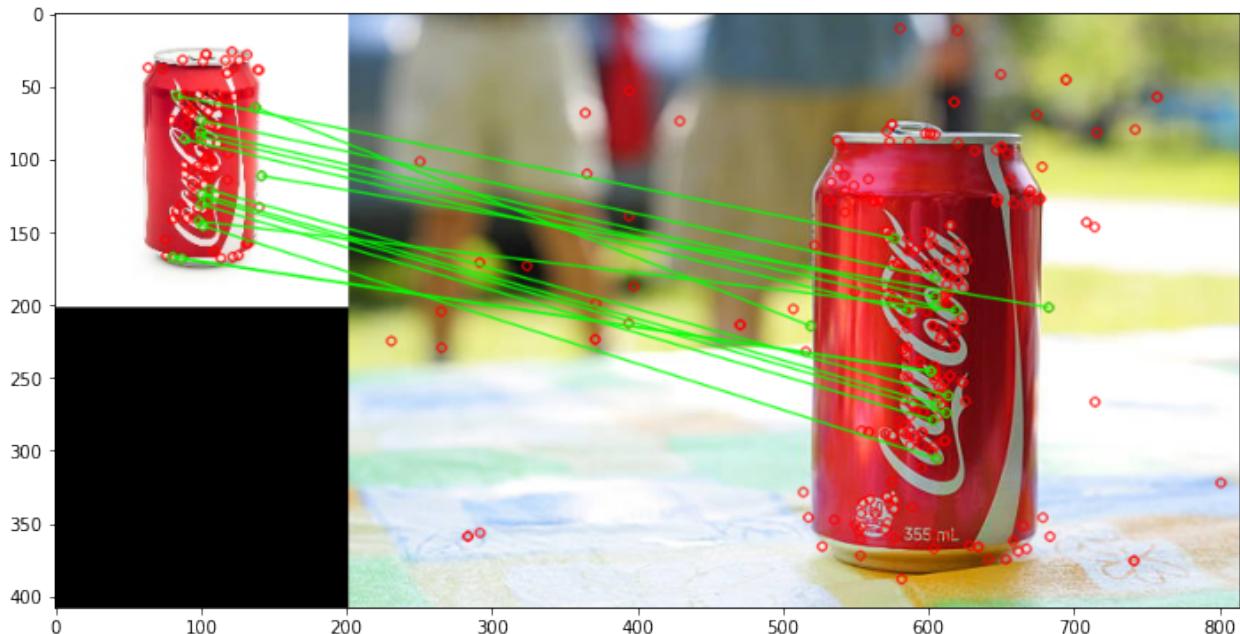


**Zadanie 1a**

Celem zadania była detekcja punktów kluczowych i sparowanie ich używając matchera z cv2. Oprócz tego trzeba było dokonać filtracji znalezionych par pod względem dystansu pary.

Do tego zadania użyłem matchera `cv2.BFMatcher`. Pary filtrowałem funkcją sprawdzającą czy dystans w parze jest mniejszy od pewnego ustawionego progu.



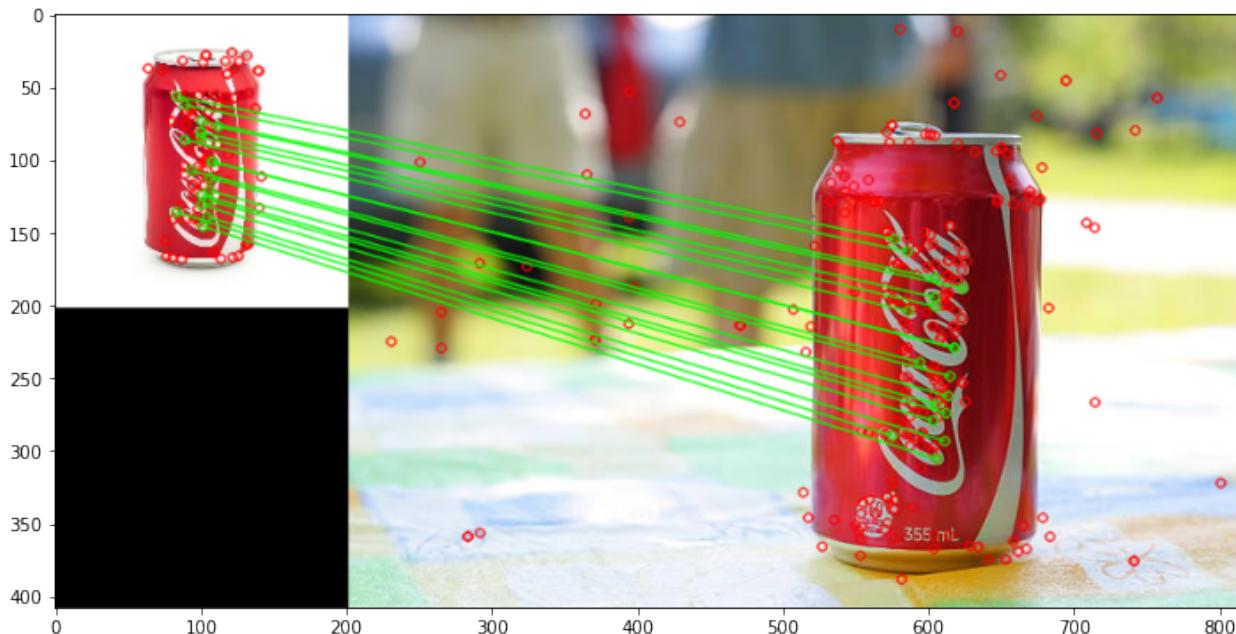
Rysunek 1: Pary odfiltrowane z progiem 200

W miarę zmniejszania progu zwiększa się liczba odfiltrowywanych par. Pomaga to pozbyć się większości niepoprawnie sparowanych punktów. Mimo to jak widać na obrazie wciąż widać złe sparowane punkty. Można na przykład zauważyć parę złe sparowanych punktów na krawędzi puszki, które wyglądają jakby zostały dobrane, ponieważ wyglądają jak ten sam punkt kluczowy, tylko odwrócony do góry nogami, co wyjaśnia też krótki dystans. Udowadnia to, że dystans nie jest metryką, która pozwala jednoznacznie określić czy para jest dobra czy nie.

**Zadanie 1b**

Celem tego zadania było sparowanie punktów metodą `knnMatch` i znalezienie metody filtrowania par na podstawie unikatowości dopasowań.

Dopasowywania knn użyłem z  $k = 3$ . Moja metoda filtrowania sprawdza dla każdego punktu kluczowego jego 3 dopasowania, a następnie sprawdza różnicę pomiędzy dystansami dopasowania z najmniejszym dystansem i dopasowania z drugim najmniejszym dystansem. Jeżeli ta różnica jest większa od danego progu to dopasowanie jest dodawane do zbioru wynikowego, odrzucane w przeciwnym wypadku. Filtrowanie w taki sposób zostawia tylko takie punkty kluczowe, dla których można znaleźć dopasowanie z pewną miarą pewności że nie wybraliśmy złej pary zamiast dobrej.



Rysunek 2: Pary odfiltrowane z progiem 100

Jak widać tej metodzie dla progu 100 udało się odfiltrować wszystkie błędy. Filtrowanie staje się agresywniejsze w miarę zwiększania progu. Jak widać unikatowość dopasowania jest wystarczającym warunkiem umożliwiającym odróżnienie dobrych od złych dopasowań. Dzieje się to, ponieważ w przypadku istnienia prawdziwego dopasowania, jego dystans powinien być znaczaco mniejszy od dystansów nieprawidłowych dopasowań. Jeżeli żadne dopasowanie się nie wyróżnia to albo wszystkie są nieprawidłowe albo to prawidłowe jest zbyt podobne do nieprawidłowych i na wszelki wypadek należy takie punkty odrzucić.

## Zadanie 1c

Celem tego zadania było wykorzystanie znalezionych par punktów kluczowych do przekształcenia pierwszego obrazu i nałożenia go na drugi za pomocą `cv2.warpPerspective`.

Jako pary punktów kluczowych wykorzystałem te znalezione w zadaniu 1b. Obrazy połączylem nakładając je na siebie i uśredniając wartości pikseli, następnie nałożyłem punkty kluczowe. Wyświetliłem tylko te, które w wyniku przekształcenia znalazły się w okolicy o ustalonym promieniu od swojej pary. Do wszystkich poniższych eksperymentów użyłem promienia 10.



Jak widać wszystkie punkty znalazły się na napisie na puszcze. Co za tym idzie obraz został przekształcony tak, że napis pokrywa się idealnie.

Jak widać źle przefiltrowane pary powodują, że transformacja przekształca obraz w nieporządzany sposób. Wzmocnienie filtracji ma niewielki wpływ na wynik.

## Zadanie 2

Celem zadania było zastosowanie metod z zadania 1 na innych obrazach.

Jak widać metoda z zadania 1 dalej nie radzi sobie dobrze z obrazami o jeszcze większej liczbie detali, w przeciwieństwie do metody z 1b dla wszystkich obrazów. Przekształcenia też działają, ale dla obrazów `detail` i `side` wymagały wyższych progów przy filtrowaniu par, w przeciwnym wypadku były znieksztalcone. Dlaczego `detail` wymagał wyższego progu od



Rysunek 3: Pary odfiltrowane z progiem 50

`fine`? Prawdopodobnie większa ilość detali generuje więcej punktów kluczowych, więcej par i więcej nieprawidłowych par, co zniekształca transformację, jeżeli nie zostanie nadrobione wzmacnieniem filtrowania.

