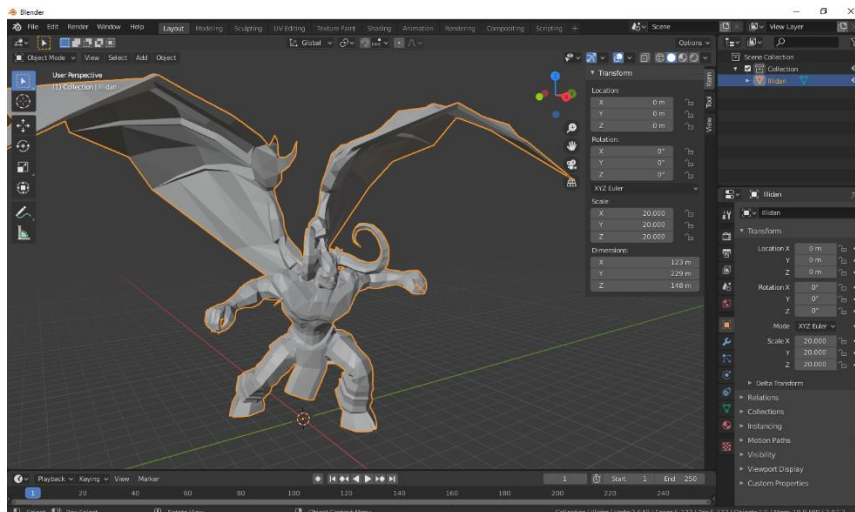


Instrukcja 12 - AR (Augmented reality)

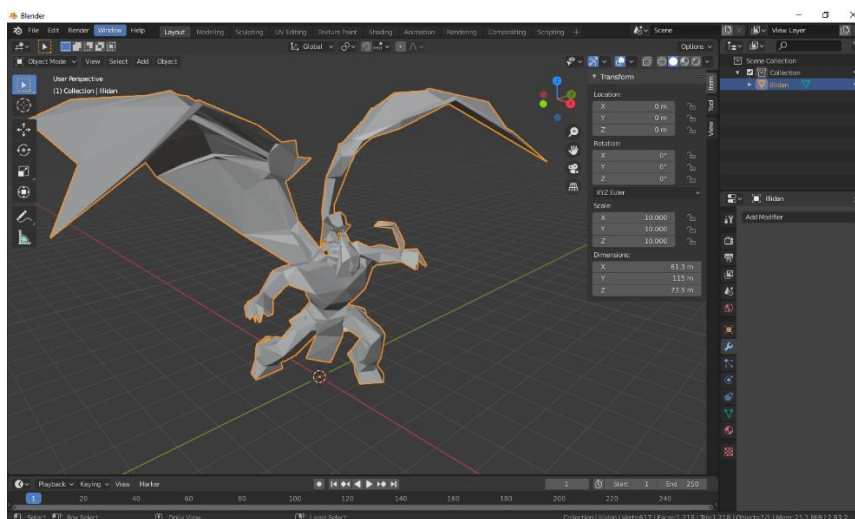
1. Program i model 3d

Model wykorzystany w programie to postać Illidan Stormrage ze świata Warcraft. Ten model wyeksportowałem z plików gry za pomocą WowModelViewer w zeszłym semestrze do projektu, więc format .stl musiałem przekonwertować do .obj za pomocą aplikacji Blender. Model miał ponad 5 tysięcy wielokątów, co drastycznie spowalniało działanie programu, więc z użyciem narzędzia „decimate”, zmniejszono złożoność modelu niemal pięciokrotnie – z 5232 trójkątów do 1218.

Model oryginalny:



Model po decymacji:



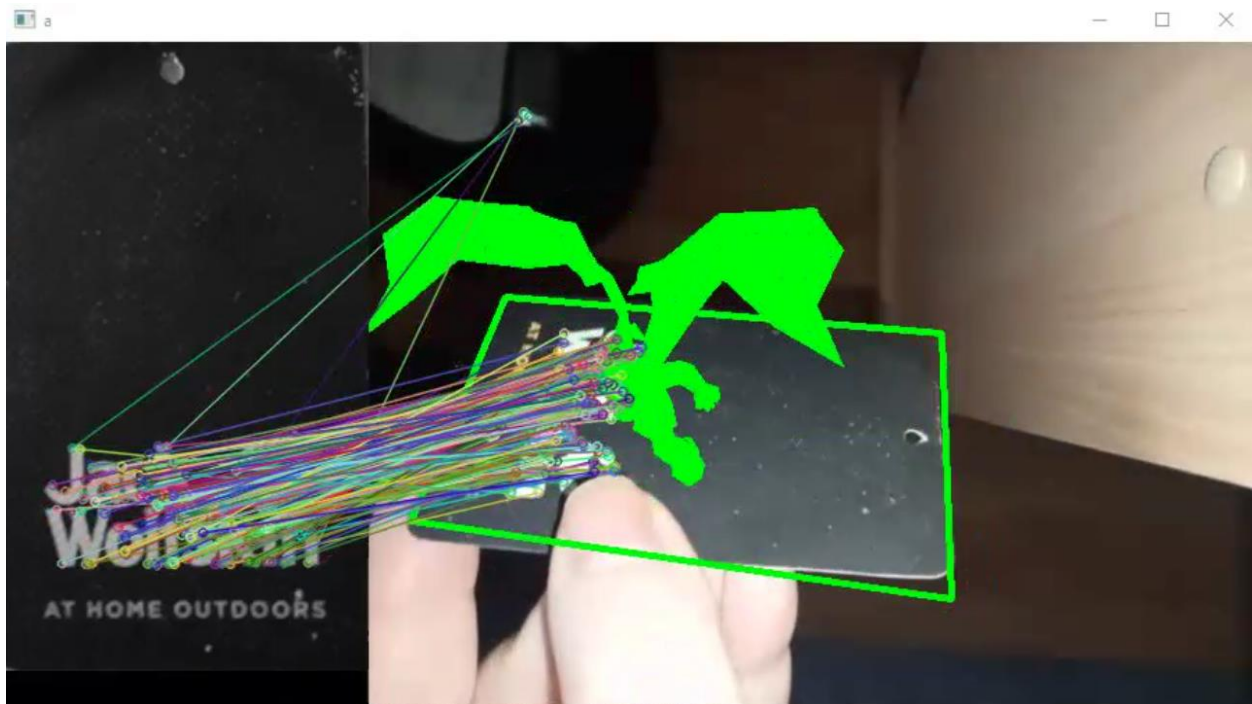
Iriun Webcam pomimo jakichś błędów, które po kilku godzinach działania „mulity” cały komputer, było lepsze od DroidCam nie tylko w szybkości działania, ale też pozwalało uruchomić lampę w telefonie, co okazało się bezcenne, bo w niskim oświetleniu algorytm pracował bardzo źle. Jednak zmiana rozdzielczości w aplikacji nie wpływała na rozdzielczość obrazu z kamery – DroidCam nie oferował żadnych opcji i działał wolniej.

2. Znaczniki i testy

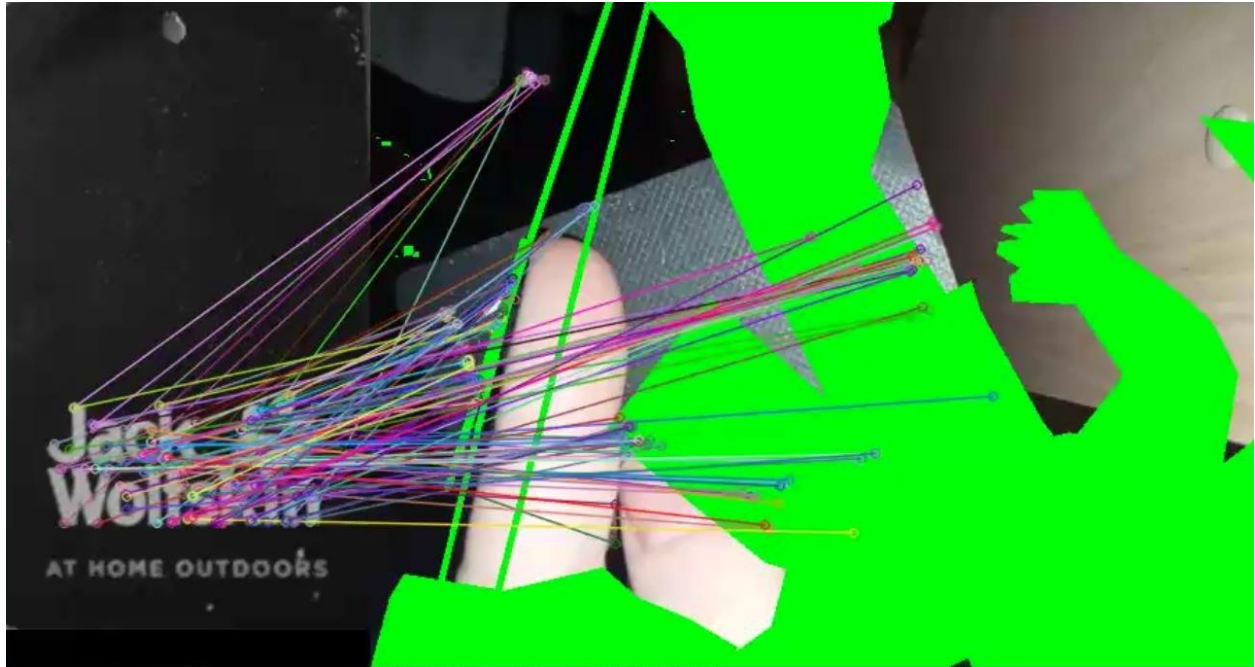
a) przód etykiety plecaka



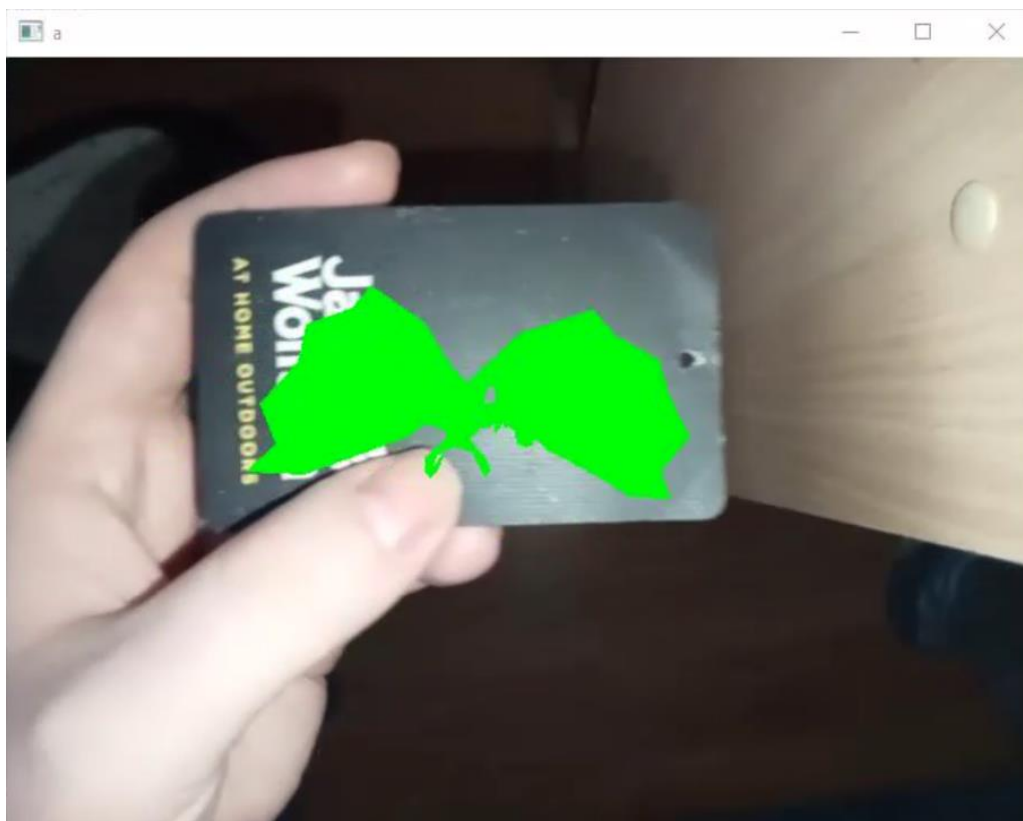
Wzorzec do znalezienia jest głównie czarny, z małą częścią z napisami, które są zgodnie z przewidywaniem, właśnie tą częścią gdzie algorytm wykrywa punkty kluczowe. Pomimo znalezienia kilku punktów w tle, program prawidłowo wykrywa prostokąt i jego zwrot w kierunku kamery:



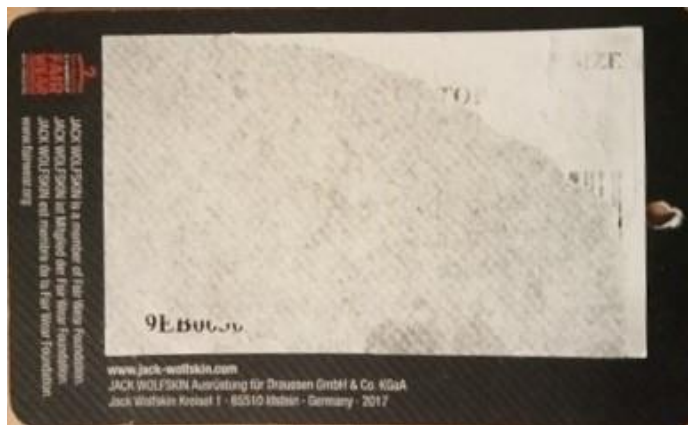
Po zakryciu większej części napisu na etykiecie, algorytm niepoprawnie wykrywa inne fragmenty etykiety, oraz mimo bliskości kamery z lampą – wykrywa wiele punktów w tle (metalowa część odbija najwięcej światła) i model wariuje na ekranie:



Model wyświetlany bez znaczników i rysowanego prostokąta:

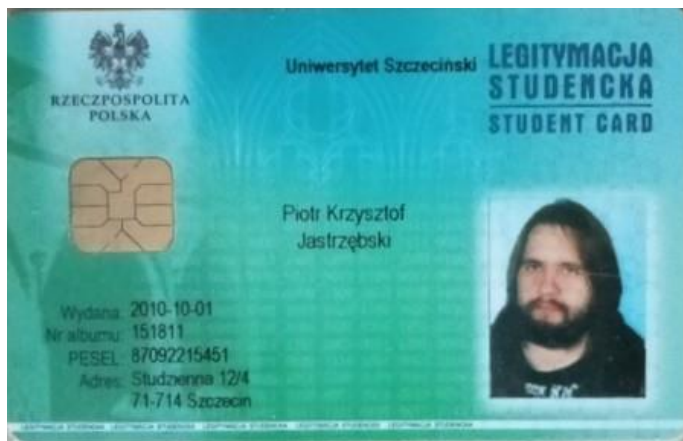


b) tył etykiety plecaka



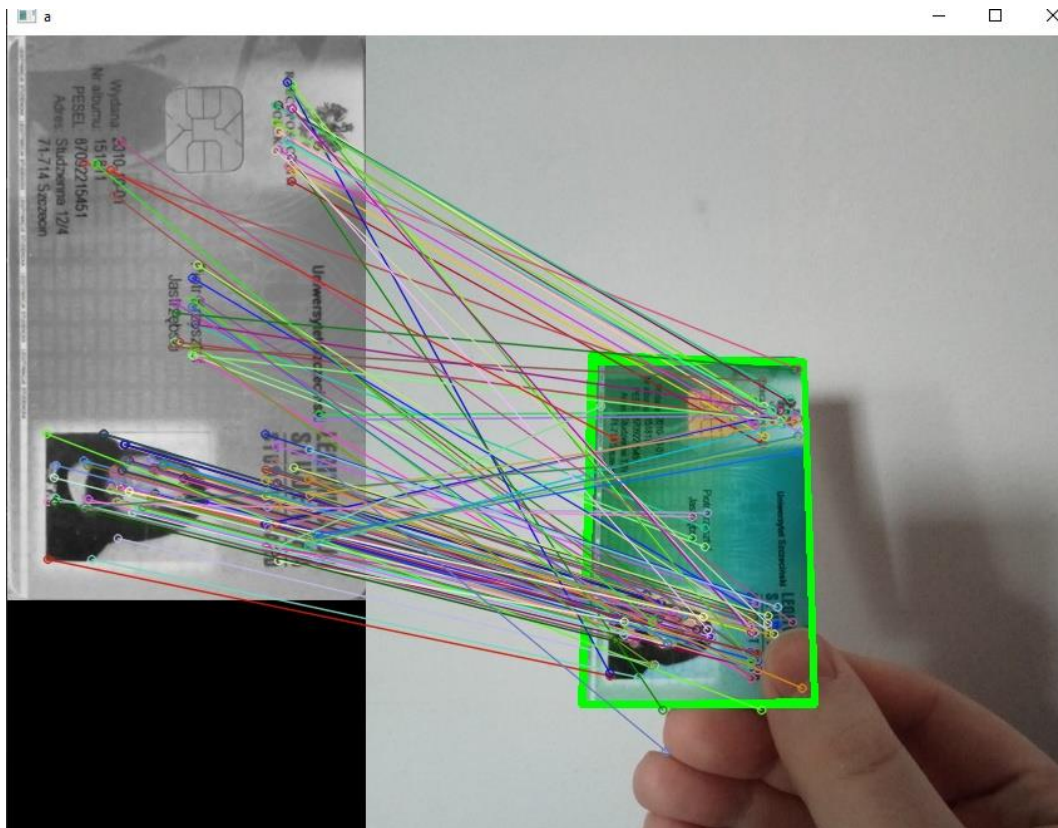
Sytuacja wyglądała podobnie do przodu etykiety, ale nie udokumentowałem przed porysowaniem tej strony. Zaskakującym było niewykrywanie narożników białego prostokąta, ale raczej napisów włącznie z resztkami zerwanej naklejki z ceną.

c) legitymacja studencka

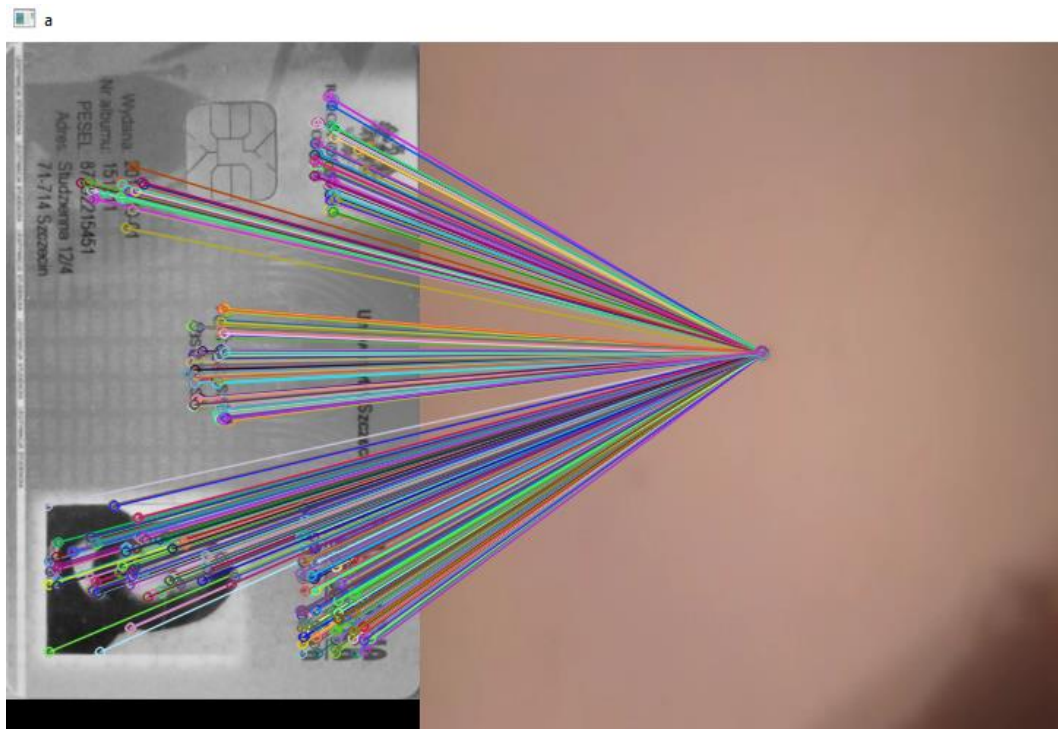


Założeniem było, że legitymacja jest monotonna wizualnie, więc może zdjęcie i orzełek będą głównymi elementami punktów kluczowych z racji wysokiego kontrastu oraz bazując na poprzednich próbach, metalowy element może mieć ciekawy wpływ na wykrywanie.

Wynik był dość zaskakujący. Na pierwszy rzut oka widać, że zdjęcie zostało wykryte i dobrze dopasowane, ale orzełek nie. Metalowa część została kompletnie zignorowana, zaś litery zostały wykryte, ale bardzo źle dopasowane! Mimo to, prostokąt został niemal idealnie dopasowany, prawdopodobnie dzięki idealnie jednolitemu tłu ściany:



Interesujący wynik wykryty na ścianie, która miała ślad po brutalnej śmierci pająka:



d) narysowany kod QR



Ten wzorec wypadł bardzo źle. Gubił model bardzo łatwo, pod lekkim kątem:



Albo z całkiem bliskiej odległości:



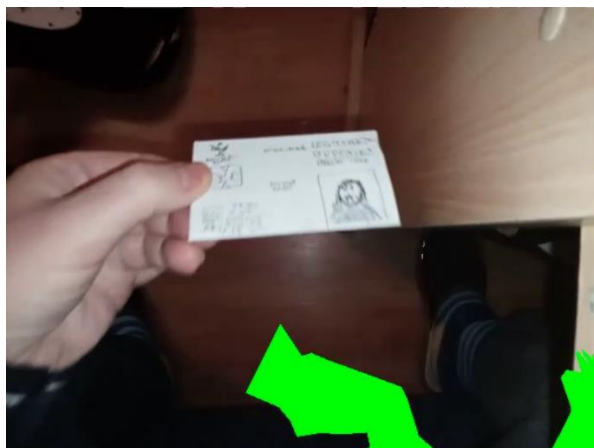
e) narysowana legitymacja studencka



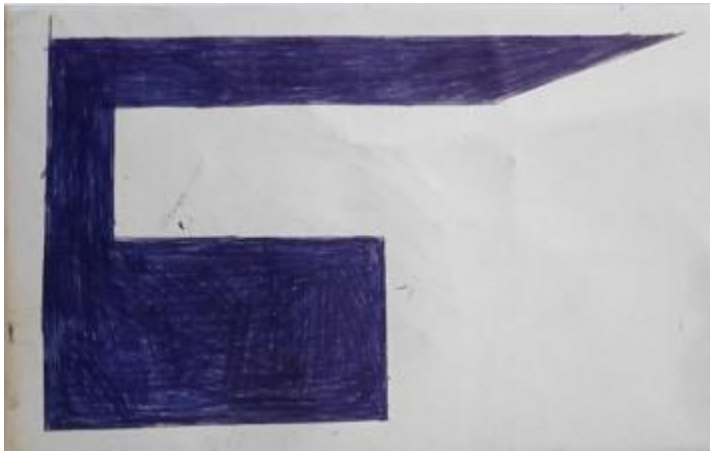
Wzorzec przygotowany jako niemal żart okazał się szokująco skuteczny:



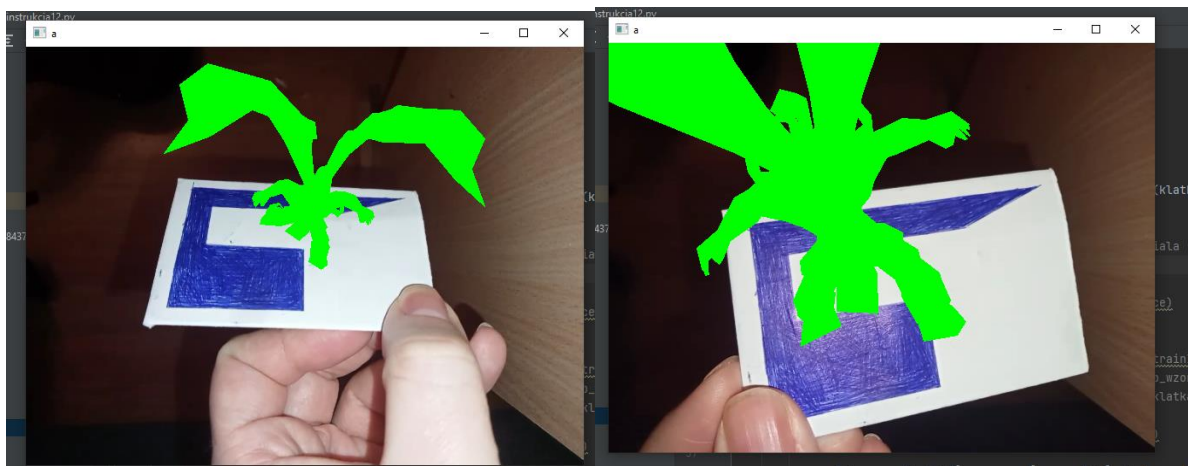
Szokująco skuteczny to wciąż było wciąż relatywne, model gubił się przy nachyleniach bądź szybszym ruchu, który rozmazywał obraz:



f) narysowany abstrakcyjny znacznik

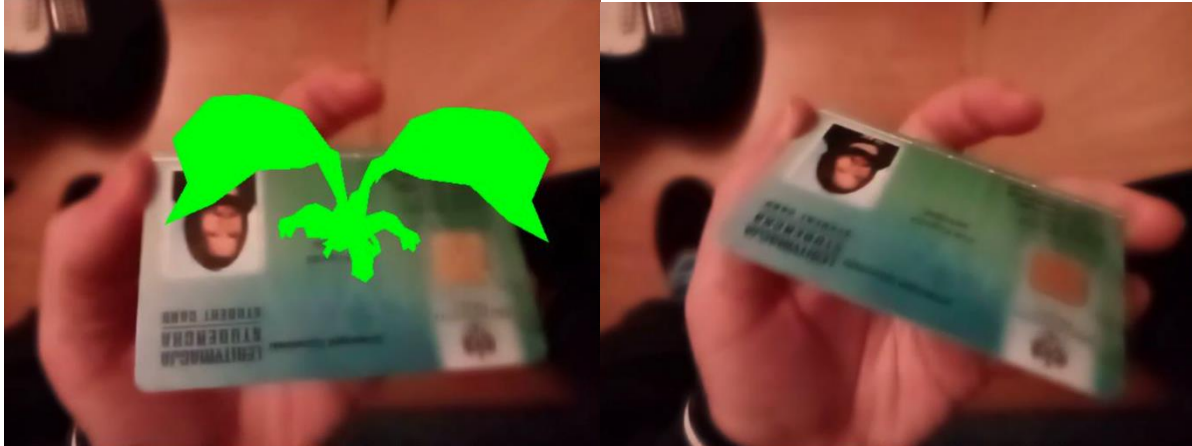


Skuteczność wzorca narysowanego niebieskim długopisem okazała się mocno zależna od oświetlenia. Niebieski tusz odbijał dużo światła i zależnie od nachylenia, potrafił przekłamać informacje mimo bardzo małej odległości od kamery:

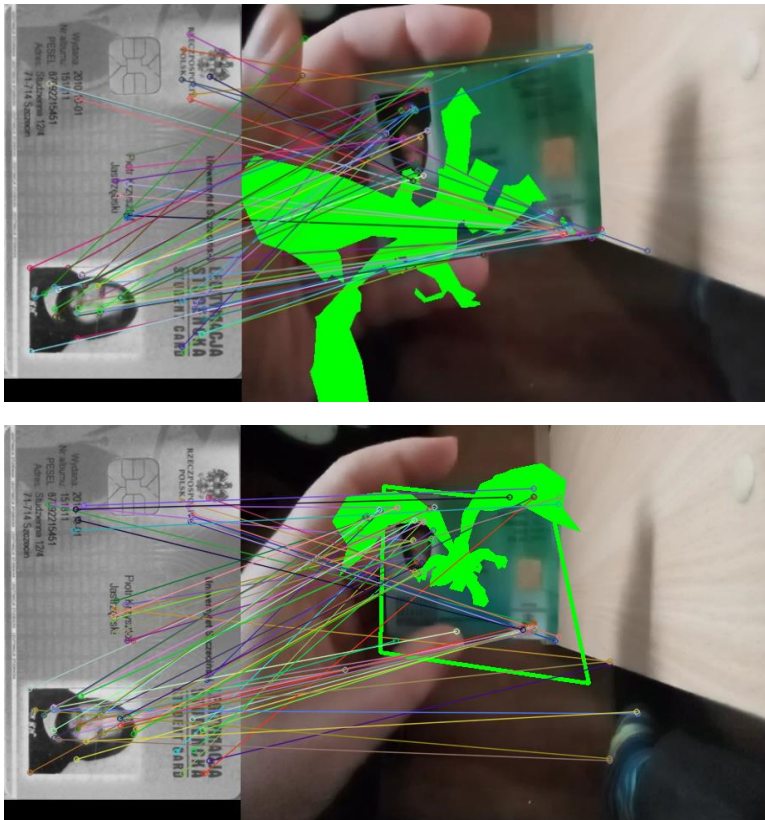


3. Inne testy

Niskie oświetlenie – wzorec był rozmazany w kamerze i bardzo łatwo gubił znaczniki. Dodatkowo, nawet przy poprawnym wykryciu, wyliczenia różniły się co klatkę, mimo pozornie nieruchomego obrazu, co sprawiało, że modelem telepało jak studentem przed sesją.



Światło dzienne – duży wpływ na losowość wyników, czasem znajdowało prostokąt poza ekranem, a czasem wykrywało różne obiekty poza wzorcem, dzięki silnemu oświetleniu tła.



Test mema – wszystko fajnie



Brak oświetlenia – w ciemności program wyświetla model na środku kamery, zasłaniając cały obraz

