

Podobieństwo obrazów

Szymon Woźniak, 235040

16.05.2019

1 Wstęp teoretyczny

1.1 Cel zadania

Celem zadania jest określania podobieństwa obrazów poprzez poszukiwanie obiektów występujących na obu.

1.2 Punkty kluczowe

Punkty kluczowe opisują lokalną charakterystykę wizualną danego obrazu. Każdy punkt kluczowy opisany jest wektorem cech. Do celów zadania zostanie użyty gotowe narzędzie do ekstrakcji cech, korzystający z detektora Harris-Affine i deskryptora SIFT.

1.3 Wyznaczanie par punktów kluczowych

Do wyznaczania par punktów kluczowych zastosowany zostanie użyty algorytm wzajemnego najbliższego sąsiada. Dla każdego punktu z jednego z obrazów zostaje wyznaczony punkt z drugiego obrazu będący najbliższym do niego w sensie podobieństwa cech. Ta sama procedura jest przeprowadzana dla punktów z drugiego obrazu. Para punktów (P, Q) zostaje utworzona tylko wtedy, gdy najbliższym punktem dla punktu P jest punkt Q , a najbliższym dla punktu Q jest punkt P . Funkcja odległości używana do porównywania punktów kluczowych ma postać:

$$distance(P, Q) = \sum_{i=1}^n |features(P, i) - features(Q, i)|, \quad (1)$$

gdzie:

- n jest liczbą cech punktu kluczowego,
- $features(R, i)$ jest wartość i -tej cechy z punktu R .

1.4 Algorytm analizy spójności sąsiedztwa

Pary punktów kluczowych wyznaczanych przy pomocy algorytmu wzajemnego najbliższego sąsiada nie są wystarczające do wyznaczania podobieństwa obrazów. Jest tak ponieważ znacząca liczba znajdowanych par stanowi szum informacyjny.

Do odfiltrowania nieznaczących par można wykorzystać algorytm analizy spójności sąsiedztwa. Algorytm ten na wejściu przyjmuje zbiór par znalezionych na obrazach. Zachowywane są tylko te pary, które spełniają założone kryterium spójności. Najpierw dla każdego punktu na obu obrazach wyznaczane zostaje jego sąsiedztwo. Sąsiedztwem nazywany jest zbiór punktów o wielkości k , z tego samego obrazu, których odległość od zadanego punktu jest najmniejsza i większa od 0.

Para punktów kluczowych (Q_i, Q_j) sąsiaduje z parą punktów (P_i, P_j) jeżeli punkt Q_i należy do sąsiedztwa punktu P_i , a punkt Q_j należy do sąsiedztwa punktu P_j . Sąsiedztwo pary punktów kluczowych jest nazywane **spójnym** jeżeli, para punktów kluczowych sąsiaduje z co najmniej n innymi parą punktów kluczowych.

1.5 Metoda Random Sample Consensus

Metoda Random Sample Consensus ("RANSAC") jest metodą pseudolosową służącą do estymacji parametrów modelu uczenia maszynowego. Jest ona zdolna wyznaczać parametry modelu nawet dla danych o dużej liczbie danych odstających. Na początku metoda losuje z całego zbioru danych D minimalną próbę danych pozwalającą na przeprowadzenie uczenia. W kolejnym kroku przeprowadza się uczenie. Na koniec oceniana jest jakość modelu. Każda obserwacja sprawdzana jest pod kątem zgodności z modelem. Jeżeli wartość błędu nie przekracza zadanego progu, to mówi się że dana obserwacja jest zgodna z modelem. Obserwacje zgodne tworzą **konsensus**, a jego rozmiar oddaje ogólną jakość modelu. W metodzie RANSAC przeprowadzanych jest i iteracji opisanych wcześniej, a wynikiem działania jest model posiadający największy konsensus. W opisywanym jako modele użyte zostaną dwa opisane poniżej przekształcenia geometryczne.

1.5.1 Model - transformata afiniczna

Transformata afiniczna jest złożeniem trzech przekształceń elementarnych - obrót, skalowanie i translacja. Może zostać wyznaczona przy użyciu trzech par punktów kluczowych w następujący sposób:

$$\begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = \begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 \\ x_2 & y_2 & 1 & 0 & 0 & 0 \\ x_3 & y_3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1 & y_1 & 1 \\ 0 & 0 & 0 & x_2 & y_2 & 1 \\ 0 & 0 & 0 & x_3 & y_3 & 1 \end{bmatrix}^{-1} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ v_1 \\ v_2 \\ v_3 \end{bmatrix} \quad (2)$$

$$\mathbf{A} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

1.5.2 Model - transformata perspektywiczna

Transformata perspektywiczna rozszerza transformatę afiniczną o efekt rozcięcia/gnięcia/spłaszczenia wynikający z perspektywy. Można ją wyznaczyć przy po-

mocy 4 par punktów kluczowych według poniższych równań:

$$\begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} = \begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -u_1x_1 & -u_1y_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -u_2x_2 & -u_2y_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -u_3x_3 & -u_3y_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -u_4x_4 & -u_4y_4 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -v_1x_1 & -v_1y_1 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -v_2x_2 & -v_2y_2 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -v_3x_3 & -v_3y_3 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -v_4x_4 & -v_4y_4 \end{bmatrix}^{-1} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} \quad (4)$$

$$\mathbf{H} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix} \quad (5)$$

1.6 Heurystyki przyspieszające metodę RANSAC

1.6.1 Heurystyki odległości par punktów

Ta heurystyka mówi, że jako próbka do algorytmu RANSAC powinny być wybierane punkty leżące stosunkowo blisko siebie. Ograniczeniem może być na przykład to, że odległość pomiędzy każdą parą wybranych punktów powinna być większa niż 1% i mniejsza niż 30% wielkości obrazu.

1.6.2 Heurystyka modyfikacji rozkładu

Ta heurystyka mówi, że rozkład prawdopodobieństwa wyboru par punktów kluczowych powinien być modyfikowany w trakcie działania metody. Pary punktów, które uzyskiwały większe rozmiary konsensusu powinny być losowane z większym prawdopodobieństwem.

1.6.3 Heurystyka liczby iteracji

W podstawowej wersji metody RANSAC liczba iteracji jest stała i musi zostać podana przez użytkownika. Jeżeli jednak podana liczba iteracji będzie zbyt duża w stosunku do trudności problemu, to od pewnego momentu metoda może nie generować już lepszych modeli. Liczbę iteracji można jednak oszacować zgodnie ze wzorem:

$$k = \frac{\log(1-p)}{\log(1-w^n)} \quad (6)$$

, gdzie:

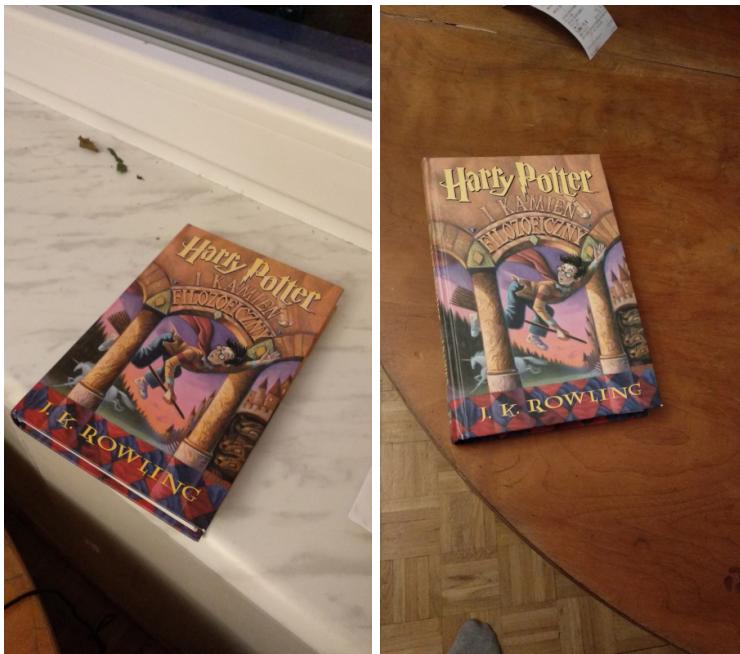
- p - prawdopodobieństwo, że po oszacowanej liczbie iteracji model jest dostatecznie dobrym przybliżeniem danych,
- w - prawdopodobieństwo, że wylosowana para nie jest szumem,
- n - liczba par potrzebnych do wyznaczenia parametrów transformaty (3 lub 4),

- k - szacowana liczba iteracji.

Prawdopodobieństwo p musi zostać zadane przez użytkownika. Prawdopodobieństwo w może zostać oszacowane jako stosunek liczby par przed i po zastosowaniu algorytmu analizy spójności sąsiedztwa.

2 Badania

2.1 Wykorzystane pary obrazów



Rysunek 1: Książka "Harry Potter i Kamień Filozoficzny" w dwóch różnych ujęciach.



Rysunek 2: Książka Harry Potter i Przeklęte Dziecko w dwóch różnych ujęciach.



Rysunek 3: Książka "Brudne Gry" w dwóch różnych ujęciach.



Rysunek 4: Kubek w dwóch różnych ujęciach.



Rysunek 5: Ossolineum w dwóch różnych ujęciach.

2.2 Skuteczność algorytmu analizy spójności sąsiedztwa

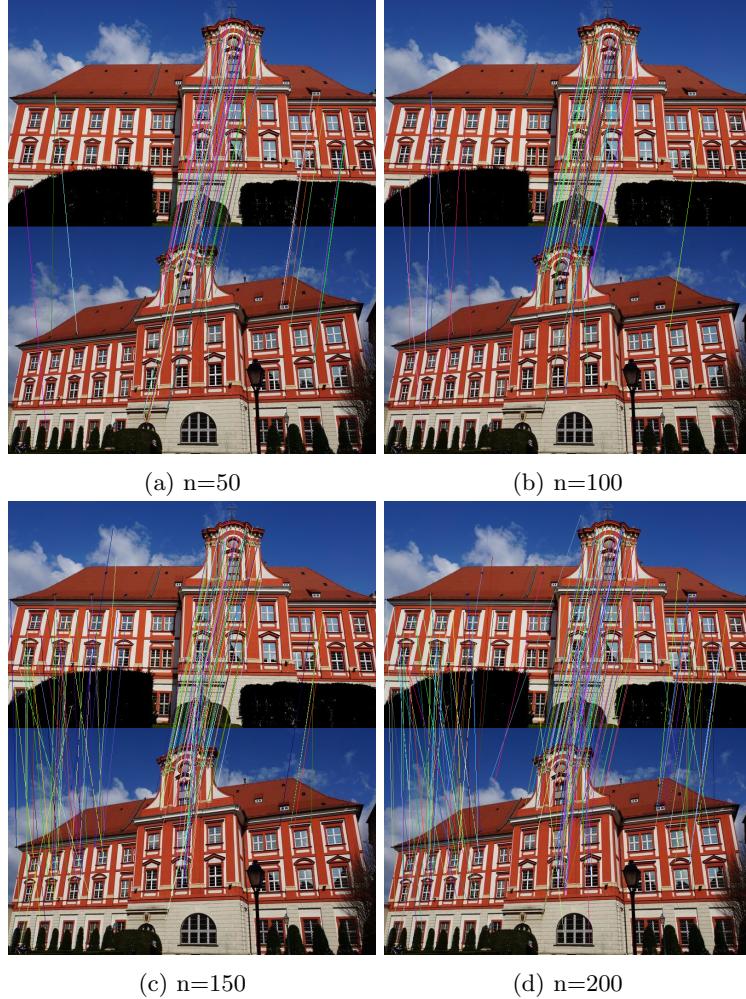
Wyniki badań skuteczność algorytmu spójności sąsiedztwa zostaną przedstawione jako porównanie dwóch par obrazów z naniesionymi parami punktów kluczowych. Jedna para będzie reprezentować wszystkie znalezione pary, a druga pary po zastosowaniu algorytmu analizy spójności sąsiedztwa. Do przeprowadzenia tych badań wykorzystane zostaną zdjęcia z rysunku 5.



Rysunek 6: Wszystkie pary punktów kluczowych znalezione na obrazach Osolineum

2.2.1 Wpływ wielkości sąsiedztwa na skuteczność algorytmu spójności

Wielkość sąsiedztwa została zbadana przy wartości progu spójności ustawionej na 0.5.



Rysunek 7: Pary punktów kluczowych odfiltrowane za pomocą algorytmu spójności sąsiedztwa dla różnych wartości wielkości sąsiedztwa.

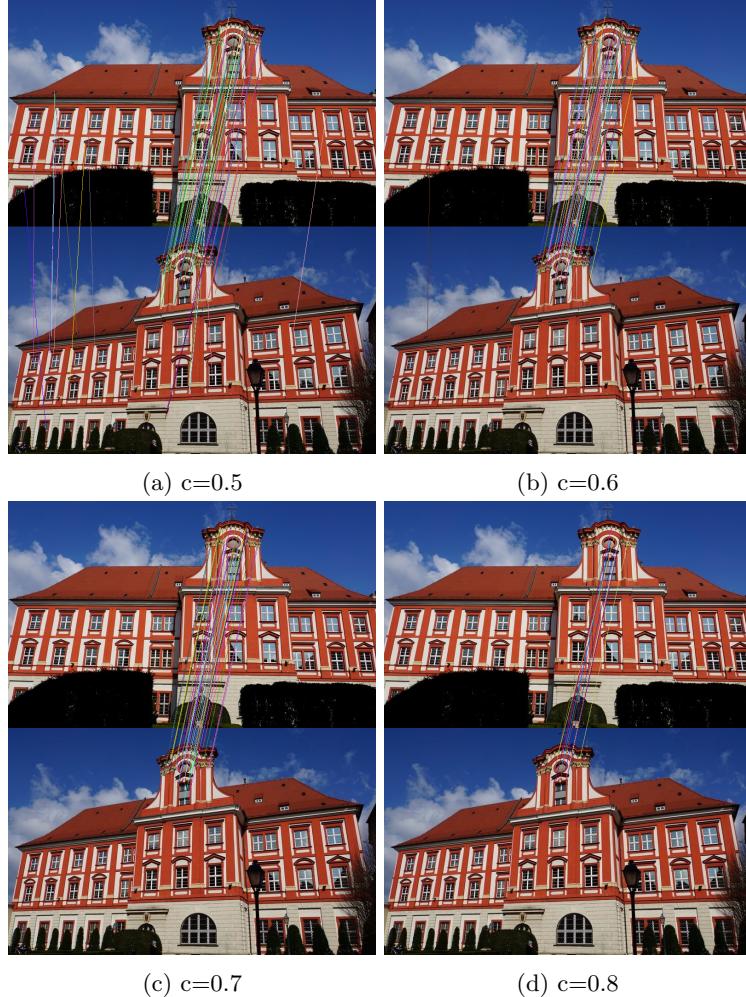
Tabela 1: Porównanie liczby zaakceptowanych par punktów kluczowych i czasu przetwarzania w zależności od wielkości sąsiedztwa

Wielkość sąsiedztwa	Liczba par spójnych	Czas[ms]
50	97	180
100	88	190
150	141	200
200	188	215

Wnioski Ustalenie dobrej wartości parametru wielkości sąsiedztwa jest istotnym wymogiem poprawnego działania algorytmu analizy spójności sąsiedztwa. Przy dobieraniu wartości tego parametru można kierować się gęstością punktów kluczowych na analizowanych obrazach. Im jest ona większa tym większa powinna być ustalona wielkość sąsiedztwa. Jak widać na rysunku 7 przy zbyt małych wartościach tego parametru algorytm może pominać istotne punkty. Z kolei jeżeli jest on za duży, to mogą zostać znalezione pary będące szumem. Dla analizowanego obrazu wartości pomiędzy 100 a 150 wydają się być odpowiednie. Jak widać w tabeli 1, wpływ ustalonej wartości tego parametru na czas działania algorytmu istnieje, natomiast jest stosunkowo mały i prawdopodobnie może zostać pominięty.

2.2.2 Wpływ progu spójności na skuteczność algorytmu

Wpływ progu spójności sąsiedztwa został zbadany przy wartości wielkości sąsiedztwa ustawionej na $n=100$.



Rysunek 8: Pary punktów kluczowych odfiltrowane za pomocą algorytmu spójności sąsiedztwa dla różnych wartości progu spójności.

Tabela 2: Porównanie liczby zaakceptowanych par punktów kluczowych w zależności od wartości progu spójności sąsiedztwa

Próg spójności	Liczba par spójnych
0.5	88
0.6	62
0.7	54
0.8	16
0.9	0

Wnioski Ustalenie dobrej wartości parametru wielkości sąsiedztwa jest kolejną istotną kwestią przy dobieraniu parametrów do algorytmu analizy spójności sąsiedztwa. Wartość tego parametru jest powiązana z wielkością sąsiedztwa więc wartości dla obu parametry powinny być ustalane równocześnie. Jak widać na rysunku 8 zbyt małe wartości tego parametru mogą skutkować akceptowaniem par stanowiących szum. Z kolei zbyt duże wartości mogą powodować zgubienie istotnej, pod kątem analizy podobieństwa, informacji.

Wpływ wielkości tego parametru na czas działania algorytmu okazał się nieznaczący.

2.3 Skuteczność działania metody RANSAC

2.3.1 Wpływ rodzaju transformaty

Badania w tej sekcji zostaną przedstawione jako zestawienia wyników działania metody RANSAC na dwóch różnych obrazach przy wykorzystaniu transformaty afinicznej i transformaty perspektywicznej.

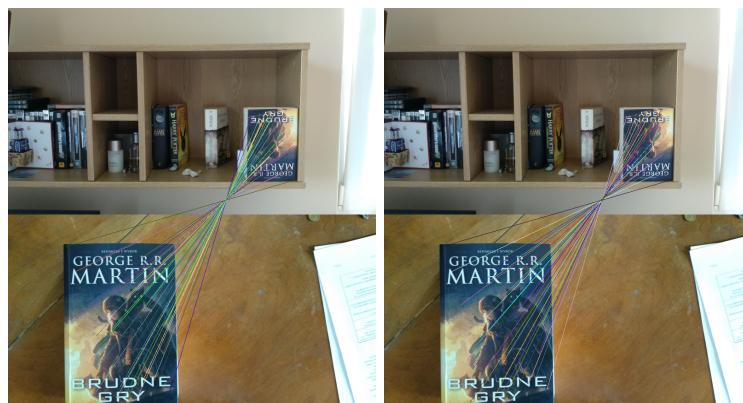
Wszystkie badania w tej sekcji zostały przeprowadzone bez użycia heurystyk, z wartościami parametrów:

- liczba iteracji = 10000,
- próg akceptacji = 1.

Para obrazów bez zmiany perspektywy Na tej parze obrazów, obiekt jest w obu przypadkach widoczny od frontu. Liczba wszystkich par punktów kluczowych na obrazie wynosi 371.



Rysunek 9: Wszystkie pary punktów kluczowych znalezione na zdjęciach książki "Brudne gry".



(a) Transformata afiniczna

(b) Transformata perspektywiczna

Rysunek 10: Pary punktów kluczowych znalezione przez metodę RANSAC na zdjęciach książki "Brudne gry" przy użyciu obu przekształceń geometrycznych.



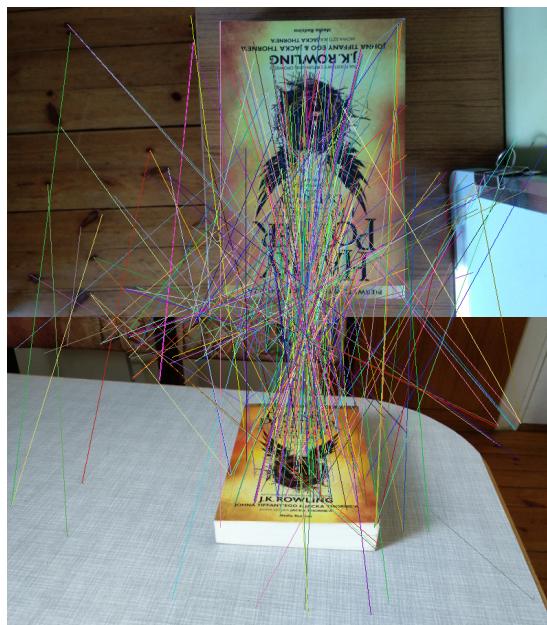
(a) Transformata afinicza



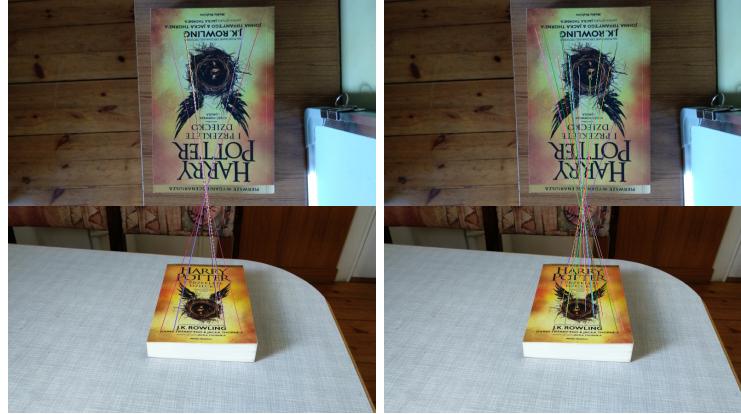
(b) Transformata perspektywiczna

Rysunek 11: Najlepsze znalezione przekształcenia geometryczne zaaplikowane do całego pierwszego zdjęcia.

Para obrazów ze zmianą perspektywy Na tej parze obrazów, kąt patrzenia na obiekt się zmienia. Liczba par punktów kluczowych na obrazach wynosi 328.



Rysunek 12: Wszystkie pary punktów kluczowych znalezione na zdjęciach książki "Harry Potter i przeklęte dziecko".



(a) Transformata afinicza (b) Transformata perspektywiczna

Rysunek 13: Pary punktów kluczowych znalezione na zdjęciach książki "Harry Potter i przeklęte dziecko" przez metodę RANSAC przy użyciu obu przekształceń geometrycznych.



(a) Transformata afinicza (b) Transformata perspektywiczna

Rysunek 14: Najlepsze znalezione przekształcenia geometryczne zaaplikowane do całego pierwszego zdjęcia.

Tabela 3: Liczba par punktów kluczowych znalezionych przez metodę RANSAC na obrazach z perspektywą i bez perspektywy (średnia z 10 przebiegów)

	Transformata afinicza	Transformata perspektywiczna
Bez perspektywy	63	60
Z perspektywą	17	35

Wnioski Jak widać na rysunku 10 i w tabeli 3 w przypadku pierwszego obrazu, obie transformacje poradziły sobie podobnie. Widać również jednak, że rozwiązanie z transformacją perspektywiczną znajduje średnio mniej par, co może być związane z większą przestrzenią poszukiwań i stałą liczbą iteracji. Na rysunkach 13 14 widać jednak, że w momencie kiedy obiekt na jednym z obrazów znajduje się pod innym kątem, transformacja perspektywiczna radzi

sobie zdecydowanie lepiej od afanicznej. Jest to spowodowane ograniczeniami wynikającymi wprost z modelu - transformacja afaniczna nie jest w stanie reprezentować perspektywy.

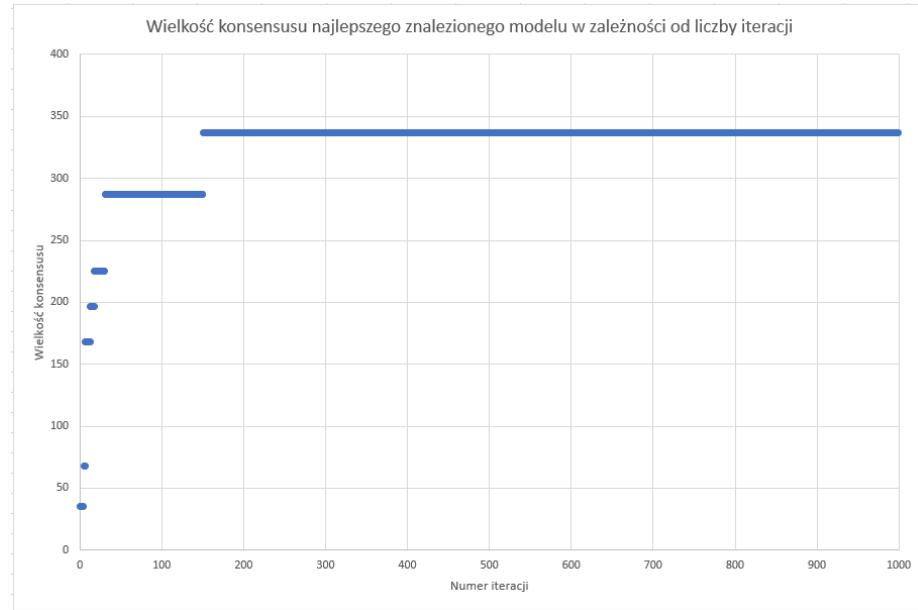
2.3.2 Wpływ liczby iteracji

Badanie wpływu iteracji zostało przeprowadzone na parze obrazów z rysunku 1. Ustalone wartości parametrów to:

- próg akceptacji - 1,
- typ transformacji - perspektywiczna.

Tabela 4: Średnia liczba par pasujących do transformacji i średni czas działania dla różnych wartości liczby iteracji

Liczba iteracji	Średnia liczba pasujących par	Średni czas[ms]
50	257,5	<1
100	301,2	1
200	307,1	2
500	332,7	5,1
1000	325,75	10,7



Rysunek 15: Przykładowy przebieg metody RANSAC.

Wnioski Liczba iteracji jest istotnym parametrem metody RANSAC, ponieważ bezpośrednio wpływa na jakość znajdowanych modeli jak i czas działania. Jak widać w tabeli 4 jak i na wykresie 15 istnieje pewna graniczna wartość liczby iteracji, od której zmiany wielkości są bardzo małe lub nie występują wcale. Liczbę iteracji można przez to bardzo łatwo przeszacować, powodując tym samym, że metoda będzie działać dłucho.

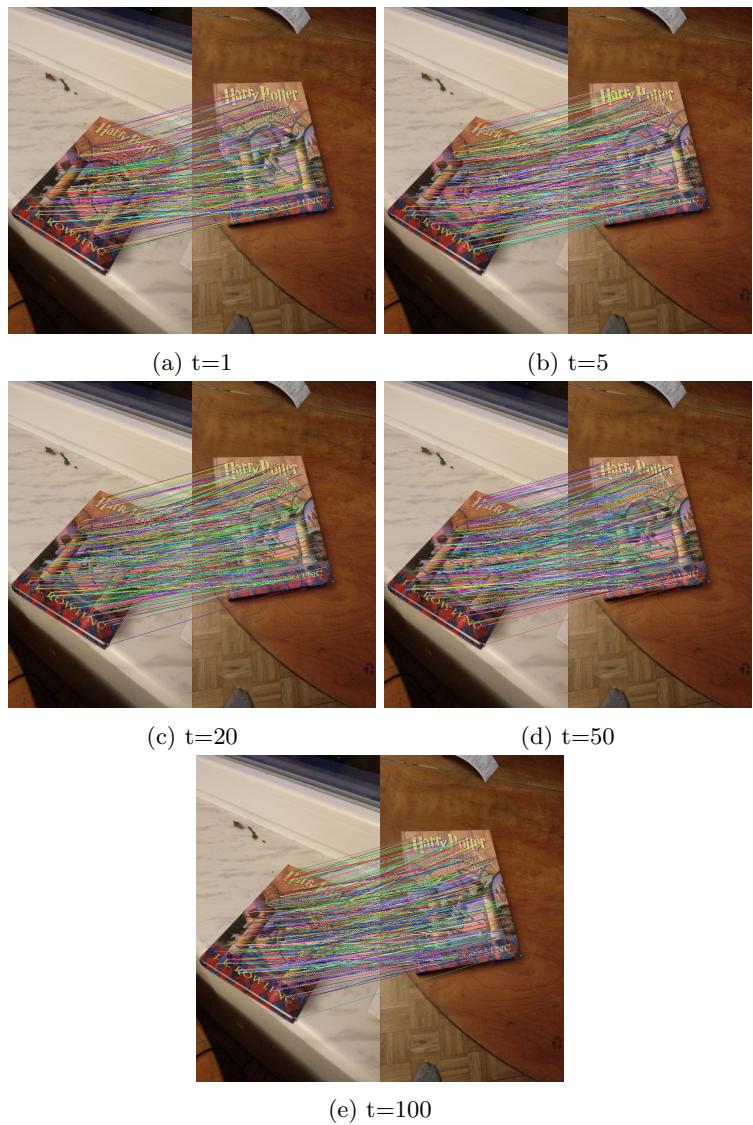
2.3.3 Wpływ wartości progu spójności

Badanie zostało przeprowadzone na tym samym obrazie co w poprzedniej sekcji. Ustalone wartości parametrów to:

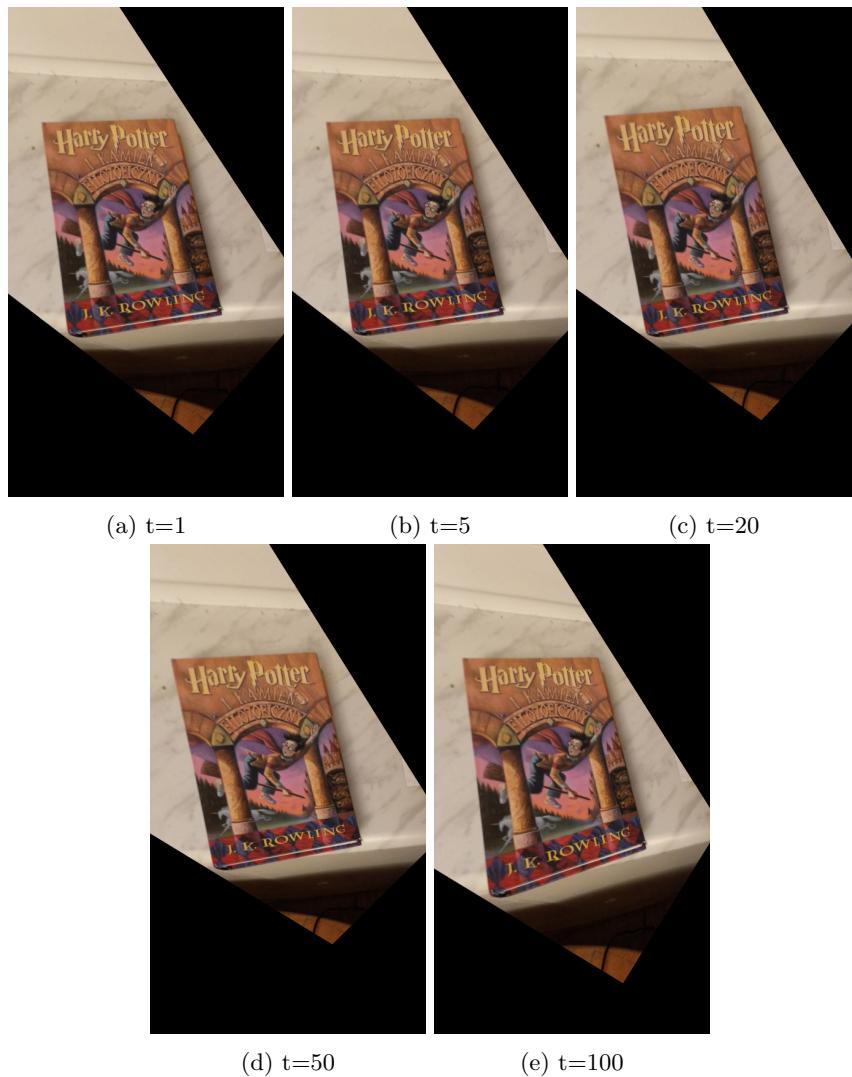
- liczba iteracji - 500,
- typ transformacji - perspektywiczna.

Tabela 5: Liczba i jakość punktów pasujących do transformacji znalezionej przez metodę RANSAC dla różnych wartości progu akceptacji

Próg akceptacji	Średnia liczba par	Jakość znalezionych par
1	342,5	Bardzo dobra
5	576,3	Dobra
20	583,2	Średnia
50	592,7	Niska
100	592,7	Niska



Rysunek 16: Pary punktów kluczowych pasujące do transformacji znalezionej przy różnych wartościach progu akceptacji.



Rysunek 17: Pierwszy obraz z pary po zaaplikowaniu najlepszej znalezionej transformacji przy różnych wartościach progu akceptacji.

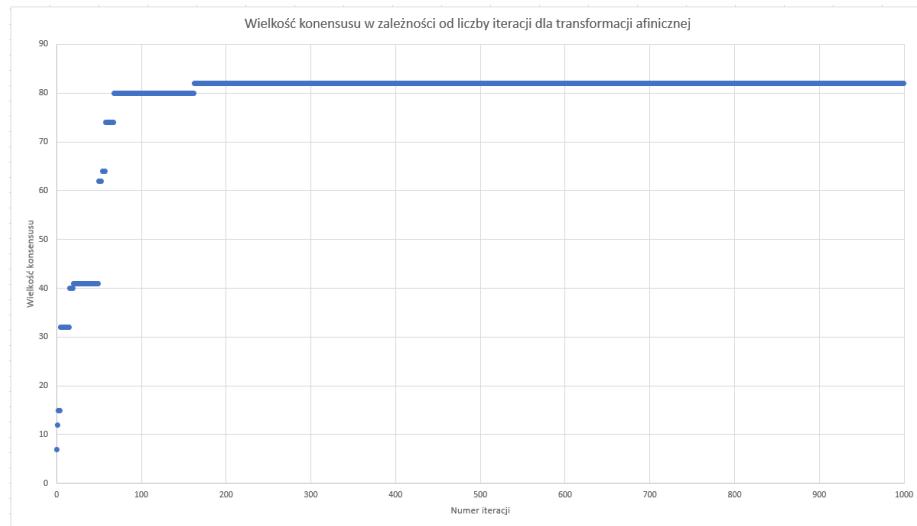
Wnioski Wartość parametru progu akceptacji danych ma istotny wpływ na jakość transformacji znajdowanych przez metodę RANSAC. Wyższe wartości tego parametru zacierają różnice pomiędzy dobrymi i słabymi modelami.

2.4 Heurystyki przyspieszające metodę RANSAC

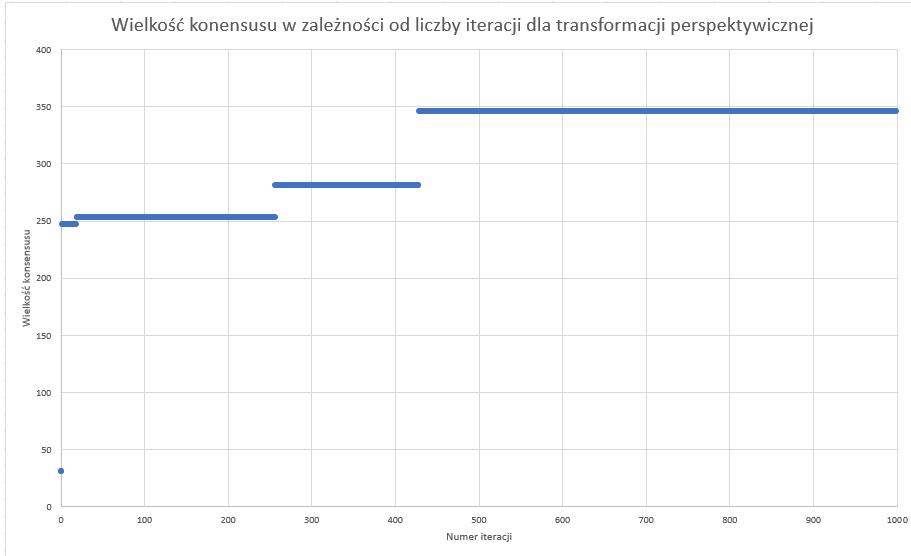
2.4.1 Heurystyka estymacji liczby iteracji

Patrząc na rysunek 15 można zauważyc, że po pewnej liczbie iteracji jakość modelu przestaje się poprawiać. Można więc, zamiast wybierać liczbę iteracji ręcznie, spróbować ją oszacować. Badania zostały przeprowadzone na zdjęciach z rysunku 1 przy następujących parametrach:

- próg spójności sąsiedztwa - 0.8,
- wielkość sąsiedztwa - 150.



Rysunek 18: Przebieg metody RANSAC dla 1000 iteracji i transformacji afanicznej.



Rysunek 19: Przebieg metody RANSAC dla 1000 iteracji i transformacji perspektywicznej.

Tabela 6: Szacowane liczby iteracji w zależności od zadanego prawdopodobieństwa sukcesu dla transformacji afinicznej i perspektywicznej.

P	Transformacja afiniczna	Transformacja perspektywiczna
0.5	32	119
0.6	43	158
0.7	57	208
0.8	72	279
0.9	108	399
0.99	217	800

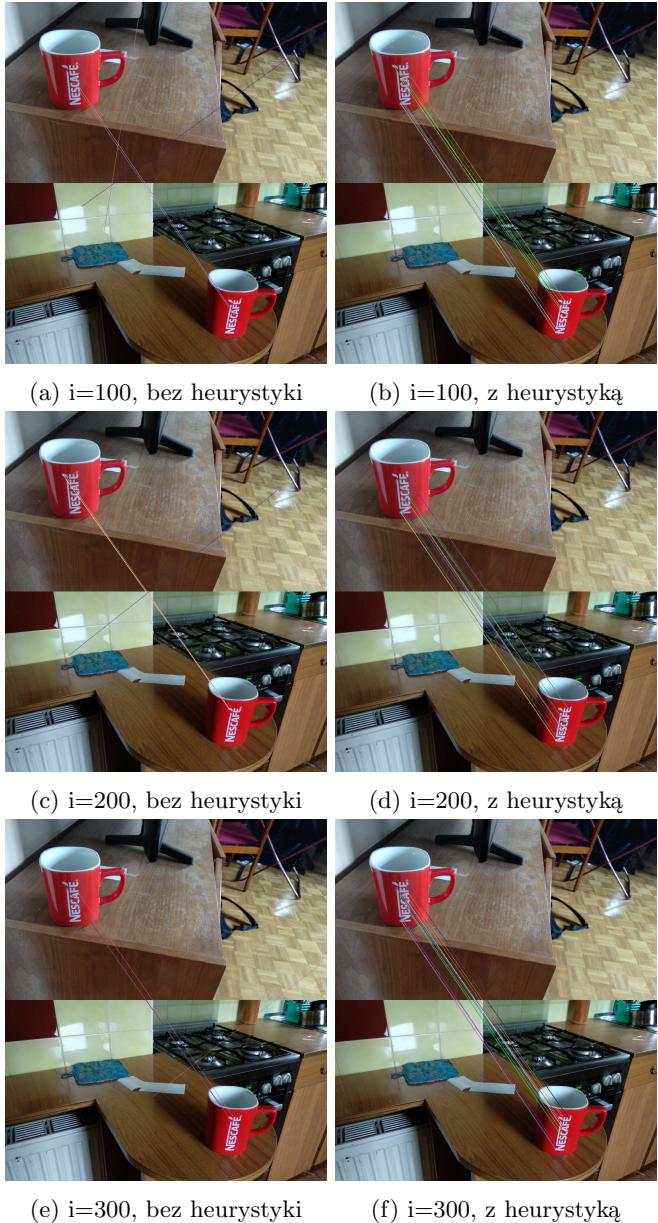
Wnioski Jak widać z rysunków 18 i 19 oraz tabeli 6, gdy zadana wartość prawdopodobieństwa zbliża się do 1, wtedy oszacowana liczba iteracji zaczyna odpowiadać faktycznej liczbie iteracji wymaganej do osiągnięcia dobrego modelu. Należy jednak pamiętać, że aby użyć tej heurystyki należy również oszacować prawdopodobieństwo wystąpienia szumu na obrazie. Prowadzi to do konieczności użycia innej metody i ustalenia dobrych parametrów dla niej.

2.4.2 Heurystyka odległości

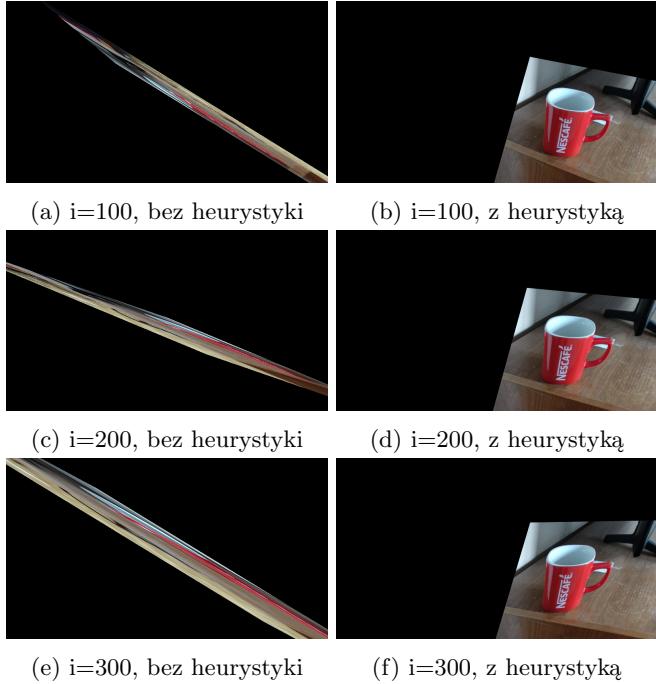
Badanie tej heurystyki zostało przeprowadzone na zdjęciach z rysunku 4. Wartości użytych parametrów to:

- rodzaj transformacji - perspektywiczna,

- próg akceptacji - 1.



Rysunek 20: Porównanie pasujących par punktów znalezionych bez heurystyki i z zastosowaną heurystyką odległości.



Rysunek 21: Porównanie najlepszych transformacji znalezionych bez heurystyki i z zastosowaną heurystyką odległości.

Wnioski Jak widać na rysunkach 20 i 21 heurystyka odległości działa bardzo dobrze i pozwala znajdować jakościowo dobre przekształcenia w dużo mniejszej liczbie iteracji. Wpływ heurystyki na **czas przetwarzania** jest rzędu 1ms i jest pomijalny.

2.4.3 Heurystyka modyfikacji rozkładu

Badanie tej heurystyki zostało przeprowadzone przeprowadzone na zdjęciach z rysunku 4.

Tabela 7: Średnia (10 przebiegów) liczba par znajdowanych na obrazie bez i z zastosowaną heurystyką modyfikacji rozkładu

Liczba iteracji	Bez heurystyki	Z heurystyką
100	6,5	6,9
500	10,3	11,1
1000	13,2	13,2
2000	12,9	15,2

Wnioski Jak widać w tabeli 7 heurystyka modyfikacji rozkładu nie ma tak dużego wpływu na liczbę par znajdowanych przez metodę RANSAC jak wcześniej opisane heurystyki.

3 Przykładowe wyniki

W tej sekcji zostaną zaprezentowane przykładowe wyniki otrzymane przez zastosowanie opisanych powyżej metod do wybranych par obrazów. We wszystkich przypadkach metoda RANSAC została zastosowana bez użycia heurystyk, z transformacją perspektywiczną i progiem akceptacji = 1. Parametry analizy spójności były ustalane indywidualnie dla poszczególnych obrazów.

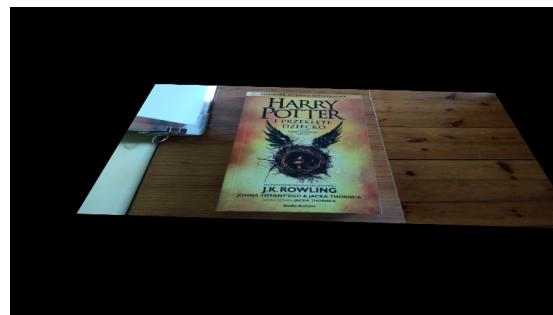


(a) Wszystkie pary

(b) Po analizie spójności

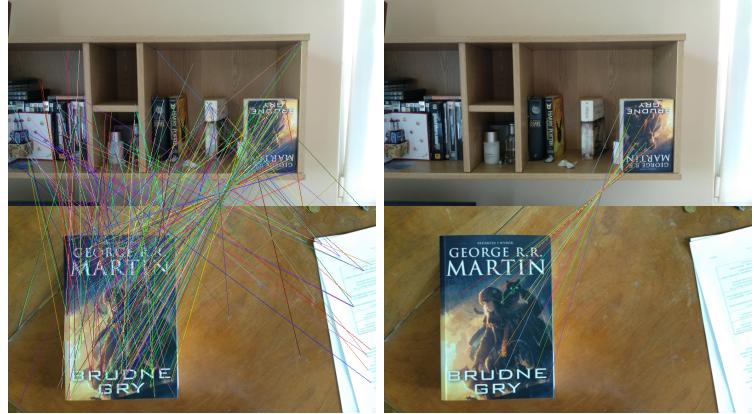


(c) Po zastosowaniu metody RAN-SAC



(d) Znaleziona transformacja

Rysunek 22: Wyniki zastosowania metod do zdjęć książki "Harry Potter i Prze-
kłecone Dziecko".



(a) Wszystkie pary

(b) Po analizie spójności



(c) Po zastosowaniu metody RAN-SAC



(d) Znaleziona transformacja

Rysunek 23: Wyniki zastosowania metod do zdjęć książki "Brudne gry".

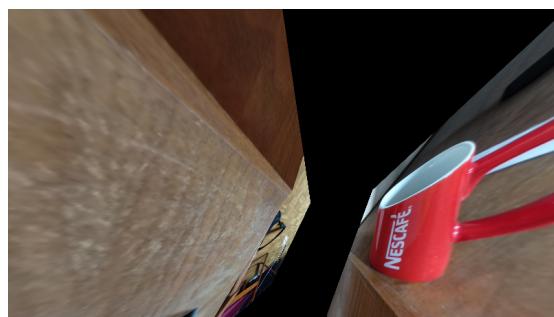


(a) Wszystkie pary

(b) Po analizie spójności

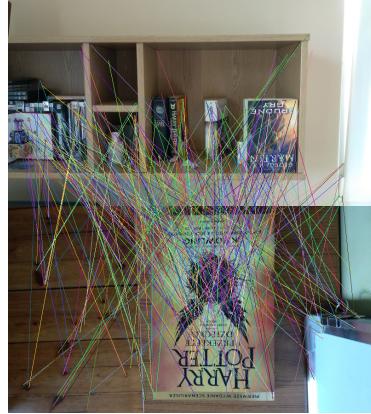


(c) Po zastosowaniu metody RAN-SAC

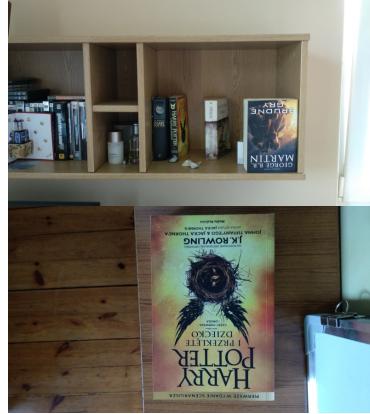


(d) Znaleziona transformacja

Rysunek 24: Wyniki zastosowania metod do zdjęć kubka



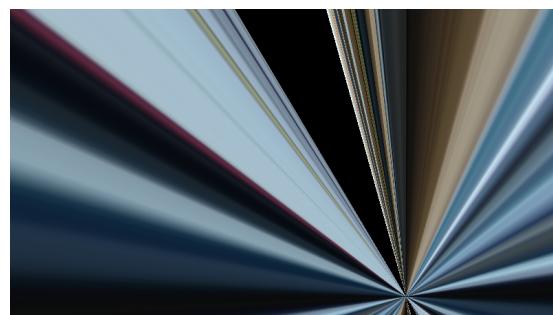
(a) Wszystkie pary



(b) Po analizie spójności



(c) Po zastosowaniu metody RAN-SAC



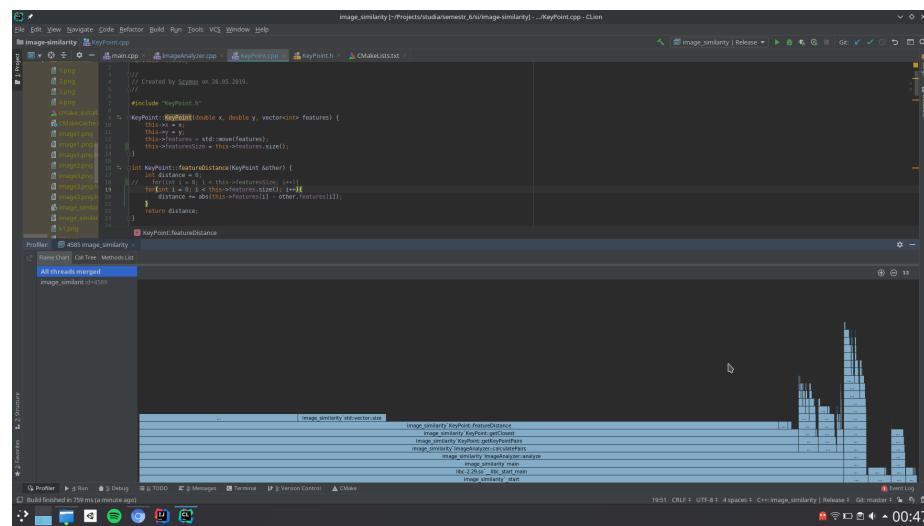
(d) Znaleziona transformacja

Rysunek 25: Wyniki zastosowania metod do zdjęć bez podobnych obiektów.

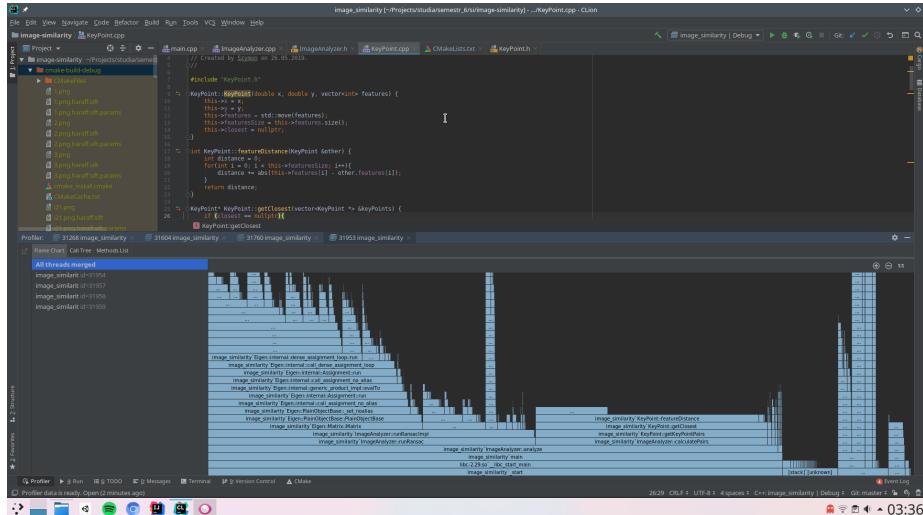
4 Podsumowanie

Zarówno algorytm analizy spójności sąsiedztwa jak i zastosowanie metody RANSAC do badania podobieństwa obrazów wydaje się być sensowne. Z przeprowadzonych badań wynika jednak, że najprawdopodobniej najlepszym wyjściem byłoby zastosowanie metody RANSAC wraz z heurystyką estymacji liczby iteracji oraz heurystyką odległości, ze względu na dobre wyniki i stosunkowo niewielką liczbę parametrów do strojenia.

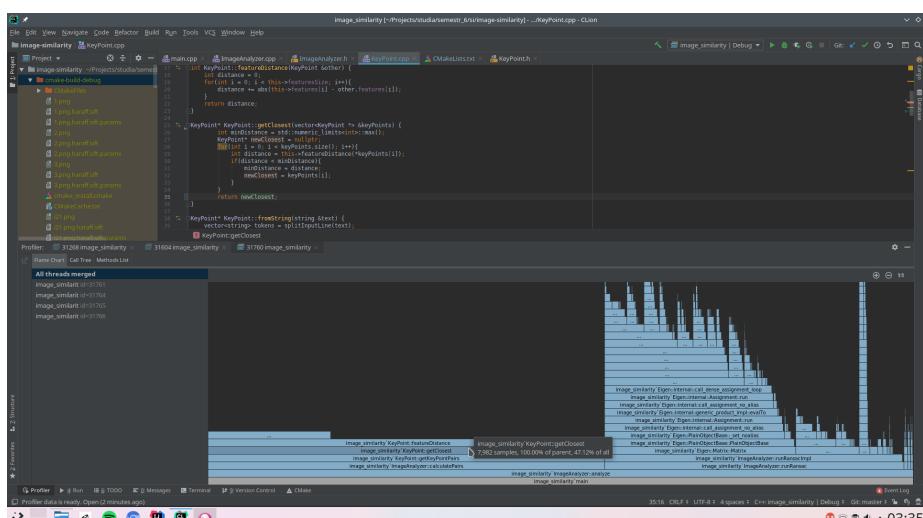
5 Profilowanie kodu



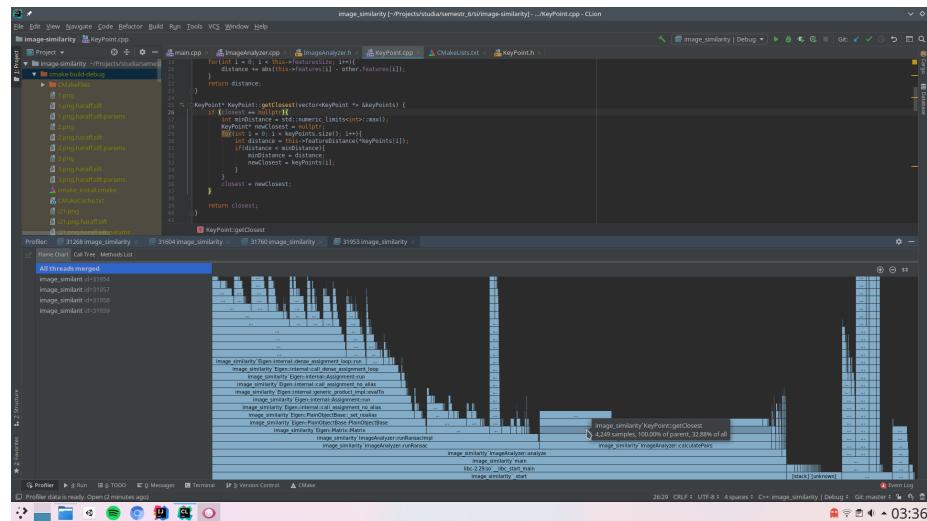
Rysunek 26: Metoda featureDistance przed profilowaniem kodu



Rysunek 27: Metoda featureDistance po profilowaniu kodu



Rysunek 28: Metoda getClosest przed profilowaniem kodu



Rysunek 29: Metoda getClosest po profilowaniu kodu