

Sprawozdanie z laboratorium: Komunikacja człowiek-komputer

Sprawozdanie I: Przetwarzanie obrazu – aplikacja

8 listopada 2012

Prowadzący: dr inż. Wojciech Jaśkowski

Autorzy:	Michał Nowicki	inf95883	alozzypl@gmail.com
	Olgierd Pilarczyk	inf100449	opilarczyk@gmail.com

Zajęcia piątkowe, 8:00 i 9.45.

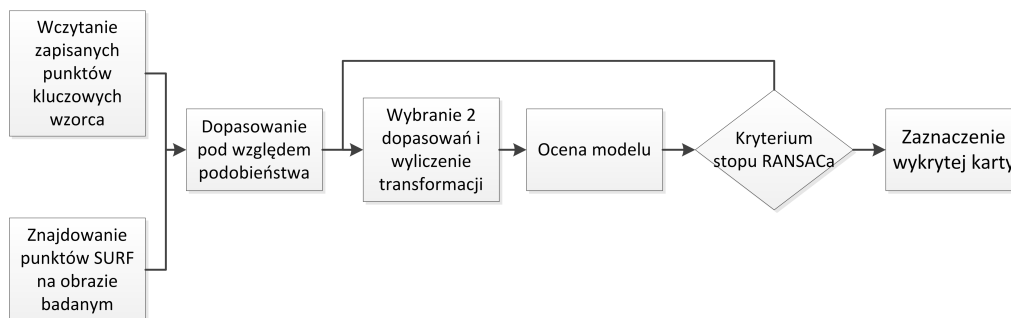
1 Wstęp

Celem projektu było zaimplementowanie i zbadanie metod rozpoznawanie karty do gry na obrazach. Postanowiliśmy podzielić się algorytmami do napisania i zbadania. Olgierd Pilarczyk zajmował się implementacją i analizą detektora Haara wspartego momentami Hu. Michał Nowicki implementacją i analizą metody opartej o SURF i ORB.

2 Metoda wykrywania kart oparta o metodę SURF

Metoda SURF (ang. Speeded Up Robust Features)[2] jest metodą pozwalającą na znajdowanie punktów charakterystycznych pod względem cech fotometrycznych na obrazach 2D. Każdy punkt kluczowy jest dodatkowo opisywany za pomocą wektora cech określających (deskryptora) charakterystykę danego punktu kluczowego.

Metoda polega na wcześniejszym znalezieniu punktów kluczowych i ich opisu dla wzorca oraz zapisaniu tych wartości np. w pliku. Następnie dla każdego badanego obrazu wyliczane są punkty SURF. Kolejnym krokiem algorytmu jest dopasowanie odpowiednich punktów kluczowych obu obrazów do siebie.



Rysunek 1: Schemat blokowy przetwarzania

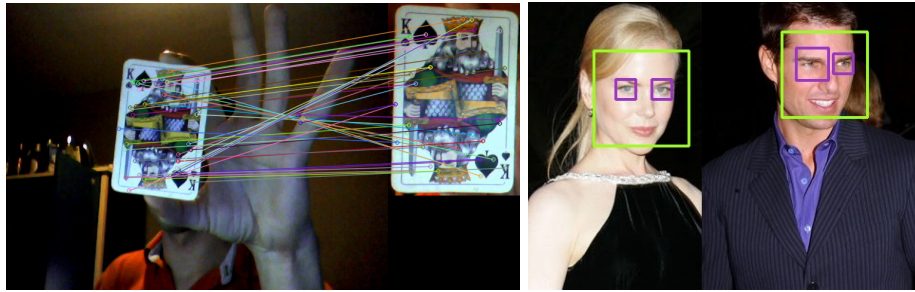
Z potencjalnej liczby dopasowań należy znaleźć te poprawne. W ramach projektu postanowiono, że poprawne dopasowanie punktów zostanie znalezione za pomocą metodyki RANSAC (ang. RANdom SAMple Consensus). Założono, że karty badane będą bez zniszczeń wynikających z projekcji kart. Dzięki takim ograniczeniom do poprawnego wyznaczenia przekształcenia potrzeba jedynie 2 par dopasowań, a parametry można wyliczyć analitycznie.

Zastosowana metodyka RANSAC opiera się na losowym wybieraniu potencjalnych dopasowań i tworzeniu modelu na ich podstawie. Po wyznaczeniu modelu badana jest liczba punktów kluczowych, dla których transformacja przekształca punkty kluczowego z pierwszego obrazu w punkt kluczowy z obrazu drugiego z ustaloną dokładnością (w programie 0.1). Gdy liczba spełnionych dopasowań dla danej transformacji przekroczy 2, przekształcenie uznawane jest za poprawne, a obiekt za znaleziony.

W projekcie zastosowano podejście probabilistyczne do wyznaczenia liczby iteracji algorytmu przy zadanym poziomie prawdopodobieństwa sukcesu (w projekcie 0.99).

3 Metoda wykrywania kart oparta o detektor Haara

Metoda wymaga uprzedniego wygenerowania wzorca (ang. Haar cascade). Służy do tego wbudowana w pakiet OpenCV aplikacja *opencv_haartraining*. Korzysta ona z zestawu po-



Rysunek 2: Obraz badany i wzorec z zaznaczonymi punktami SURF oraz ich dopasowaniami oraz detektor Haara zastosowany do rozpoznawania twarzy i oczu

zytywów (obrazy poszukiwanego obiektu) oraz negatywów (obrazy niezawierające szukanego obiektu, najlepiej takie, które występować będą jako tło).

Jakość wygenerowanego wzorca zależy od liczby i jakości pozytywów i negatywów. Do poszukiwania złożonych i różnorodnych obiektów (np. ludzkie twarze) konieczne jest dostarczenie kilku tysięcy pozytywów, natomiast do rozpoznawania jednego i niezmiennego obiektu jak karta do gry, wystarczy jedno dobrej jakości zdjęcie. Z tego zdjęcia, aplikacja jest w stanie wygenerować dowolną ilość pozytywów, stosując przekształcenia geometryczne, którym będzie ulegał poszukiwany obiekt w warunkach rzeczywistych (rotacja, perspektywa).

Operacja generowania wzorca (kaskady) trwa kilka godzin i jest proporcjonalna do złożoności (ilości cech Haara) obiektu. (ok. 5h dla jednego zdjęcia karty, na komputerze wyposażonym w procesor Intel Core i5 2,5GHz)

Domyślnie generowane wzorce opierają się na pozytywach przeskalowanych do formatu 24x24 piksele, przez co część informacji jest gubiona. W działaniu programu zaowocowało to oznaczaniem przez metodę Haara oprócz karty, także innych obszarów (ang. false positives).

Aby zawęzić wynik, dla każdego obszaru znalezionej przez klasyfikator Haara, liczone są momenty H_u , a następnie porównywane ze wzorcem. Jako ostateczny rezultat wybierany jest obszar, dla którego suma odchyłek momentów od wzorca (pod uwagę brane są pierwsze 4 momenty H_u) podzielona przez rozmiar boku znalezionej obszaru, jest najmniejsza.

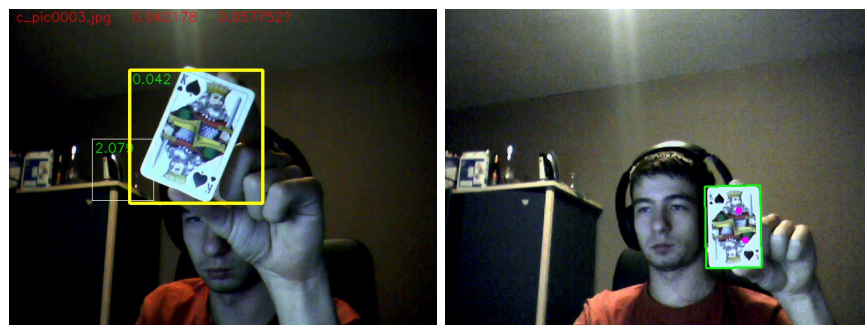
4 Eksperymenty

4.1 Założenia

Eksperymenty przeprowadzano na danych pochodzących z kamery laptopowej o rozdzielczości 640x480. Podana rozdzielczość obrazu została wybrana jako kompromis pomiędzy czasem przetwarzania, a jakością zdjęć. Eksperymenty przeprowadzano ze szczególnym uwzględnieniem czasu przetwarzania i skuteczności rozpoznawania. Starano się, aby krytycznie oceniać działania algorytmów. W przypadku, gdy karta została wykryta jedynie częściowo, przypadek uznawany był za błędny. Zależało nam na pełnej i obiektywnej ocenie algorytmów.

4.2 Seria zdjęć z dobrze widoczną kartą

Początkowo przygotowano zestaw 23 zdjęć, na których wzorec był dobrze widoczny i nie-rozmaźany. We wszystkich przypadkach człowiek potrafił poprawnie określić pozycję karty. Algorytm wykorzystujący mechanizm SURF wykrył poprawnie 14 z 23 zdjęć, co oznacza



Rysunek 3: Udana rozpoznania Haara, gdzie metody SURF i ORB zawiodły oraz pozytywne rozpoznania dla metody SURF

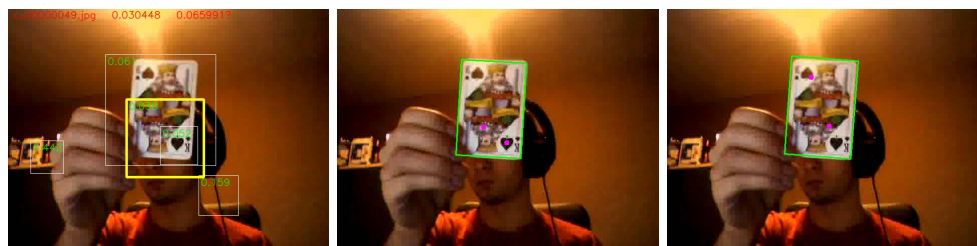
skuteczność na poziomie 61%. Duża różnica w jasności wzorca i obrazu badanego uniemożliwiała skuteczną detekcję. Dodatkowo algorytm działa gorzej, gdy karty znajdują się pod innym kątem niż we wzorcu.

Następnym krokiem eksperymentu było zastąpienie mechanizmu SURF za pomocą nowo opublikowanego mechanizmu ORB. Mechanizm podobnie do algorytmu SURF wykrywa punkty kluczowe i opisuje je za pomocą deskryptora BRIEF. W tym przypadku uzyskano 10 poprawnych detekcji na 23 zdjęcia, a więc skuteczność wynosi około 43,5 %. Zaletą mechanizmu ORB jest jednak jego szybkość. Jeden obraz był przetwarzany średnio 10 razy szybciej niż z użyciem metody SURF.

Metoda Haara połączona ze sprawdzaniem momentów Hu poprawnie rozpoznała kartę na 19 obrazach spośród 23 (83% skuteczności). Przykłady dla których nie udało się znaleźć wzorca to 1x karta obrócona o 90°, 2x karta pochylona górną w stronę kamery (perspektywa), 1x wybór złego obszaru przez porównanie momentów Hu.

4.3 Seria danych z filmu video

W celu zbadania skuteczności metod w realnej aplikacji nagrano film zawierający 418 klatek. Następnie dla każdej klatki przeprowadzono eksperyment z metodą opartą o klasyfikator Haara, o SURF i o ORB. Karta była przesuwana, obracana i zmieniano także kąt nachylenia karty do kamery. Zmianie ulegało także oświetlenie.



Rysunek 4: Błędne rozpoznania Haara i pozytywne rozpoznania karty przez odpowiednio SURF i ORB

Dla powyższej serii, algorytm SURF w 1 wersji pozytywnie rozpoznał kartę na 105 obrazach z 418, co oznacza skuteczność na poziomie 25 %. Większość z obrazów była jednak

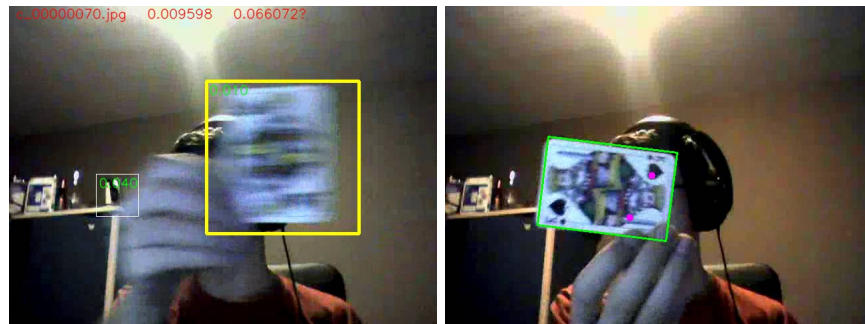
nieostra z powodu ruchu karty. W dużej liczbie z tych przypadków sami nie potrafiliśmy stwierdzić, czy na obrazie znajduje się badana karta. Prosty test, polegający na ręcznym wybraniu 1 wzorca z danych pobranych w takich samych warunkach, pozwolił na poprawne rozpoznanie karty na 188 zdjęciach. Drastyczna poprawa sugeruje potrzebę stworzenia pewnego generycznego modelu wzorca danego obiektu. Niestety z powodów czasowych prace w tym kierunku nie zostały ukończone.

Tabela 1: Porównanie wyników dla serii 418 obrazów

	Haar	SURF	SURF z wczytaniem	ORB
Czas dla 1 zdjęcia	0,05s	0,65s	0,58s	0,06s
Max FPS	20	1.53	1,72	16,7
Skuteczność	70%	45%	45%	2%

Zaskakujący rezultat uzyskano dla algorytmu opartego o mechanizm ORB. Okazał się on całkowicie nieodporny na zmiany oświetlenia i nieostrość badanej karty. Na 418 obrazów, tylko 7 zostało pozytywnie rozpoznanych. Wynik pokazuje, że zastosowany mechanizm ma mały potencjał, jeśli chodzi o rzeczywista aplikacje. Zaskakująca jest różnica w porównaniu z wynikami z pracy [1], w której w serii eksperymentów wykazano podobie parametry mechanizmów SURF i ORB.

Algorytm Haara prawidłowo odnalazł kartę na 292 obrazach, co daje skuteczność rzędu 70%. Rezultaty błędne w dużej części spowodowane były tym, że jeden spośród zwracanych przez metodę Haara obszarów miał odchylenie momentów H_u od wzorca niższe, niż ten który faktycznie zawierał kartę.



Rysunek 5: Pozytywne rozpoznanie karty przez detektor Haara oraz przez metodę SURF

4.4 Seria danych bez karty

W celu zmierzenia liczby błędnie wykrytych kart postanowiono zbadać działanie algorytmu na serii obrazów nie zawierających badanej karty. Liczba obrazów w serii wynosiła 15. Każde potencjalne wykrycie karty uznawane było za błędne. W tym eksperymencie zależało nam na zbadaniu jak wiarygodna może być potencjalna detekcja.

W przypadku algorytmu SURF liczba błędnie wykrytych kart wynosi 0. Dla algorytmu ORB jest to liczba 2. Algorytm Haara we wszystkich przypadkach znalazł błędne dopasowanie wzorca. Problem potencjalnie błędnych wykryć wynika przede wszystkim z zasady działania detektora i wybranego wzorca, który posiada mało cech charakterystycznych, gdy zostanie pomniejszony do rozmiaru 24x24 pixeli.



Rysunek 6: Błędna detekcja karty dla metody Haara

W przypadku, gdy poszukiwany obiekt miałby być analizowany z wieloma wzorcami lepsza wydaje się metoda oparta o mechanizm SURF. Powinien on znaleźć 1 pasujący wzorec. Algorytm oparty o detektor Haara przez dużą liczbę błędnych detekcji w czystej formie nie nadaje się do takich zastosowań.

5 Wnioski

Okazało się, że zaproponowana metoda SURF jest zdecydowanie wolniejsza od standardowego detektora Haara. Jeżeli zależy nam na detekcji online z częstotliwością równą około 20 FPS to jedynym rozwiązaniem jest stosowanie detektora Haara. ORB nie jest dobrą alternatywą. Własna implementacja pokazała, że dobrze nauczony detektor Haara jest rozwiązaniem optymalnym, gdy wiemy, że obiekt znajduje się na scenie. Warto zwrócić uwagę na fakt bardzo dobrego rozpoznawania nieostrej karty przez detektor Haara. W zależności od aplikacji, wspomniana cecha może być pożądana lub wpływać negatywnie na działanie całego systemu.

Kluczowym elementem dla uzyskania dobrych wyników metodą Haara jest wygenerowanie wysokiej jakości wzorca. Jest to zadanie czasochłonne i łatwo popełnić błąd dobierając pozytywy i negatywy. Utworzony na potrzeby projektu wzorec zwracał zbyt wiele wyników fałszywie pozytywnych, tzn. oznaczał miejsca bez karty jako pasujące. Przyczyną takiego stanu może być fakt, że po przeskalowaniu obrazu karty do rozmiaru 24x24 piksele, część szczegółów zanika i wzorec łatwo dopasowywał się do obrazów tła.

6 Możliwe kierunki rozwoju

Niestety z powodu ograniczeń czasowych nie udało się zbadać wszystkich aspektów porównania obu metod. Prawdziwy potencjał metody SURF mógłby ujawnić się przy wygenerowaniu punktów kluczowych wzorca za pomocą serii poprawnych wzorców. W takiej sytuacji algorytm SURF nadal byłby wolniejszy od metody opartej o detektor Haara i momentów Hu, jednak liczba poprawnych dopasowań mogłaby być lepsza i nawet bliska liczbie z metody Haara. Niezależnie od zbioru uczącego, liczba błędnych dopasowań powinna być zdecydowanie mniejsza w metodzie SURF. Warto byłoby także porównać czasy przetwarzania metody SURF zaimplementowanej w architekturze GPU. Możliwe, że w takich warunkach metoda SURF uzyskałaby czasy podobne do metody opartej o detektor Haara.

Literatura

- [1] Kurt Konolige Gary Bradski Ethan Rublee, Vincent Rabaud. *ORB: an efficient alternative to SIFT or SURF*. 2012.
- [2] David G. Lowe. *Distinctive Image Features from Scale-Invariant Keypoints*. 2004.