

# AUTOMATYCZNE ROZWIĄZYWANIE PUZZLI



ALICJA FIGAS

MARCIN GRUZA



# Dlaczego ten temat?



## Puzzle są fajne

Dobra okazja do  
eksploracji tradycyjnych  
metod CV

## Brak istniejących rozwiązań

Obecne rozwiązania  
skupiają się na  
kwadratowych  
elementach

## Złożony problem

Mieliśmy tu kilka  
podproblemów CVi nie  
tylko :)

# Założenia upraszczające



Puzzle pochodzą z  
jednego obrazka



Niewielka liczba  
elementów



Jednorodne tło



Puzzle wygenerowane  
komputerowo lub zdjęcia

# Metodologia





# Segmentacja puzzli

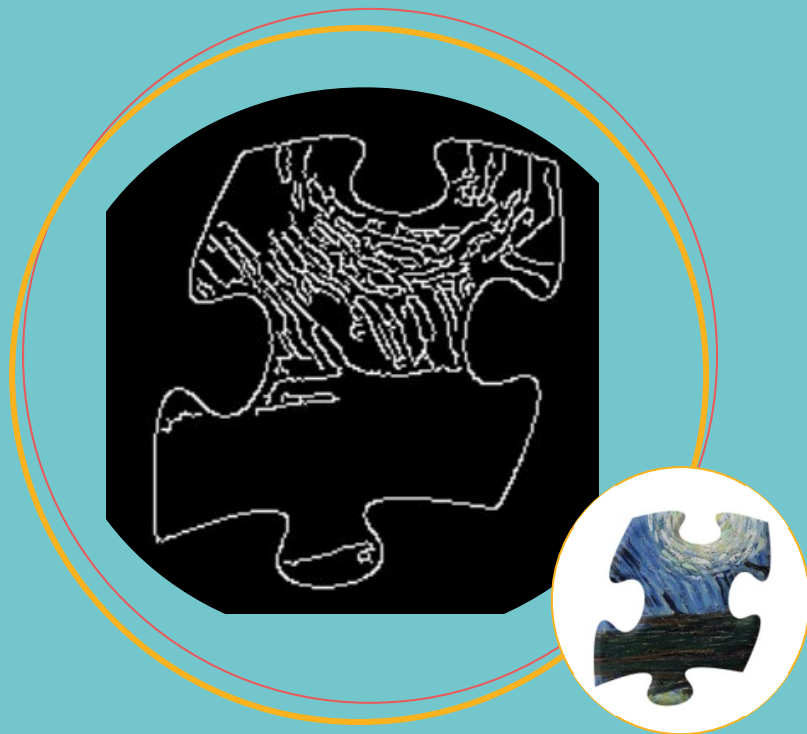




# Detekcja krawędzi



Za pomocą filtra Canny'ego  
znajdujemy wszystkie  
krawędzie na obrazie

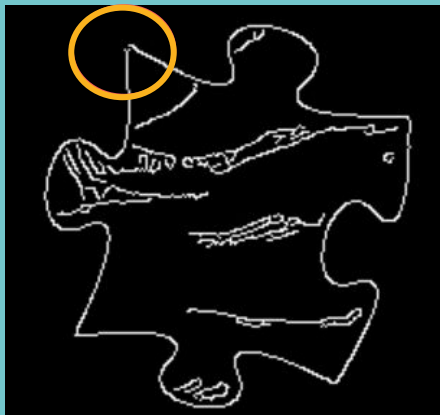




# Operacje morfologiczne



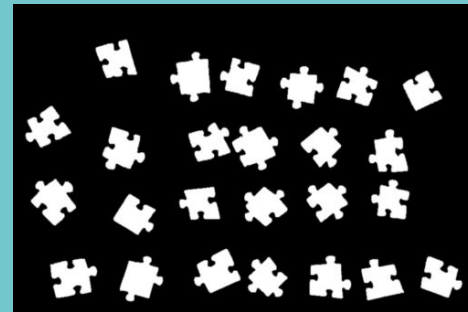
Pozwalają nam na domknięcie  
kawałków puzzli





# Znalezienie kawałków puzzli

Znajdujemy najbardziej zewnętrzne  
kontury







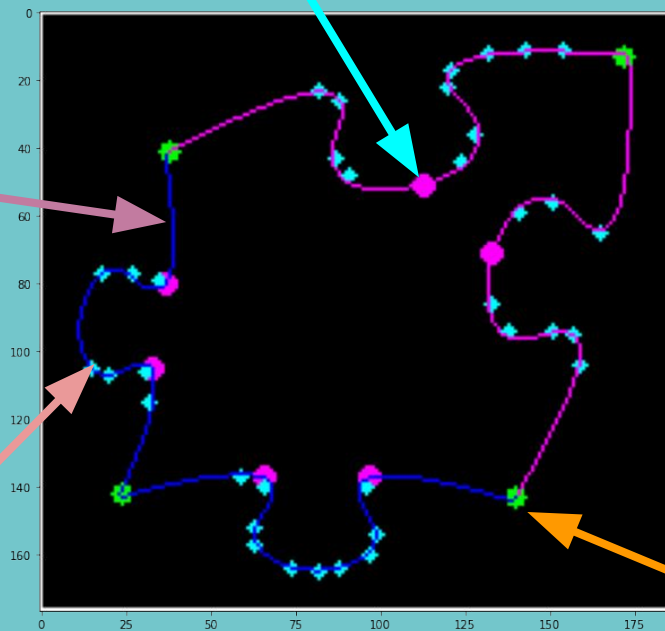
Punkty oddalone od  
otoczki wypukłej

# Rozpoznawanie krawędzi



Typ krawędzi

Potencjalne  
narożniki



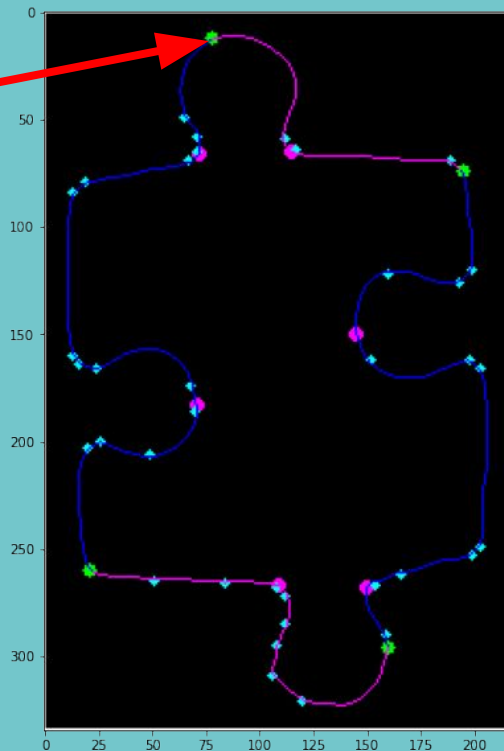
Narożniki puzzla



# Rozpoznawanie krawędzi

## Problem!

Dla niektórych puzzli "główki" są mocno wysunięte, co prowadzi do błędnego rozpoznania prawidłowego narożnika

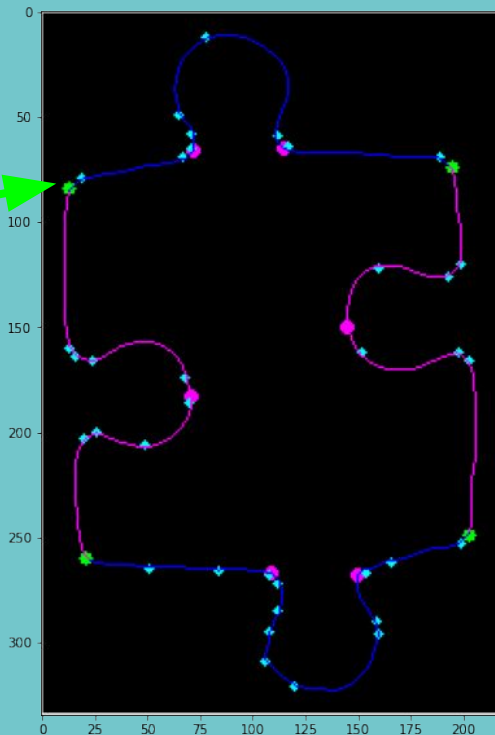




# Rozpoznawanie krawędzi

## Rozwiązanie

Rozpoznanie "główek" i ignorowanie potencjalnych narożników leżących na jej konturze.



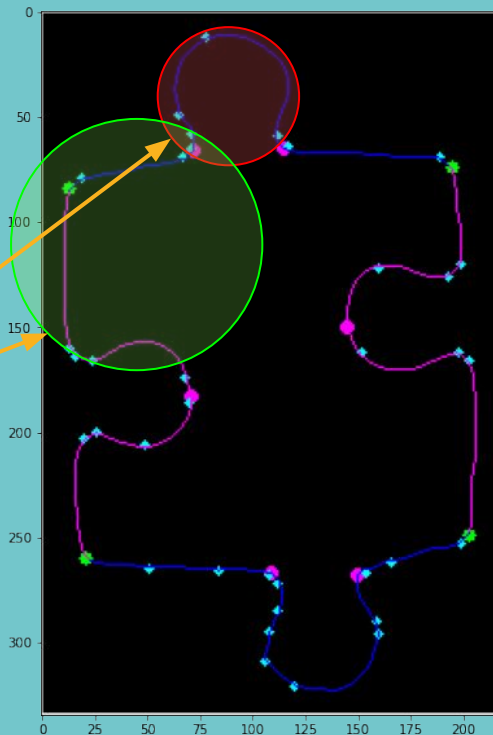


# Rozpoznawanie krawędzi

## Rozwiązanie

Rozpoznanie “główek” i ignorowanie potencjalnych narożników leżących na jej konturze.

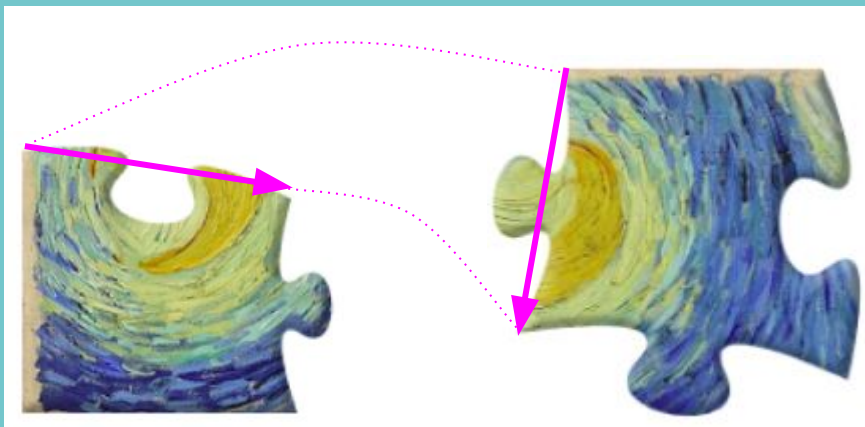
Rozpoznawanie główek bazujące na sprawdzeniu miary “roundness” dla kawałka puzzla pomiędzy punktami wklęsłymi.





# Porównywanie krawędzi geometryczne

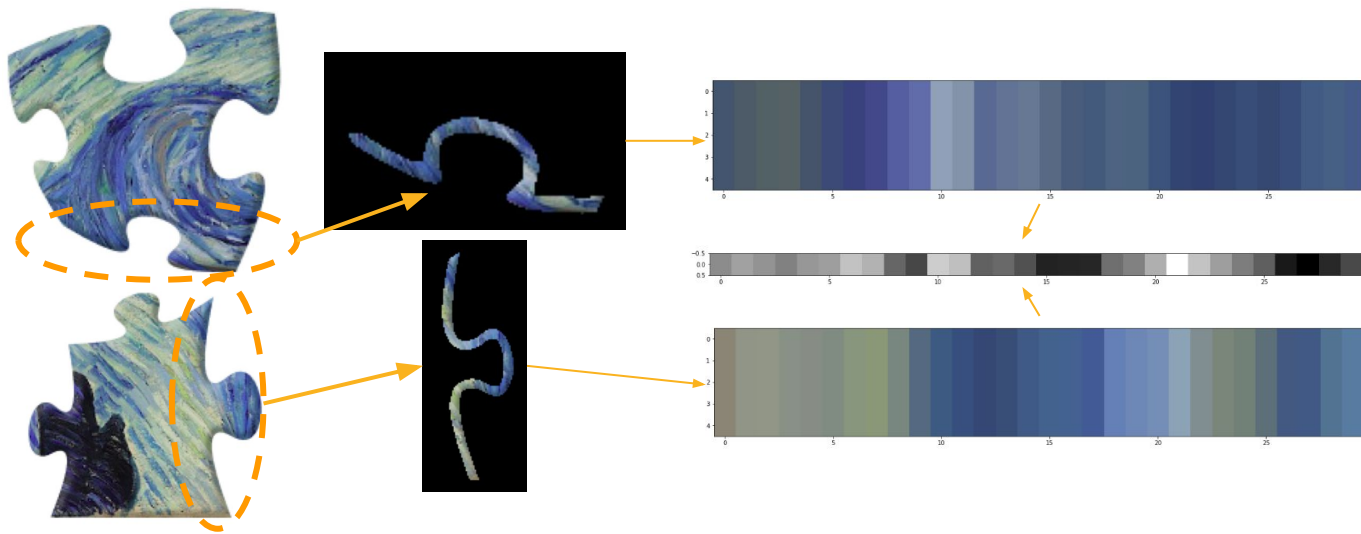
Znalezienie obrotu i przesunięcia  
nakładającego na siebie narożniki, a  
następnie zmierzenie średniej  
odległości punktów pomiędzy  
krawędziami nałożonymi na siebie w  
ten sposób





# Porównywanie krawędzi kolorystyczne

Wyznaczenie wektorów  
kolorystycznych krawędzi puzzli i  
porównanie ich segmentami

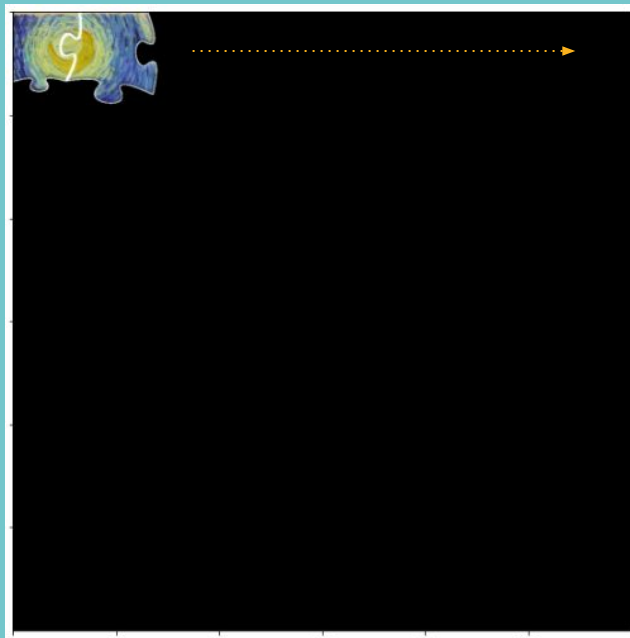




Standardowy algorytm z  
nawrotami

Dla ułatwienia puzzle są  
układane od górnego  
lewego rogu, w prawo

W momencie braku  
"dobrze" pasujących  
puzzli algorytm wycofuje  
się



# Algorytm



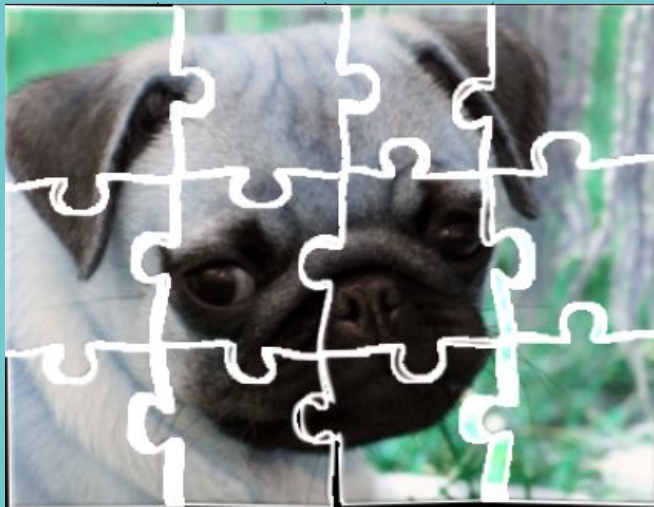
# WYNIKI







# Puzzle wygenerowane sztucznie



**Liczba elementów:** 12

**Liczba nawrotów:**

Odległość punktowa: 13

Odległość kolorystyczna: 21

Odległość kolorystyczna +  
punktowa: 12



# Puzzle wygenerowane sztucznie



**Liczba elementów:** 30

**Liczba nawrotów:**

Odległość punktowa: 0

Odległość kolorystyczna: brak  
rozwiązania

Odległość kolorystyczna +  
punktowa: 0





# Puzzle prawdziwe

Liczba elementów: 54

Liczba prawidłowo  
ustawionych puzzli:

Odległość punktowa: brak  
rozwiązania

Odległość kolorystyczna:  
brak rozwiązania

Odległość kolorystyczna +  
punktowa: 0





# Puzzle prawdziwe

Liczba elementów: 20

Liczba prawidłowo  
ustawionych puzzli:

Odległość punktowa: brak  
rozwiązania

Odległość kolorystyczna:  
brak rozwiązania

Odległość kolorystyczna +  
punktowa: 0



# Problemy i wnioski


- Duża liczba parametrów do dostosowania dla konkretnego zestawu puzzli
- Niedoskonałość algorytmu
- Problem w kolorystycznym porównywaniu krawędzi

A large orange square in the top-left corner and a smaller orange circle below it, both partially cut off by the left edge of the slide.

# Dziękujemy

A horizontal row of eight small red dots centered under the word 'Dziękujemy'.

CREDITS: This presentation template was created by [Slidesgo](#), including icons by [Flaticon](#), and infographics & images by [Freepik](#) and illustrations by [Stories](#)

A large red square in the bottom-right corner and a smaller red circle above it, both partially cut off by the right edge of the slide.