyników.