

Michał BUGAŁA

WYKRYWANIE ORIENTACJI OBIEKTÓW Z WYKORZYSTANIEM BIBLIOTEKI OPENCV

Streszczenie. W artykule przedstawiono podstawowe metody wykrywania obiektów na obrazie z kamer cyfrowych oraz próby wykorzystania metod do zbadania orientacji obiektu.

Słowa kluczowe: rozszerzona rzeczywistość, biblioteka OpenCV, analiza obrazu, przetwarzanie obrazu, grafika trójwymiarowa.

1. WPROWADZENIE

OpenCV (ang. *Open Computer Vision*) to biblioteka funkcji wykorzystywanych podczas obróbki obrazu, oparta na otwartym kodzie i zapoczątkowana przez korporację Intel. Zawiera około pięciuset funkcji, które obejmują takie dziedziny, jak: obrazowanie medyczne, robotyka, kalibracja kamery, interfejs użytkownika, kontrola jakości. Ze względu na częste wykorzystywanie systemów uczących się¹ w specjalistycznej obróbce obrazu, biblioteka OpenCV zawiera podbibliotekę MML (ang. *Machine Learning Library*) nastawioną na statystyczne rozpoznawanie wzorców i klastrów² [1].

Wykrycie obiektu oraz zbadanie jego orientacji w przestrzeni 3D rozpoczęto od zapoznania się z podstawowymi metodami porównującymi obrazy. Szybka i dokładna metoda porównawcza obrazów (dwuwymiarowych) ma kluczowe znaczenie dla dalszych obliczeń i przekształceń wyników do interesującej nas przestrzeni trójwymiarowej.

Na rysunku 1 przedstawiono obiekt (pudełko) znany a priori, którego zobrazowania pod różnymi kątami posłużyły jako szablony porównawcze z rzeczywistym obrazem z kamery (rysunek 2) i analizie wybranych metod.



Rys. 1. Przykładowe szablony porównawcze (wirtualny obiekt w środowisku 3D zobrazowany pod różnymi kątami)

1 Uczenie maszynowe – dziedzina wchodząca w skład nauk zajmujących się problematyką sztucznej inteligencji [2].

2 Klaster - zbiór elementów o podobnych cechach.

2. METODY

Zbadano i porównano podstawowe metody zawarte w bibliotece OpenCV w celu wyodrębnienia tej najbardziej właściwej, tj. szybkiej i dokładnej. Metody poddano eksperymentowi, który polegał na porównaniu około 1200 różnych szablonów (obiekt wirtualny zobrazowany pod różnymi kątami) z obrazem z kamery i znalezieniu najlepszego dopasowania, a tym samym przybliżonej orientacji obiektu rzeczywistego.

2.1. Dopasowywanie szablonów



Rys. 2. Przykładowy obraz z kamery z naniesionym obwodem szablonu porównawczego

Dopasowywanie szablonów odbywa się za pomocą funkcji *cvMatchTemplate* znajdującej się w bibliotece OpenCV. Zaletą metody dopasowywania szablonów jest możliwość otrzymania informacji o pozycji 2D szablonu na obrazie z kamery oraz współczynnika dopasowania, który w tym przypadku dąży do wartości 1 (idealne dopasowanie). Wadą tej metody jest długi czas obliczeń.

Krótkie podsumowanie metody:

- plusy:
 - funkcja zwraca współczynnik dopasowania,
 - możliwość pobrania informacji o pozycji 2D szablonu na obrazie z kamery,
 - metoda dokładna,
- minusy:
 - długi czas obliczeń dla jednego dopasowania (≤ 62 [ms] dla założonych wielkości obrazów: 1280×720 [px] (kamera), 614×563 [px] (szablon)).

2.2. Porównywanie kształtów

Porównywanie kształtów odbywa się za pomocą funkcji *cvMatchShapes* i wcześniej przygotowanych obrazów (rysunek 3). Funkcja *cvMatchShapes* wykorzystuje Momenty HU [3] i nie przyniosła zadowalających rezultatów, jeśli chodzi o dokładność wyników. Należałoby również dołożyć wielu starań, aby odseparować badany obiekt od tła i porównywać obrazy tej samej wielkości³.

³ Funkcja porównuje kształty, nie dopasowuje kształtów.



Rys. 3. Obraz z kamery z wyodrębnionym kształtem

Krótkie podsumowanie metody:

- plusy:
 - funkcja zwraca współczynnik porównania,
 - krótki czas obliczeń (≤ 16 [ms] dla wielkości obrazów: 1280×720 [px] i 1280×720 [px]),
- minusy:
 - metoda mało dokładna i wymaga odpowiedniego przygotowania obrazów (nadaje się do analizowania mniej skomplikowanych kształtów).

2.3. Porównywanie konturów

Porównywanie konturów odbywa się za pomocą funkcji *cvMatchShapes* i użytej wcześniej funkcji *cvFindContours* wyznaczającej kontury. Podobnie jak w przypadku porównywania kształtów, również i ta metoda nie okazała się zbyt dokładna, pomimo względnie poprawnie przygotowanych obrazów (rys. 4).



Rys. 4. Porównanie konturów szablonu (po lewej) i obrazu z kamery

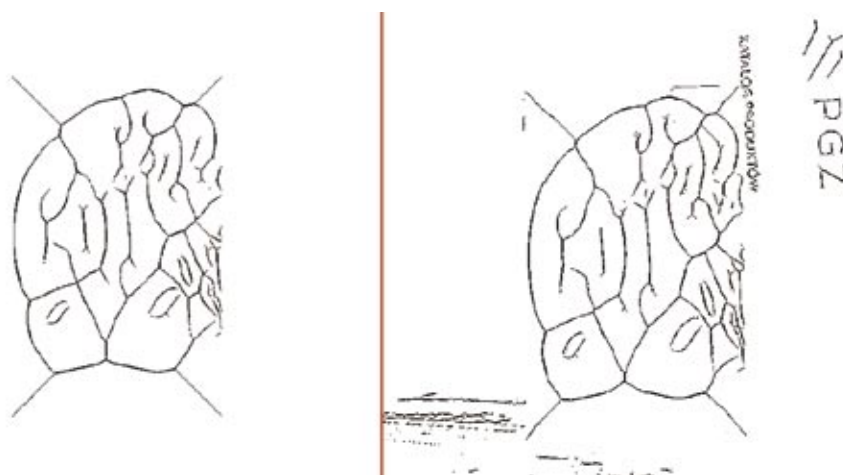
Krótkie podsumowanie metody:

- plusy:
 - funkcja zwraca współczynnik porównania,
 - krótki czas obliczeń (≤ 16 [ms] dla obrazów: 1280x720 [px] i 1280x720 [px]),
- minusy:
 - metoda mało dokładna i wymaga odpowiedniego przygotowania analizowanych obrazów.

2.4. Porównywanie szkieletów

Porównywanie szkieletów (rys. 5) to propozycja przygotowywania obrazów do analizy i może być punktem wyjściowym dla nierozpatrywanej tutaj metody fazowej (tłumaczenie własne, z ang. *chamfer matching*) [4], w której dopasowuje się szablon - kontury lub szkielet w tym przypadku) do obrazu przetworzonego algorytmem mapy odległości (tłumaczenie własne, z ang. *distance transform* lub *distance map*) [5] – rysunek 6.

Porównywanie szkieletu oparto na metodzie dopasowywania szablonów *cvMatchTemplate* i porównywania konturów *cvMatchShapes*, jednak nie zauważono poprawy wyników po szkieletyzacji⁴.

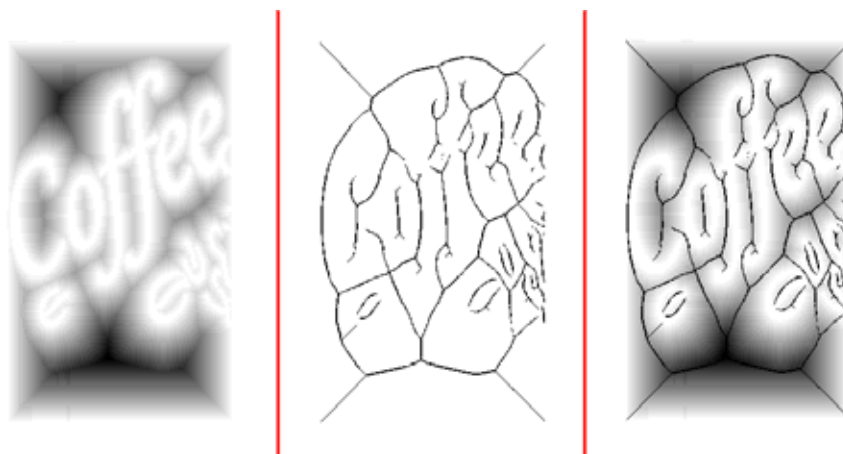


Rys. 5. Porównanie szkieletu szablonu (po lewej) z obrazem z kamery

Krótkie podsumowanie metody:

- plusy:
 - funkcja zwraca współczynnik porównania,
 - czas obliczeń zależny od porównywania konturów,
- minusy:
 - czas obliczeń zależny od wykorzystanej metody dopasowywania szablonów,
 - metoda mało dokładna (również w przypadku wykorzystania funkcji *cvMatchTemplate*) i wymaga odpowiedniego przygotowania analizowanych obrazów (szkieletyzacja).

⁴ Szkieletyzacja jest operacją wyznaczania liniowej reprezentacji figur na analizowanym obrazie [6].



Rys. 6. Koncepcja metody fazowej (z ang. *chamfer matching*) z wykorzystaniem mapy odległości obrazu z kamery (po lewej) i szkieletu szablonu (w środku)

2.5. Porównywanie metodą SURF

Metodę SURF (ang. *Speeded-Up Robust Features*) można zastosować dzięki funkcji *cvExtractSURF* znajdującej się w bibliotece OpenCV. Metoda polega na dopasowywaniu wielkości cech (tłumaczenie własne, z ang. *robust features*) szablonu do wielkości cech obrazu z kamery [7] - rysunek 7.



Rys. 7. Wizualizacja metody SURF

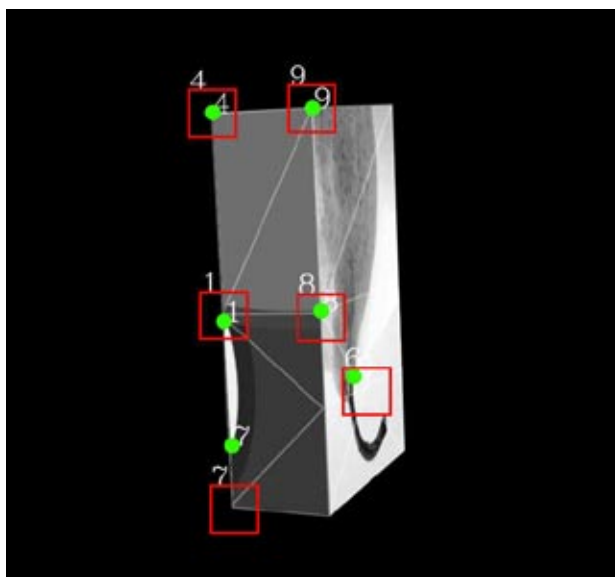
Krótkie podsumowanie metody:

- plusy:
 - funkcja zwraca współczynnik porównania,
 - akceptowalny czas obliczeń (≤ 30 [ms] dla obrazów: 307×281 [px] i 307×281 [px]),
 - możliwość pobrania informacji o pozycji 2D szablonu na obrazie z kamery,
- minusy:
 - brak bezpośredniej możliwości obliczenia pozycji i orientacji w przestrzeni 3D (problem PnP⁵ dla punktów leżących na jednej płaszczyźnie – biblioteka OpenCV w wersji 2.4.11 nie zawiera gotowego rozwiązania tego problemu),
 - metoda względnie dokładna (duża fluktuacja wyników dla powtórzonych porównań).

2.6. Dopasowywanie par charakterystycznych

Porównywanie obrazów metodą dopasowywania par charakterystycznych jest propozycją odnajdowania orientacji obiektów zarejestrowanych obrazów. Metoda polega na dopasowaniu wycinków szablonu o najbardziej charakterystycznych cechach do obrazu z kamery. Cały szablon analizowany jest za pomocą funkcji *cvGoodFeaturesToTrack*, znajdującej się w bibliotece OpenCV, która odnajduje cechy charakterystyczne (najlepsze miejsca do śledzenia) zobrazowanego obiektu 3D (szablonu).

Punkty charakterystyczne (cechy) widoczne są na rysunku 8, przy czym wycinki pobierane są z najbliższego wierzchołka obiektu 3D⁶ – w dalszym etapie umożliwi to łatwe obliczenie orientacji obiektu.

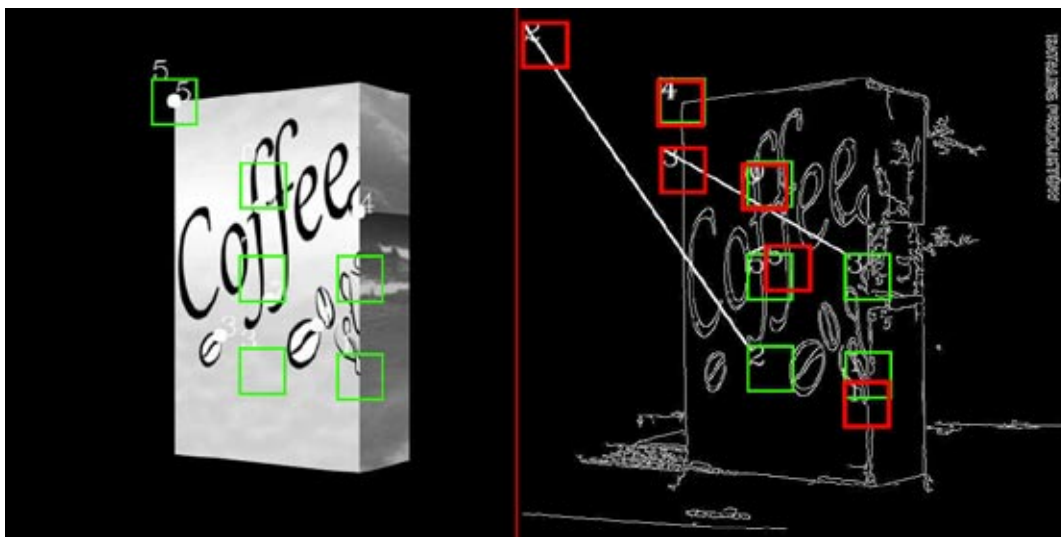


Rys. 8. Szablon z naniesionymi obwodami wycinków, siatką obiektu 3D i punktami charakterystycznymi

-
- 5 Problem PnP (ang. *Perspective-n-Points Problem*) – zagadnienia dotyczą problematyki obliczeń pozycji i orientacji obiektów na podstawie zależności pomiędzy zbiorem punktów 3D lub 2D i odpowiadającym im rzutom 2D na obrazie [8].
 - 6 Rozwinięciem metody dopasowywania par charakterystycznych może być możliwość dodania wierzchołka do obiektu 3D w punkcie najbardziej charakterystycznym, a nie pobieranie najbliższego wierzchołka.

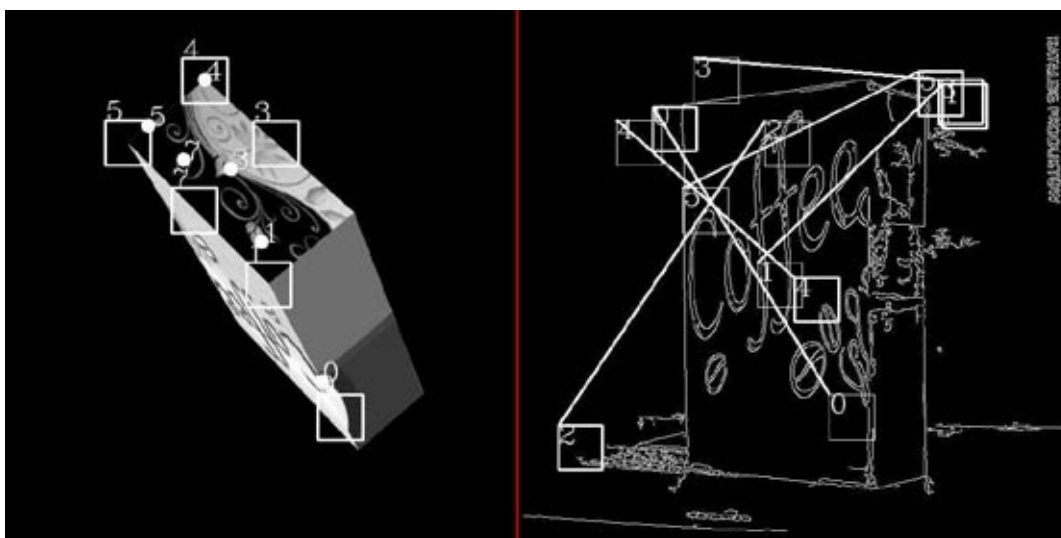
W celu przyspieszenia działania metody, wycinki szablonu, informacje na temat wierzchołków obiektu 3D i położenia 2D wycinków zapisywane są na twardym dysku i nie ma potrzeby powtórnej analizy obiektu. Dodatkowo wycinki szablonu zostały zmniejszone do rozmiaru 16x16 [px] i przetworzone algorytmem detekcji krawędzi *cvCanny* (biblioteka OpenCV). Wielkość wycinków można dowolnie modyfikować i tym samym kontrolować stosunek jakości do szybkości działania metody.

Po odpowiednim przygotowaniu danych, algorytm sprowadza się do standardowego procesu dopasowywania wycinków szablonu do obrazu za pomocą funkcji *cvMatchTemplate* (rys. 9), usunięciu błędnych dopasowań i wybraniu najlepszych par dla aktualnego szablonu.



Rys. 9. Wizualizacja dopasowania wycinków szablonu (po lewej) do obrazu z kamery (numeracja wycinków pomiędzy lewym a prawym obrazem nie jest wiążąca)

Rysunek 10 prezentuje zachowanie się algorytmu w przypadku niepasującego szablonu.



Rys. 10. Wizualizacja dopasowania wycinków w przypadku niepasującego szablonu (widoczne duże przesunięcia par na obrazie po prawej stronie)

Krótkie podsumowanie metody:

- plusy:
 - funkcja zwraca współczynnik dopasowania,
 - możliwość pobrania informacji o pozycji 2D szablonu na obrazie z kamery,
 - możliwość obliczenia pozycji i orientacji w przestrzeni 3D (funkcja *cvPOSIT*⁷),
 - metoda dokładna,
- minusy:
 - krótki czas obliczeń (≤ 16 [ms] dla sześciu wycinków o rozmiarze 16x16 [px] i obrazu z kamery 307x281 [px]),
 - konieczność utworzenia tablicy z danymi (jednak nie ma to wpływu na szybkość działania samej metody).

3. WNIOSKI

Metody porównywania kształtów i konturów są metodami szybkimi, ale mało precyzyjnymi w przypadku obrazów o dużej złożoności. Metoda SURF i zaproponowana metoda dopasowywania par charakterystycznych wyróżniają się akceptowalnym czasem wykonywania i dużą dokładnością. Najbardziej czasochłonną okazała się pierwsza analizowana metoda – podstawowa metoda dopasowywania szablonów.

Zbadane metody dopasowywania i porównywania obrazów, zawarte w bibliotece OpenCV, z różnym skutkiem umożliwiają odnajdowanie orientacji obiektów zarejestrowanych na obrazie, jednak żadne nie mogą być używane w czasie rzeczywistym (w zaprezentowanej formie). Analiza setki szablonów (obiekt wirtualny zobrazowany pod różnymi kątami) nie jest możliwa do przeprowadzenia w czasie krótszym niż 40 [ms] (40 [ms] = 25 [fps]), który zapewniłby płynność odtwarzanego obrazu z kamery przy jednoczesnym obliczaniu orientacji obiektu.

Metoda dopasowywania par charakterystycznych również nie może być użyta, jednak może być przydatna w pierwszym etapie śledzenia obiektu jako wyznacznik pozycji startowej, tzn. umożliwi odnalezienie orientacji obiektu 3D oraz przyporządkuje wierzchołki obiektu (punkty 3D) do punktów 2D na obrazie z kamery (umożliwia to bezpośrednie obliczenie pozycji i orientacji obiektu z wykorzystaniem funkcji *cvPOSIT* znajdującej się w bibliotece OpenCV). W dalszym etapie realizacji metody wykrywania pozycji i orientacji obiektów, którą będzie można zastosować w czasie rzeczywistym (z uwzględnieniem skali obiektu i perspektywy obrazu), należy wykorzystać metody z zakresu technik śledzenia ruchu oraz bardziej zaawansowane funkcje zawarte w podbibliotece MML.

Analiza obrazu i rzeczywistość rozszerzona to dziedziny informatyki o bardzo dużym potencjale rozwojowym i możliwościach zastosowania w przemyśle wojskowym oraz cywilnym. Zakres zagadnień związanych z rzeczywistością rozszerzoną, jakie można zgłębiać i analizować, jest wciąż olbrzymi i czeka na nowe, błyskotliwe rozwiązania.

7 Funkcja *cvPOSIT* - funkcja zwraca informacje o położeniu i orientacji obiektu, jeśli znane są położenia wierzchołków (punkty 3D) obiektu wirtualnego i odpowiadające im punkty 2D na obrazie z kamery (problem PnP dla punktów nieznajdujących się na jednej płaszczyźnie) [1].

4. LITERATURA

- [1] Bradski G., Kaehler A.: Learning OpenCV, O'Reilly Media, 2008, ISBN: 978-0-596-51613-0.
- [2] Mitchell M. T.: Machine Learning, 1997, McGraw-Hill Science, ISBN: 978-0070428072.
- [3] Hu M. K.: Visual Pattern Recognition by Moment Invariants, <http://www.sci.utah.edu/~gerig/CS7960-S2010/handouts/Hu.pdf> [dostęp: 15.08.2016].
- [4] Mitsubishi Electric Research Laboratories: Fast Directional Chamfer Matching, <http://www.merl.com/publications/docs/TR2010-045.pdf> [dostęp: 04.04.2016].
- [5] Mitsubishi Electric Research Laboratories: Adaptively Sampled Distance Fields: A General Representation of Shape for Computer Graphics, <http://www.merl.com/publications/docs/TR2000-15.pdf> [dostęp: 15.08.2016].
- [6] Tadeusiewicz R, Korohoda P.: Komputerowa analiza i przetwarzanie obrazów, 1997, Wydawnictwo Fundacji Postępu Telekomunikacji , ISBN: 9788386476152.
- [7] OpenCV 2.2 documentation: Feature detection and description, http://opencv.jp/opencv-svn_org/c/features2d_feature_detection_and_description.html [dostęp: 04.04.2016].
- [8] Lepetit V., Moreno-Noguer F., and Fua P.: EPnP: Efficient Perspective-n-Point Camera Pose Estimation, <https://infoscience.epfl.ch/record/160138/files/top.pdf?version=1>, [dostęp: 15.08.2016].

OBJECT ORIENTATION RECOGNITION USING THE OPENCV LIBRARY

Abstract. The article presents the basic methods of detecting objects in an image from a digital camera and attempts to apply methods of determining the orientation of an object.

Keywords: augmented reality, OpenCV library, image analysis, image processing, 3D graphics.