

# Metody Rozpoznawania Obrazów

## Raport 2.2

Oscar Teeninga

### 1. Wycinanka

Stworzyłem 8 kształtów parasola stosując metodę zginania kartki, aby kopie były odzwierciedlone najlepiej jak się da. Niestety, nie wszystkie parasole są identyczne, więc można zakładać, że pojawią się pewne błędy w działaniu samego wykrywania krawędzi. Ze względu na szerokokątny obiektyw w moim telefonie również trudno było osiągnąć tę samą wielkość wszystkich kopii bez znaczącego zoomu cyfrowego, który mógłby zmniejszyć znacznie jakość samego zdjęcia.



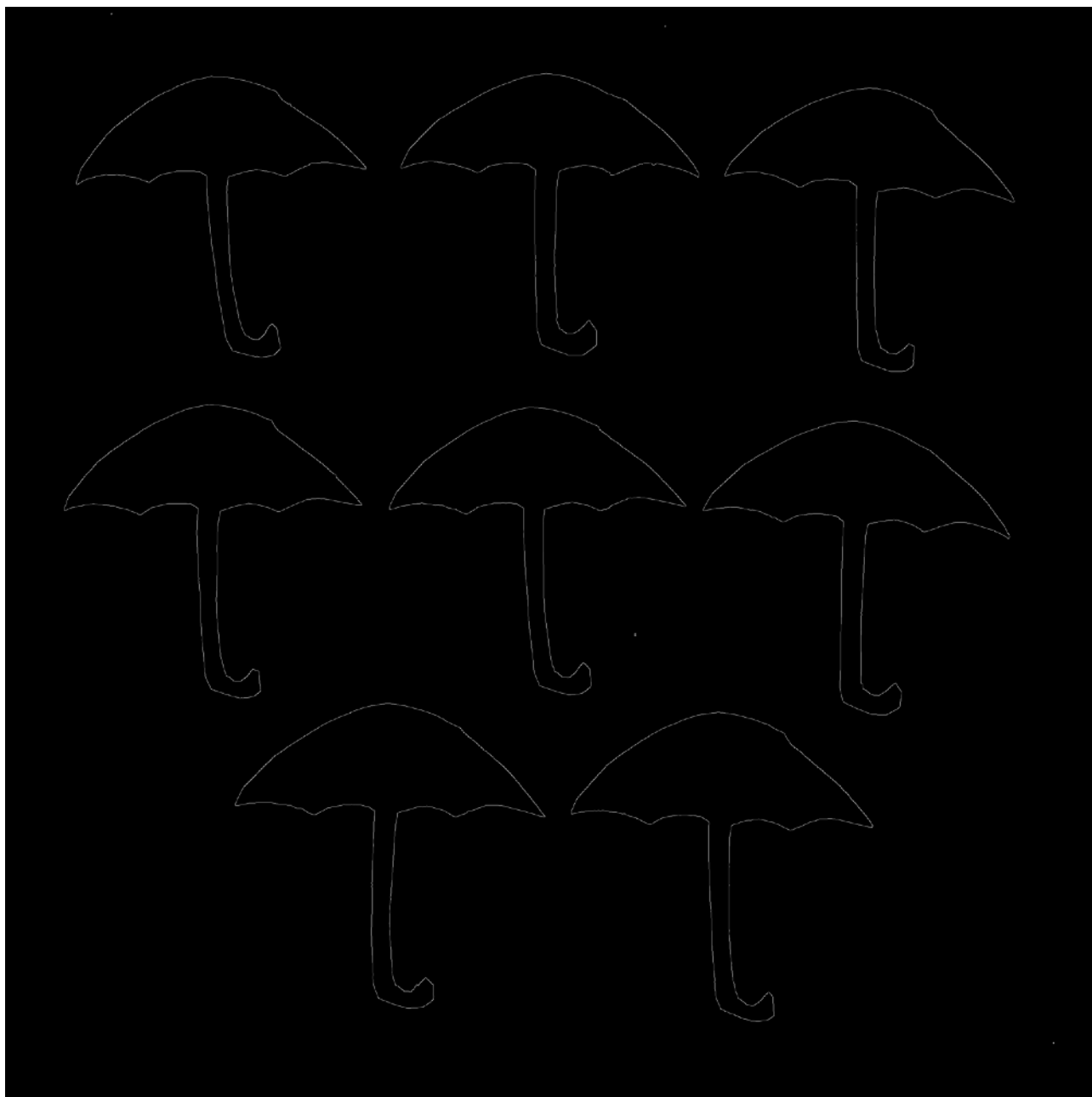
## 2. Modyfikacja w programie graficznym

Ponieważ samo tło nie było idealnie czarne postanowiłem podrasować kolory w programie graficznym oraz zmienić proporcje obrazka na kwadratowe. Zastosowałem darmowy program GIMP.



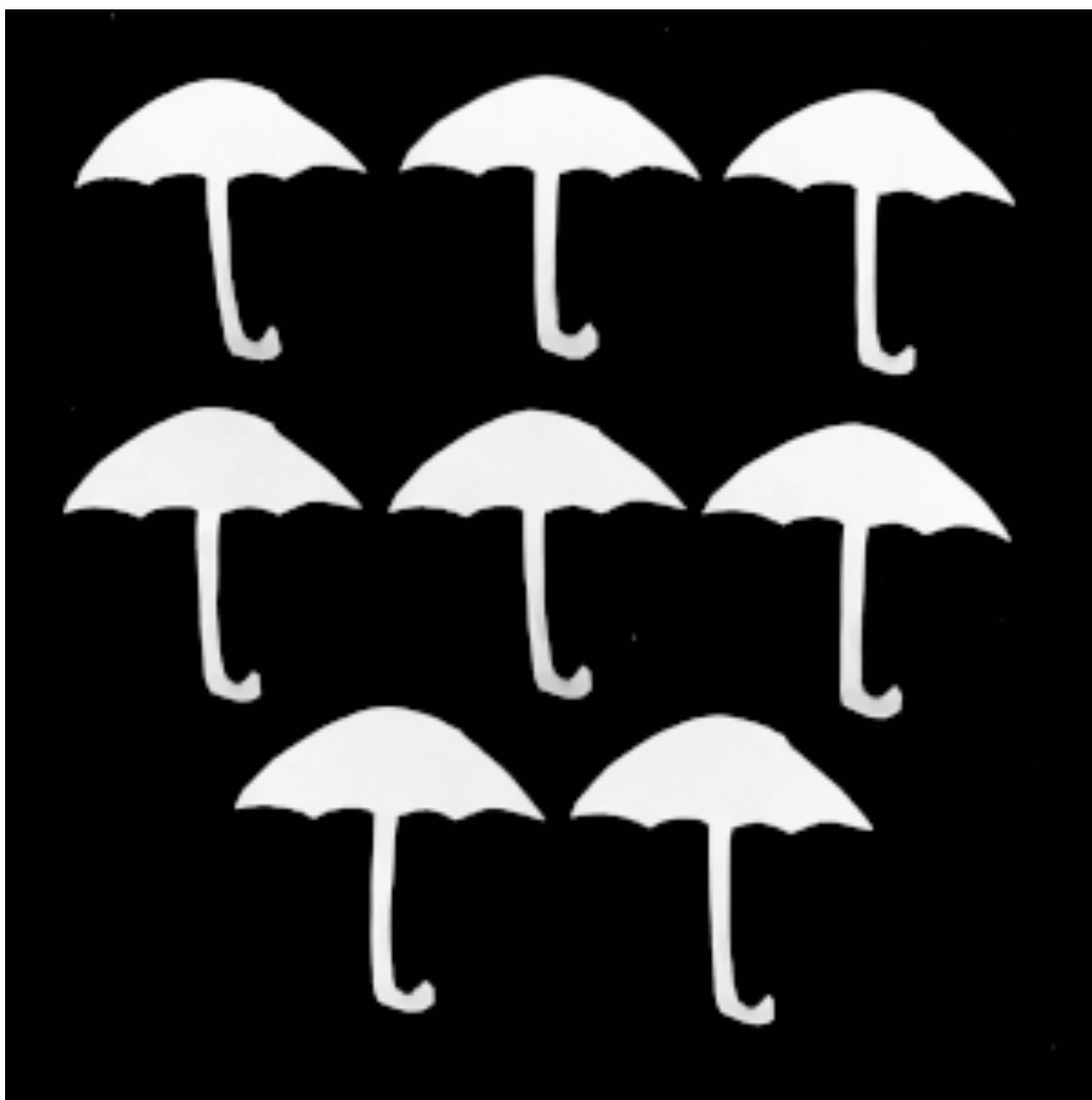
### 3. Canny

Następnie dokonałem modyfikacji obrazu za pomocą biblioteki opencv oraz dostarczonej przez nią metody Canny.



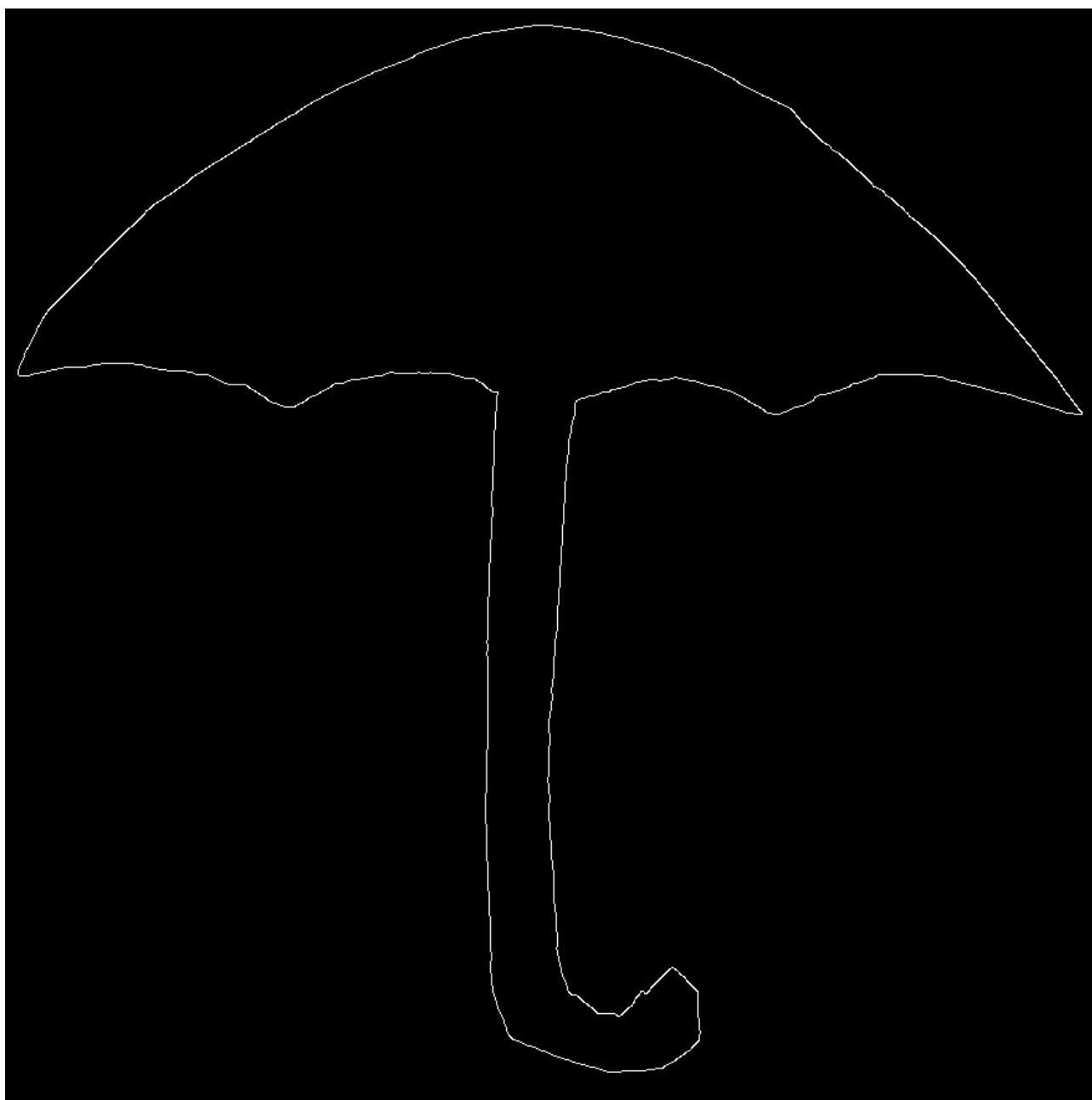
4. Zmiana rozdzielczości

Obraz miał rozdzielczość 3024x3024, stworzyłem dodatkowo testowy obraz 300x300.



## 5. Template

Wybrana parasolka jest to dolna po lewej stronie.



6. GHT dla pomniejszonego obrazu

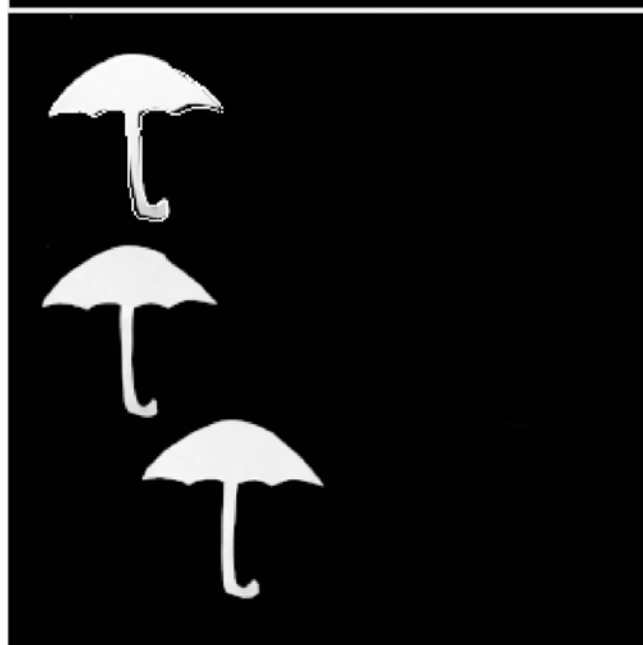
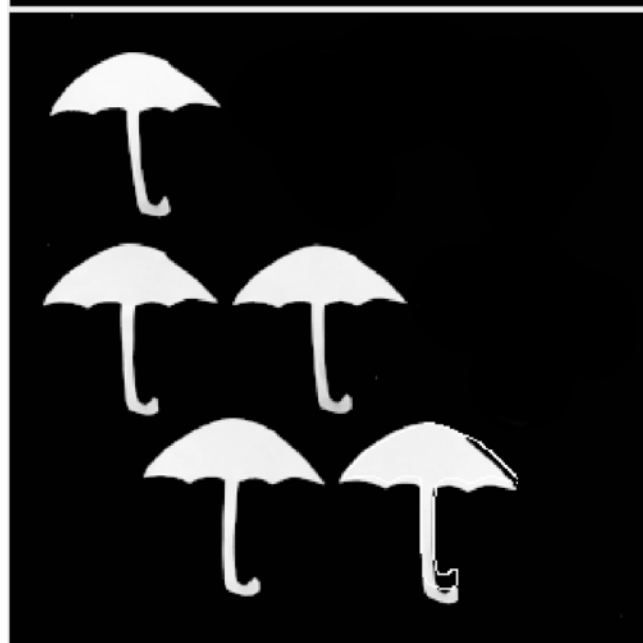
Następnie przeprowadzałem transformatę Hougha dla parasolek korzystając z template. Skorzystałem z projektu ze strony MatLab:

<https://www.mathworks.com/matlabcentral/fileexchange/50330-generalized-hough-transform-with-rotation>.

Algorytm był w stanie wykryć tylko jedno dopasowanie. Dla obrazka 300x300 czas działania zmieścił się w 5 sekundach.



Jak widać, udało się odnaleźć kopniętą parasolkę za pomocą tego algorytmu. Jeżeli zasłanialibyśmy parasolki już wykryte to za każdym razem dostawalibyśmy następną.



7. GHT dla pełnego obrazu

W tym przypadku czas ewaluacji algorytmu to było ~ 5 minut, więc nie przeprowadzałem już symulacji odnajdywania kolejnych obrazków. Ze względu na kolor i rozdzielczość nie widać tego dobrze, jednak jest dostrzegalne obramowanie wokół lewego, dolnego obiektu, a więc templateowego. Zmieniłbym kolor obramowania, jednak kolor był typu int i rozważałem tylko skalę szarości.





## 8. Wnioski i podsumowanie

Jak widzimy GHT jest w stanie odnajdywać dowolny kształt na obrazie. Co ciekawe w przypadku obrazka niższej rozdzielczości okazało się, że najlepszym dopasowaniem okazała się parasolka, która nie była wzorcem. Algorytm był pisany w ten sposób, żeby rozważać obiekty już rotowane, biorąc jako start 1 stopień, co może tłumaczyć to dziwne zjawisko. Natomiast dla wysokiej rozdzielczości już nie ma zaskoczenia, pierwszą wybraną przez algorytm próbką jest ta wzorcowa. Więcej punktów, większa precyzja nie pozwala tutaj pomylić się algorytmowi.

Porównując tę metodę do innych sposobów klasyfikacji i odnajdywania przedmiotów na obrazie, np. opierających się na CNN - YOLO, zdecydowanie widać tutaj większą precyzję w interpretacji obiektów. GHT pozwala z bardzo wysoką dokładnością określić kształt poszukiwanego obiektu, gdzie YOLO może korzystać z bardziej generalnych klas takich jak np. człowiek. Dużym minusem GHT jest wolne przetwarzanie, jednak implementacja nie zakładała działania równoległego, a więc jest to pewne pole do manewru.

Dużą zaletą również GHT jest praktycznie zerowa potrzeba nauki i zbioru testowego, wystarczy jeden kształt i możemy go potem szukać na dowolnym obrazie. Oczywiście dobrze byłoby dopasowywać na podstawie kilku kształtów, ponieważ nawet kąt patrzenia na obiekt w pewnym stopniu go skaluje, ale jest to niewielki koszt względem kilkutysięcznych zbiorów treningowych i długich nauk sieci na kartach graficznych niezbędnych do prawidłowego działania CNN.