

Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Інститут прикладного системного аналізу
Кафедра математичних методів системного аналізу

Звіт

про виконання лабораторної роботи №2
з дисципліни “Розпізнавання образів”

Виконали:
Студенти 4 курсу
Групи КА-76 і КА-72
Борбела Артур
Фалілеєва Дар’я

Київ 2020

Теоретичні данні:

ORB представлений в 2011г. В його основі лежить комбінація таких алгоритмів як детектор FAST (Features from Accelerated Segment Test) і дескриптор BRIEF (Binary Robust Independent Elementary Features) з некоторими удосконаленнями. Зображення розбивається на патчі. Нехай патч P має розміри $S \times S$ пікселів. З патча вибирається некоторим образом множество пар пікселів $\{(X, Y), \forall X, Y \text{ в околиці}\}$, для яких строится набір бінарних тестів:

$$\tau(P, X, Y) = \begin{cases} 1, & I(X) < I(Y) \\ 0, & \text{иначе} \end{cases}$$

де $I(Y)$, $I(X)$ – інтенсивність пікселів Y , X відповідно. Для кожного патча вибирається множество, що містить n_d пар точок, які однозначно визначають набір бінарних тестів. Далі на основі цих тестів строится бінарна строка:

$$f_{n_d}(P) = \sum_{1 \leq i \leq n_d} 2^{i-1} \tau(P, X_i, Y_i)$$

SIFT (Scale Invariant Feature Transform). Будується простір змінного масштабу, в ньому обчислюються функції LoG з різним параметром згладжування. Точка вважається ключовою, якщо вона є локальним екстремумом різниці Гауссіанів. Після безліч передбачуваних ключових точок уточнюється (видаляються точки з малим контрастом і на кордоні об'єктів) і обчислюються їх орієнтації. Для цього будується зважена гістограма градієнтів в околиці, вибирається напрямок, що відповідає максимальній компоненті гістограми. Точці присвоюються всі напрямки, яким відповідають значення компонент гістограми, великих заданого порогу. Інваріантний щодо зрушень, обертань, масштабу, що не зміщується локальні екстремуми

$G(x, y, \sigma)$ – гауссове ядро зі степінню розмиття σ

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

Для формування дескриптора SIFT (Scale Invariant Feature Transform) спочатку обчислюються значення магнітуди і орієнтації градієнта в кожному пікселі, що належить околиці особливої точки розміром $16 * 16$ пікселів. Магнітуди градієнтів при цьому враховуються з вагами, пропорційними значенню функції щільності нормального розподілу з математичним очікуванням в розглянутій особливій точці і стандартним відхиленням, рівним половині ширини околиці (ваги Гауссова розподілу використовуються для того, щоб зменшити вплив на підсумковий дескриптор градієнтів, обчислених в пікселях, які перебувають далі від особливої точки).

У кожному квадраті розміром $4 * 4$ пікселя обчислюється гістограма орієнтованих градієнтів шляхом додавання зваженого значення магнітуди градієнта до одного з 8 бінов гістограми. Щоб зменшити різні "граничні" ефекти, пов'язані з віднесенням схожих градієнтів до різних квадратів (що може виникнути внаслідок невеликого зсуву розташування особливої точки) використовується билинейна інтерполяція: значення магнітуди кожного градієнта додається не тільки в гістограму, відповідну квадрату, до якого даний піксель відноситься, але і до гістограми, відповідним сусіднім квадратах. При цьому значення магнітуди додається з вагою, пропорційним відстані від пікселя, в якому обчислений даний градієнт, до центру відповідного квадрата. Всі обчислені гістограми об'єднуються в один вектор, розміром, рівним 128

Отриманий дескриптор перетворюється, щоб зменшити можливі ефекти від зміни освітленості. Зміна контрасту зображення (значення інтенсивності кожного пікселя множиться на деяку константу) призводить до такого ж зміни в значеннях магнітуд градієнтів. Тому очевидно, що даний ефект може бути знівельовано шляхом нормалізації дескриптора таким чином, щоб його довжина стала дорівнює одиниці. Зміни яскравості зображення (до значення інтенсивності кожного пікселя додається деяка константа) не впливають на значення магнітуд градієнтів. Таким чином, SIFT-дескриптор є інваріантні по відношенню до афінних змін освітленості. Однак можуть виникати і нелінійні

зміни в освітленості внаслідок, наприклад, різної орієнтації джерела світла по відношенню до поверхонь тривимірного об'єкту. Дані ефекти можуть викликати велика зміна в ставленні магнітуд деяких градієнтів (при цьому мають незначний вплив на орієнтацію вектора градієнта). Щоб уникнути цього, використовують відсікання по деякому порогу (за результатами експериментів показано, що оптимальним є значення 0.2), яке застосовують до компонентів нормалізованого дескриптора. Після застосування порога дескриптор знову нормалізується. Таким чином, зменшується значення великих магнітуд градієнтів і збільшується значення розподілу орієнтацій даних градієнтів в околиці особливої точки.

Отримані дані :

Датасет 1

src/result/ORB/ result_correct *src/result/ORB/ result_wrong*

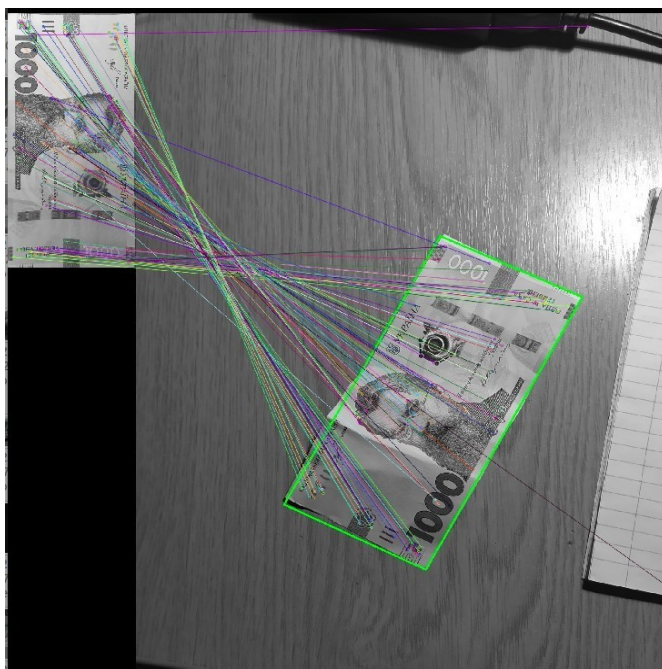
src/result/SIFT/ result_correct *src/result/SIFT/ result_wrong*

Датасет 2

src/result1/ORB/result_correct *src/result1/ORB/result_wrong*

src/result1/SIFT/result_correct *src/result1/SIFT/result_wrong*

Зразок



Пояснення, що до назв директорій та зображень :

Якщо директрія має шлях ./result/ORB — це результати роботи алгоритму ORB, аналогічно й з алгоритмом SIFT

Якщо в шляхові є

...../correct_data - це зображення на яких справді є предмет розпізнавання;

...../wrong_data — зображення на якому предмета немає;

Назва зображень в результаті роботи може бути :

1602152796.5985005_jopa.jpg

1602152825.7172604_undefined_jopa.jpg

1602152798.2051291_good.jpg

resize3243536 -//- .jpg

Пояснення:

....._jopa - не пройшла перевірку, тобто в моєму випадку не розпізнано більше десяти характерних точок із-за цього не намагаємось визначити контури об'єкту.

....._good — характерних точок більше десяти, малюємо контур

об'єкта._undefined_jopa - все погано (не був згенерований дескриптор або некоректно знайдені matches що взаємо пов'язане)

Якщо починається на resize... зображення було масштабоване.

Дані функцій порівняння:

Датасет 1

```
artur@pkzver: ~/PycharmProjects/image_recognition/opencv
File Edit View Search Terminal Help
[+] Stopped python3 lab2.py
artur@pkzver:~/PycharmProjects/opencv_lab1/lab2$ python3 lab2.py

Correct Data resize
Time win: ORB 53; SIFT 43
Correct win: ORB 0; SIFT 96
Distance win: ORB 0; SIFT 96
Full time ORB : 80.31812715530396
Full time SIFT : 81.14322710037231
Full Correct ORB: 4.0701071080817925
Full Correct SIFT: 4.0701071080817925
Full Dist ORB: 28278.94193098817
Full Dist SIFT: 28278.94193098817

artur@pkzver:~/PycharmProjects/opencv_lab1/lab2$ python3 lab2.py
Traceback (most recent call last):
  File "lab2.py", line 273, in <module>
    print(compare_correct_data(origin, data_correct))
NameError: name 'data_correct' is not defined
artur@pkzver:~/PycharmProjects/opencv_lab1/lab2$ python3 lab2.py

Correct Data
Time win: ORB 68; SIFT 28
Correct win: ORB 0; SIFT 96
Distance win: ORB 0; SIFT 96
Full time ORB : 297.4426643848419
Full time SIFT : 299.8878126144409
Full Correct ORB: 3.632911392405064
Full Correct SIFT: 3.632911392405064
Full Dist ORB: 26940.30632654578
Full Dist SIFT: 26940.30632654578

artur@pkzver:~/PycharmProjects/opencv_lab1/lab2$ cd ../../
```

```
artur@pkzver: ~/PycharmProjects/opencv_lab1/lab2
File Edit View Search Terminal Tabs Help
artur@pkzver: ~/program/pycharm/pycha... x artur@pkzver: ~/PycharmProjects/opencv... x

artur@pkzver:~/PycharmProjects/opencv_lab1/lab2$ python3 lab2.py

Wrong Data
Time win: ORB 10; SIFT 10
Correct win: ORB 0; SIFT 20
Distance win: ORB 0; SIFT 20
Full time ORB : 69.06411814689636
Full time SIFT : 68.89301323890686
Full Correct ORB: 0.0730282375851996
Full Correct SIFT: 0.0730282375851996
Full Dist ORB: 5908.033792579302
Full Dist SIFT: 5908.033792579302

Wrong Data resize
Time win: ORB 12; SIFT 8
Correct win: ORB 0; SIFT 20
Distance win: ORB 0; SIFT 20
Full time ORB : 18.917373180389404
Full time SIFT : 18.870035409927368
Full Correct ORB: 0.09542356377799417
Full Correct SIFT: 0.09542356377799417
Full Dist ORB: 6151.141927096953
Full Dist SIFT: 6151.141927096953

artur@pkzver:~/PycharmProjects/opencv_lab1/lab2$
```

Датасет 2

CorrectData resize

Time win: ORB 52; SIFT 48
Correct win: ORB 0; SIFT 100
Distance win: ORB 0; SIFT 100
Full time ORB : 10.805208206176758
Full time SIFT : 10.82294750213623
Full Correct ORB: 8.592356687898098
Full Correct SIFT: 8.592356687898098
Full Dist ORB: 28904.135509175096
Full Dist SIFT: 28904.135509175096

CorrectData

Time win: ORB 43; SIFT 57
Correct win: ORB 0; SIFT 100
Distance win: ORB 0; SIFT 100
Full time ORB : 30.156322956085205
Full time SIFT : 30.054237604141235
Full Correct ORB: 8.01273885350319
Full Correct SIFT: 8.01273885350319
Full Dist ORB: 27184.438814746347
Full Dist SIFT: 27184.438814746347

WrongData resize

Time win: ORB 10; SIFT 10
Correct win: ORB 0; SIFT 20
Distance win: ORB 0; SIFT 20
Full time ORB : 2.2352287769317627
Full time SIFT : 2.2325305938720703
Full Correct ORB: 0.8662420382165604
Full Correct SIFT: 0.8662420382165604
Full Dist ORB: 5359.003339293656
Full Dist SIFT: 5359.003339293656

WrongData

Time win: ORB 12; SIFT 8
Correct win: ORB 0; SIFT 20
Distance win: ORB 0; SIFT 20
Full time ORB : 6.124811887741089
Full time SIFT : 6.151655197143555
Full Correct ORB: 0.8471337579617834
Full Correct SIFT: 0.8471337579617834
Full Dist ORB: 4845.924688229894

Full Dist SIFT: 4845.924688229894

Записи які починаються на Full були зроблені для визначення впливу розмірів зображення на результат алгоритмів.

Time win , correct win, distance win :

Ми беремо одне зображення отримуємо метрики після першого алгоритму, потім отримуємо метрики з другого алгоритму за цим же зображенням. Потім ми порівнюємо ці метрики між собою й визначаємо який алгоритм кращим за певним критерієм наприклад

Time win: ORB 10; SIFT 10 — цей запис значить що по метриці швидкість для 10 зображень був кращий алгоритм ORB, а для інших 10 - SIFT

Time — час за який відпрацьовує алгоритм

Correct — відношення good_point до всіх характерних точок (good_point — хар. точки що проходять перевірку локалізації)

Distance — середня похибка локалізації

Порівняння дескрипторів:

	Вірні дані	Вірні масштабовані і дані	Невірні дані	Невірні масштабовані дані
Швидкість	Датасет 1 ORB	Датасет 1 ORB	Датасет 1 ORB/SIFT	Датасет 1 ORB
	Датасет 2 SIFT	Датасет 2 ORB	Датасет 2 ORB	Датасет 2 ORB/SIFT
Відповідність зображенню	Датасет 1 SIFT	Датасет 1 SIFT	Датасет 1 SIFT	Датасет 1 SIFT
	Датасет 2 SIFT	Датасет 2 SIFT	Датасет 2 SIFT	Датасет 2 SIFT
Похибка	Датасет 1	Датасет 1	Датасет 1	Датасет 1

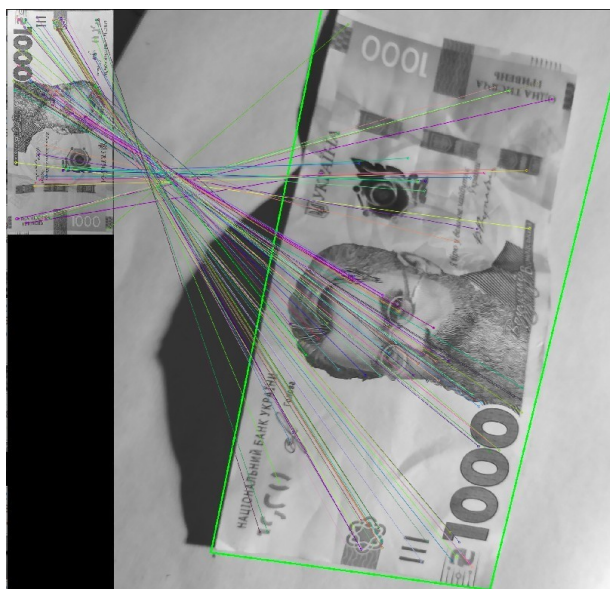
локалізації	SIFT	SIFT	SIFT	SIFT
	Датасет 2 SIFT	Датасет 2 SIFT	Датасет 2 SIFT	Датасет 2 SIFT

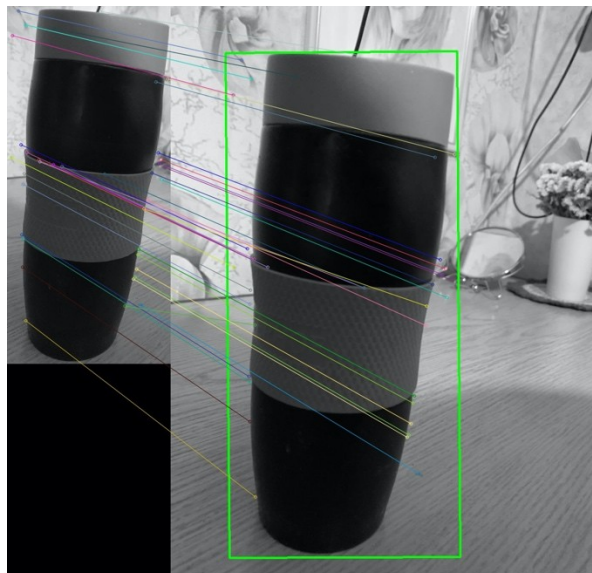
Висновки :

- 1) Зменшений розмір зображення позитивно впливає на швидкість відпрацювання алгоритму (аж в 3-рази) й не як не впливає на два інші показники(майже не змінились).
- 2) ORB працює швидше, але фатально програє в точності SIFT для двох Датасетів.

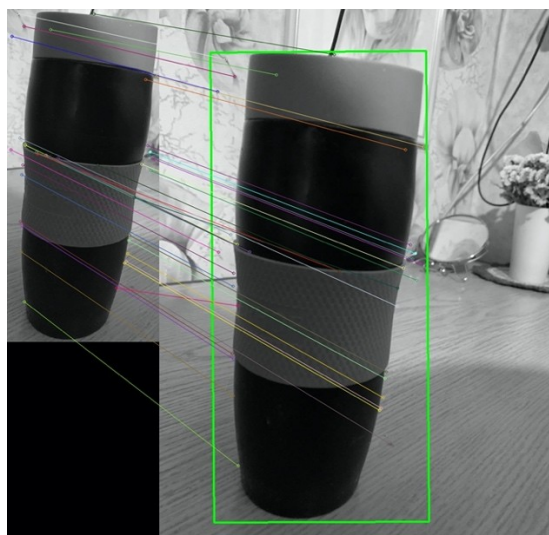
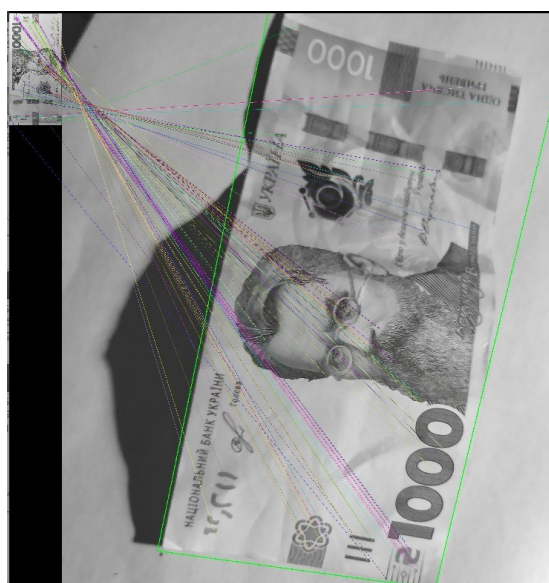
З графічних даних видно що кількість характерних точок за алгоритмом orb завжди менша на 15-20%.

SIFT





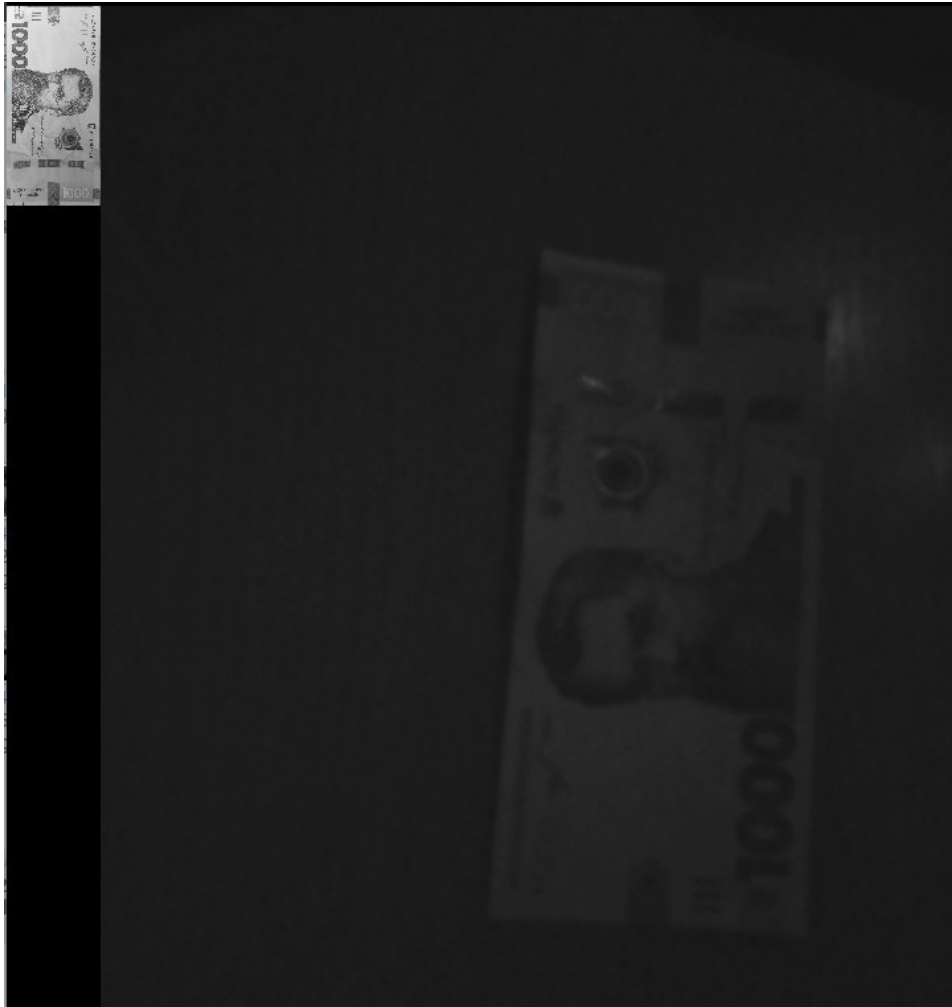
ORB

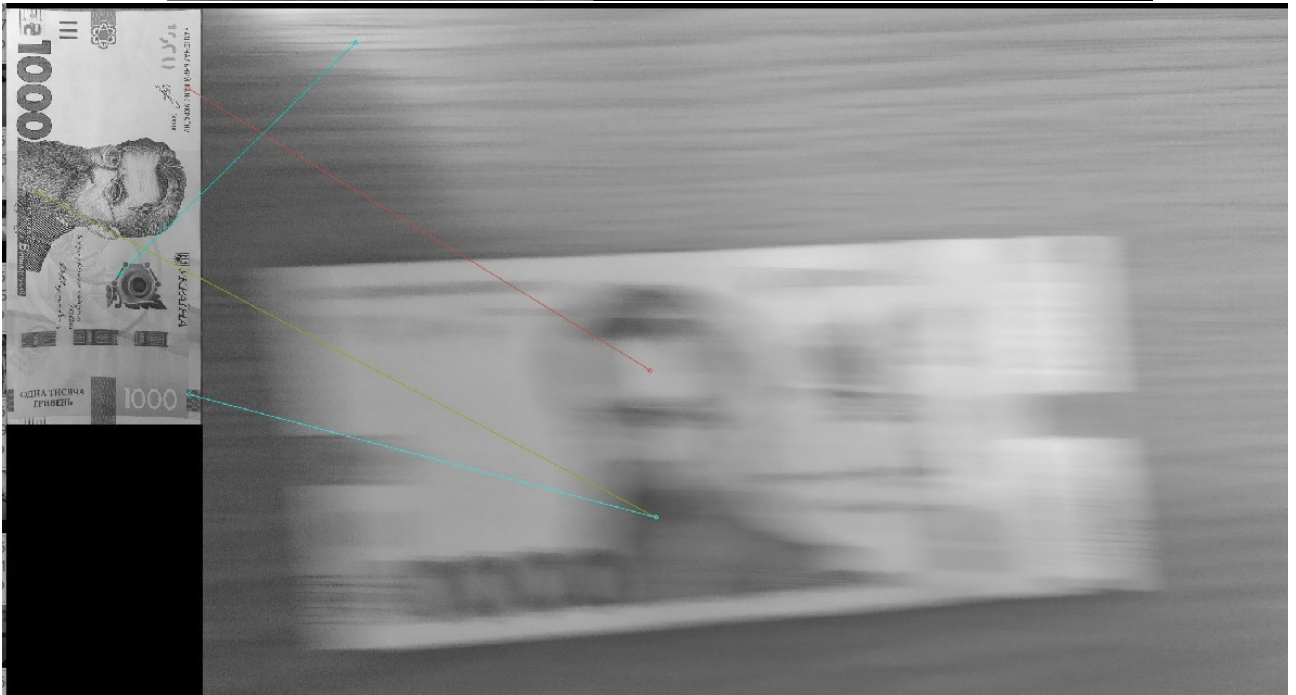
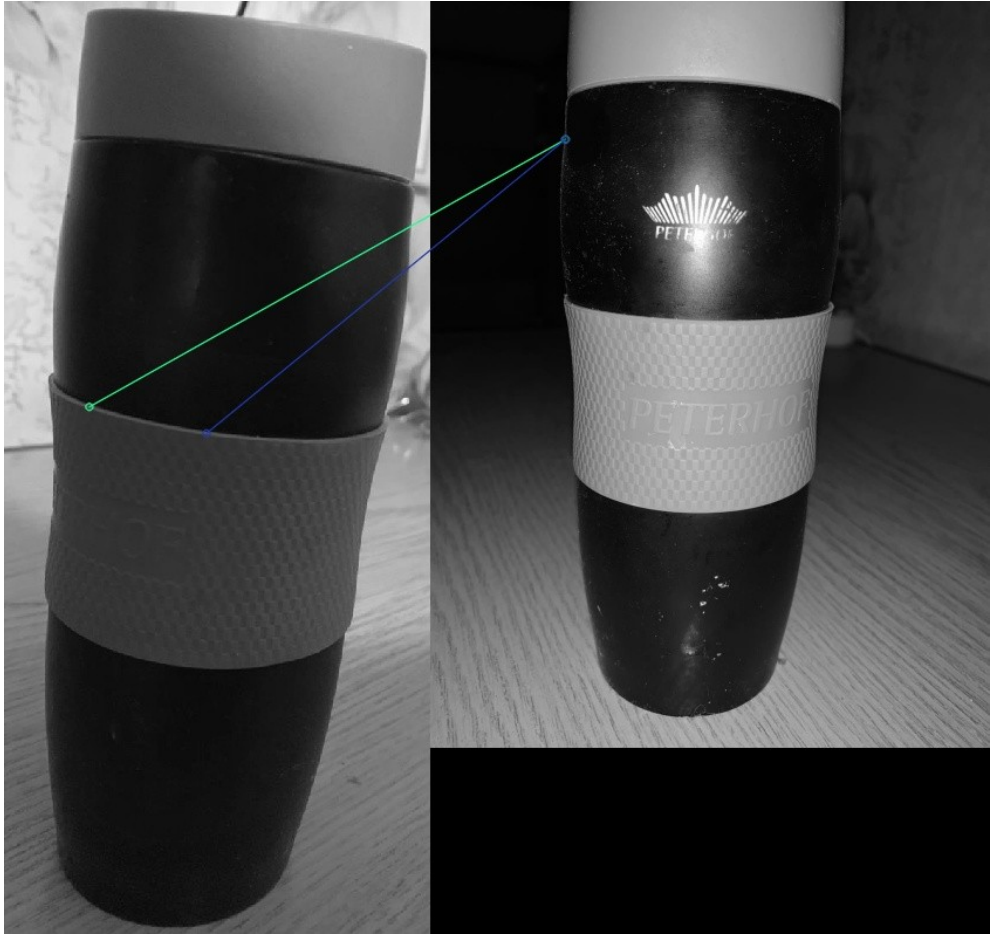


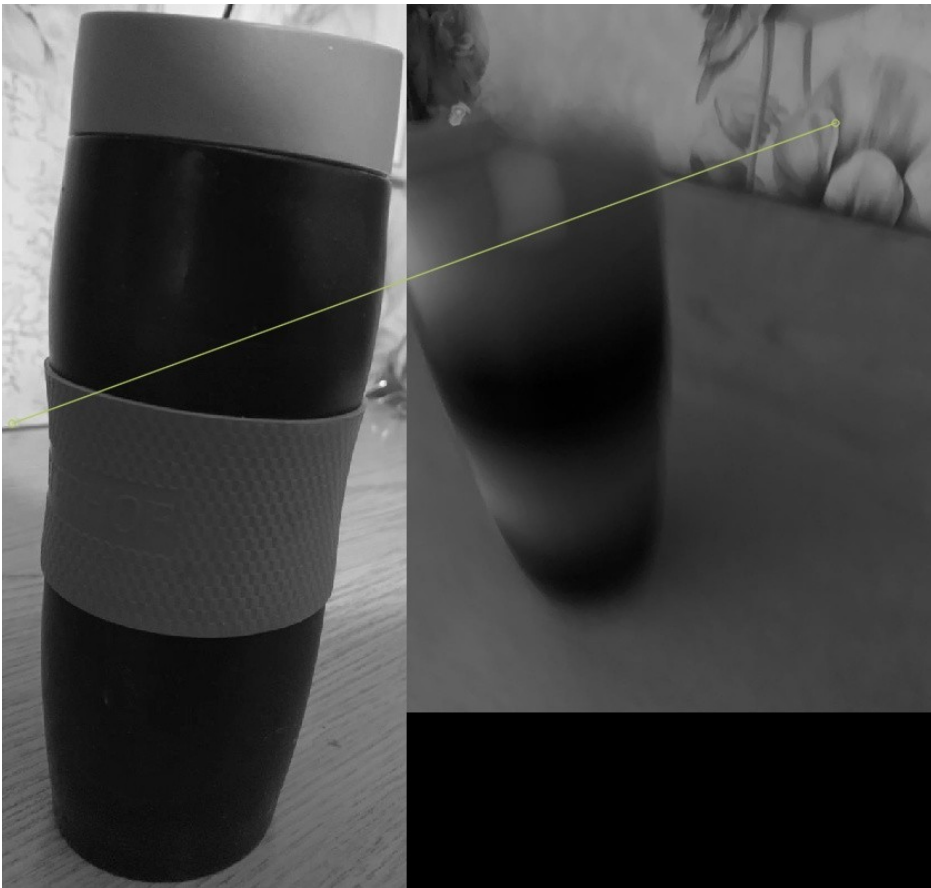
Обидва алгоритма добре розпізнають об'єкт під кутом нахилу не більше 70%, та з любого ракурсу. Також був вражений точністю розпізнавання об'єкта за його певною частиною.



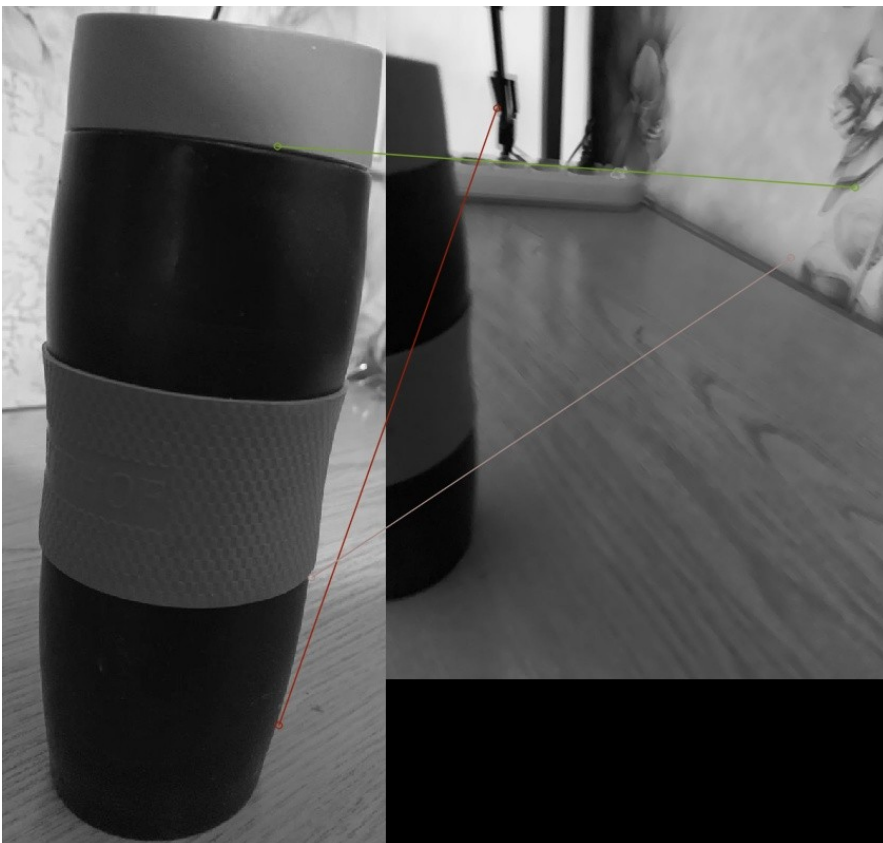
Але погано, обидва алгоритма безсильні перед сильним затемненням, фотоспахом , та при динаміці (коли зображення розмите)







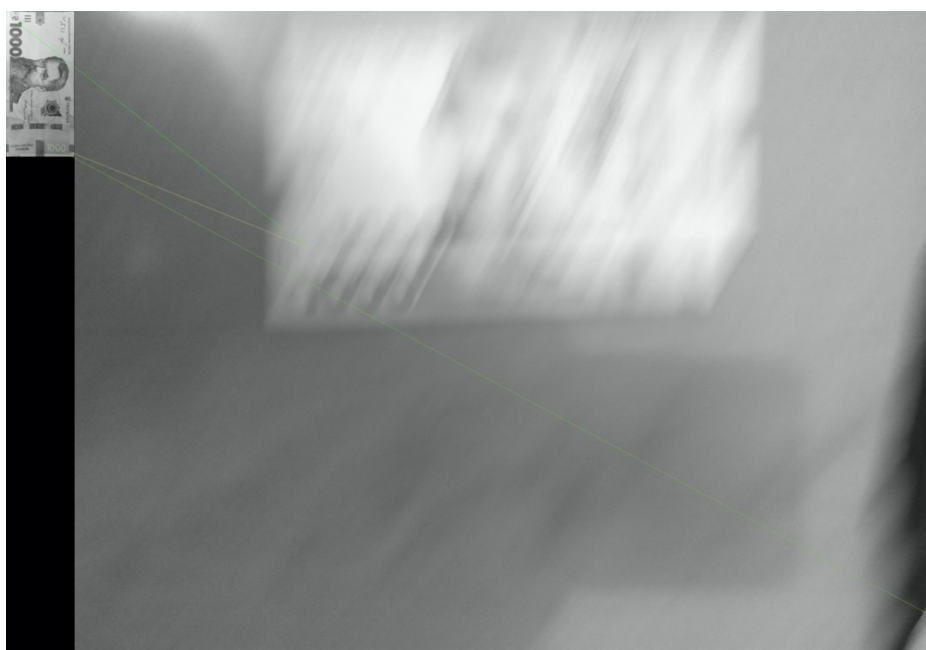
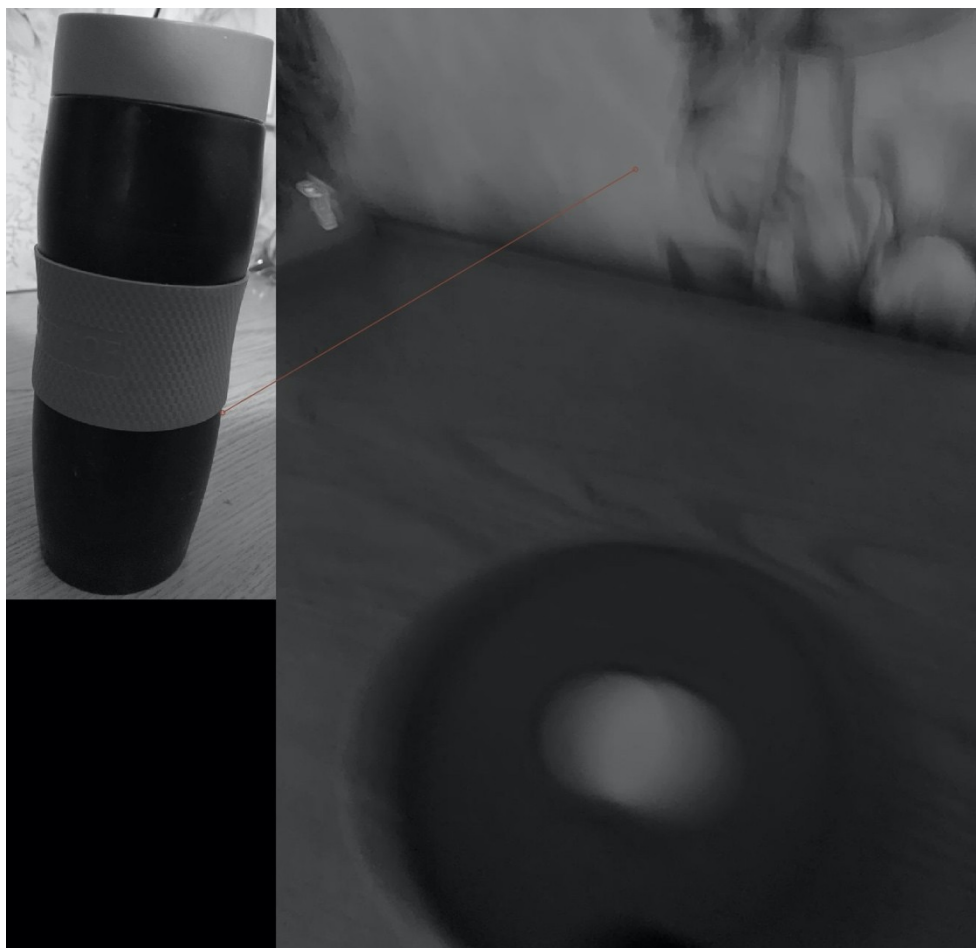
Також алгоритм погано працює при частковому зображенні предмета



На зображеннях де об'єкт відсутній обидва алгоритми не розпізнавали нічого навіть характерних точок не більше 3- 4 хоч об'єкт дуже подібний до оригіналу



З найбільшою різницею в часі між алгоритмами виявилось частково зображені предмети в розмитій якості



=> алгоритми виявились дуже схожі на даному наборі зображення
дуже схожі =>
якщо нам важливий час вибираємо ORB якщо точність
(кількість характерних точок) то SIFT