

## Opis projektu

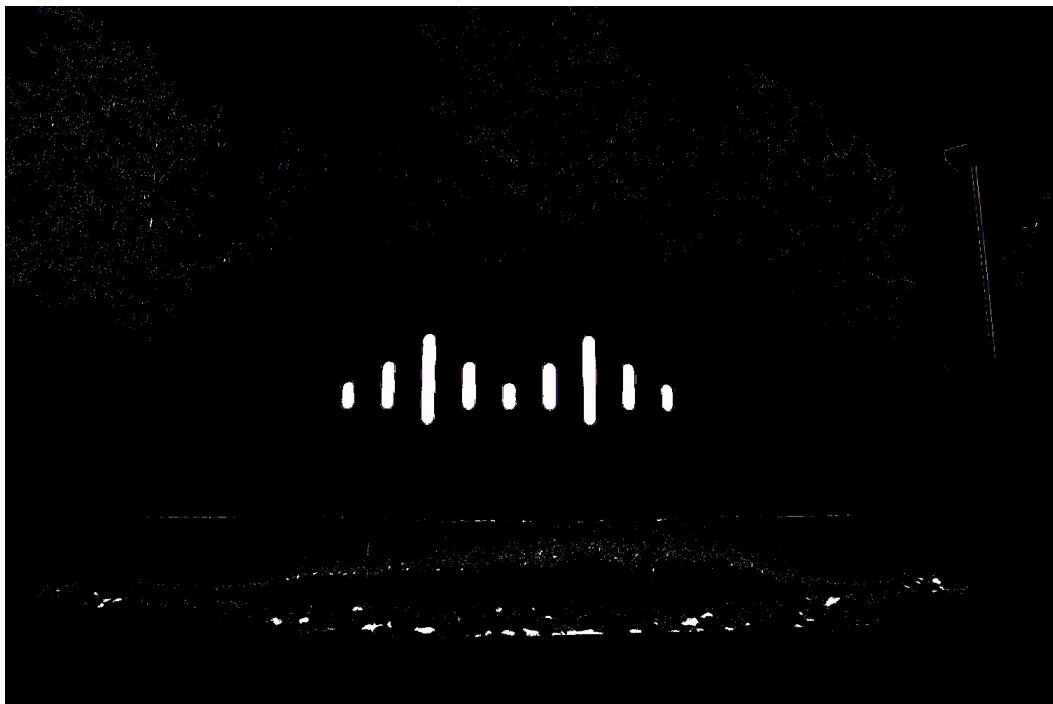
Celem projektu była implementacja wykrywania logotypu na zbiorze testowym zdjęć zawierających różne jego wariacje.



Logotyp firmy Cisco

## Wstępne przetworzenie

W początkowych fazach algorytmu, następuje przekształcenie modelu barw do przestrzeni HSV. Zmiana ta umożliwia następnie wyodrębnienie z oryginalnego zdjęcia elementów mogących kwalifikować się jako litery (elementy czerwone) lub fragmenty logo (części niebieskie/granatowe). Efektem tej fazy jest stworzenie dwóch zdjęć. Na pierwszym zdjęciu na białą zaznaczone są piksele niebieskie/granatowe a reszta zaznaczona jest na czarno. Na drugim zdjęciu na białą zaznaczone są piksele czerwone a reszta zaznaczona jest na czarno. Poniżej przedstawione są obrazy otrzymywane pod koniec tego etapu przetwarzania.



Wyodrębnione niebieskie kolory



Wyodrębnione czerwone kolory

### **Wykrywanie segmentów oraz filtracja**

Obrazy uzyskane jako efekt poprzedniego etapu są analizowane oddzielnie. Każdy piksel zdjęcia jest sprawdzany i w przypadku natrafienia na biały kolor, algorytmem zalewowym wykonywane jest wykrycie reszty pikseli należących do danego segmentu. W celu uniknięcia wielokrotnego analizowania tych samych punktów, wszystkie punkty, które zostały już sprawdzone są oznaczane losowym kolorem. Po utworzeniu takiej listy segmentów, jest ona jeszcze raz sprawdzana i odfiltrowywana z segmentów, których wielkość nie zawiera się w przedziale określonym po przeprowadzeniu wielokrotnych testów. Proces ten znacznie skraca czas działania algorytmu ponieważ ogranicza liczbę później analizowanych segmentów.

### **Analiza wykrytych segmentów**

Dla każdego z wcześniej otrzymanych segmentów wyliczane są niezmienniki momentowe: M1, M2, M3, M7 oraz jego punkt centralny. Po wyliczeniu tych wartości następuje segregacja segmentów na podstawie eksperymentalnie otrzymanych zakresów niezmienników, charakterystycznych dla każdego z elementów (poszczególne litery oraz fragmenty logo). Wynikiem tego działania jest otrzymanie segmentów wraz z informacją o ich charakterystyce.

### **Grupowanie segmentów**

W tym etapie, wcześniej analizowane segmenty są grupowane. Ponieważ niezmienniki momentowe litery 's' są najbardziej charakterystyczne to właśnie ta litera jest punktem wyjścia grupowania. Dla każdej możliwej litery 's' w naszym zbiorze wyszukiwane są najbliższe położone pozostałe litery. Tak otrzymany zbiór jest następnie sprawdzany pod kątem drastycznych zmian współrzędnych punktów centralnych poszczególnych segmentów co świadczyło by, iż najbliższe dobrane segmenty nie są ze sobą powiązane i stanowią szum lub należą do innej grupy segmentów. Następnie każda litera z danej grupy jest analizowana aby wyznaczyć czy napis jest poziomy czy pionowy, i w którym z kierunków powinniśmy szukać segmentów logo. Ponieważ na tym etapie znamy już położenie segmentów logo możemy ograniczyć ten zbiór do tych, których położenie jest

zgodne z przeszukiwanym przez nas kierunkiem. Z tak przefiltrowanego zbioru wybierane jest 9 najbliższych elementów, które składają się na logo. Otrzymane segmenty dodawane są do zbioru liter dzięki czemu posiadamy zgrupowane segmenty składające się na cały logotyp. Tak wykryty logotyp jest następnie oznaczany na oryginalnym zdjęciu na podstawie skrajnych wartości współrzędnych segmentów.

### Wynik działania algorytmu

Algorytm poprawnie wykrywał wszystkie logotypy występujące w końcowym zbiorze testowym. Oprócz sprawdzania działania programu na końcowym zbiorze testowym, przeprowadzone zostały również testy na zdjęciach, które zostały z niego usunięte. Powodem tego usunięcia był fakt, iż logotyp występował na nich na tle zbliżonym do koloru logo. Z tego powodu, pierwszy etap przetwarzania nie wyodrębniał segmentów logo na tyle dokładnie aby późniejsza ich analiza dawała oczekiwane wyniki. Poniżej przedstawione są wyniki działania pierwszego etapu przetwarzania dla takiego zdjęcia oraz przykład poprawnej analizy zdjęcia z końcowego zbioru testowego.



Wyodrębnione czerwone kolory



Wyodrębnione niebieskie kolory



Wynik działania programu