

# Sprawozdanie 2.

## Laboratoria 5., 6. i 7.

Mirosław Kołodziej

26.11.2021

### 1. Laboratoria 5.

Celem naszych zajęć laboratoryjnych było zapoznanie się z operacjami morfologicznymi przeprowadzanymi na obrazach. Używaliśmy poniższego obrazka:



*Rysunek 1. Oryginalna wersja obrazka wykorzystywanego w trakcie zajęć*

Początkowo naszym zadaniem było oddzielenie kaczek znajdujących się na obrazku od tła (nieba). Po przejściu na skalę szarości wykonaliśmy na naszym obrazie dwie operacje – binaryzację oraz zamknięcie. Otrzymaliśmy następujący efekt:



*Rysunek 2. Obrazek po binaryzacji i operacji zamknięcia*

Następnie poznaliśmy polecenie `bwmorph` oraz jego opcje. Omówiliśmy je w następującej kolejności:

- `bwmorph(im, 'erode', 5)` – polecenie, które wykonuje na obrazie erozję (poznaną na jednym z poprzednich zajęć) w pięciu powtórzeniach. Otrzymaliśmy następujący efekt:



*Rysunek 3. Obrazek po pięciokrotnym wykonaniu erozji*

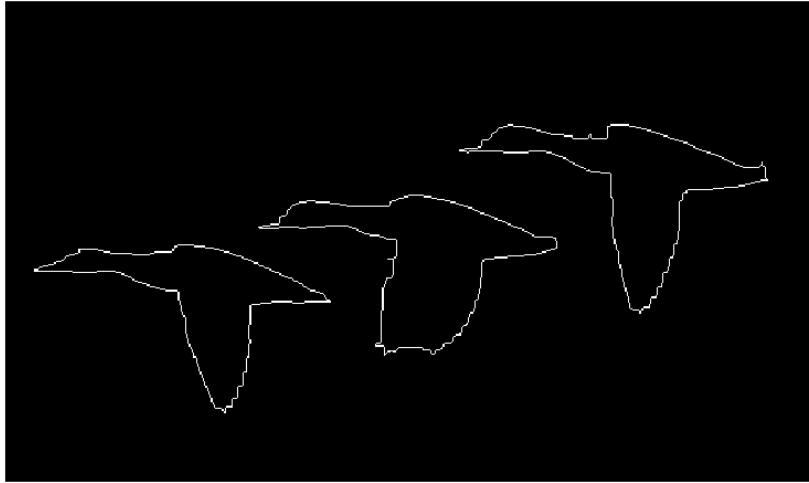
- `bwmorph(im, 'open')` – polecenie, które wykonuje na obrazku operację otwarcia,
- `bwmorph(im, 'fill')` – polecenie oczyszczające obrazek z pojedynczych czarnych pikseli,



*Rysunki 3., 4. Fragment obrazka przed wykonaniem operacji zamknięcia oraz ten sam obrazek po wykonaniu operacji `fill`*

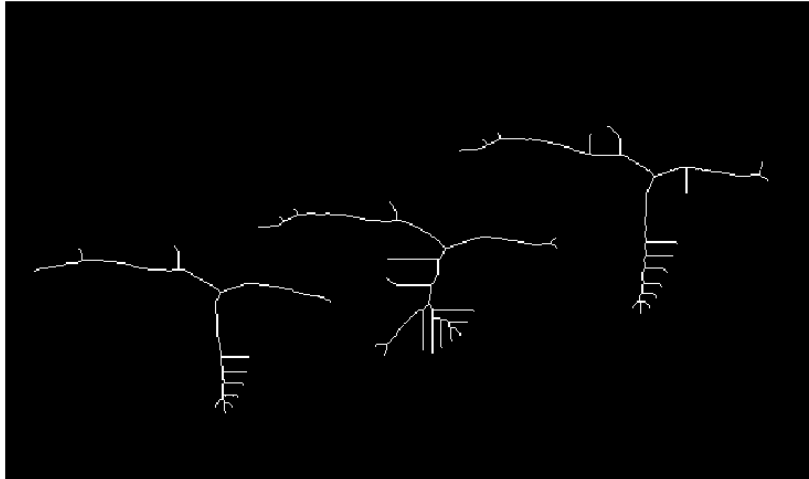
- `bwmorph(im, 'clean')` – polecenie odwrotne do `fill`,

- `bwmorph(im, 'remove')` – polecenie usuwające środek obiektu, zostawia tylko jego krawędzie zewnętrzne:



*Rysunek 5. Obrazek po wykonaniu operacji `remove`*

- `bwmorph(im, 'skel', inf)` – polecenie wykonujące szkieletyzację, czyli wyznaczenie punktów równoodległych od co najmniej dwóch krawędzi,



*Rysunek 6. Obrazek po wykonaniu operacji `skel` z maksymalną liczbą powtórzeń*



*Rysunek 7. Obrazek po wykonaniu dziesięciu powtórzeń operacji `skel`*

- `bwmorph(skel, 'endpoints')` – operacja służąca do uzyskania punktów krańcowych krawędzi, wykonywana po szkieletyzacji:



*Rysunek 8. Obrazek po wykonaniu operacji `endpoints`*

- `bwmorph(skel, 'branchpoints')` – operacja służąca do uzyskania punktów będących miejscami powstawania rozgałęzień szkieletu, wykonywana po szkieletyzacji:

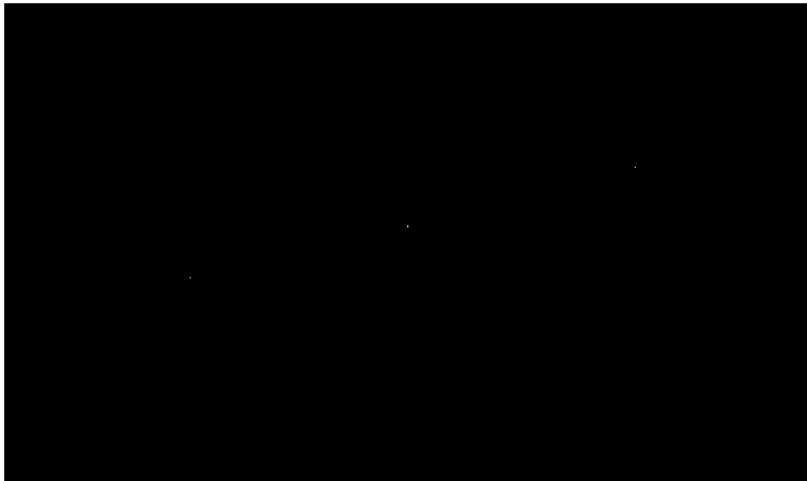


*Rysunek 9. Obrazek po wykonaniu operacji `branchpoints`*

- `bwmorph(im, 'shrink')` – operacja zwężająca obiekty. Dla parametru `inf` zwęża obiekty do pojedynczych punktów



*Rysunek 10. Obrazek po wykonaniu trzech powtórzeń operacji `shrink`*

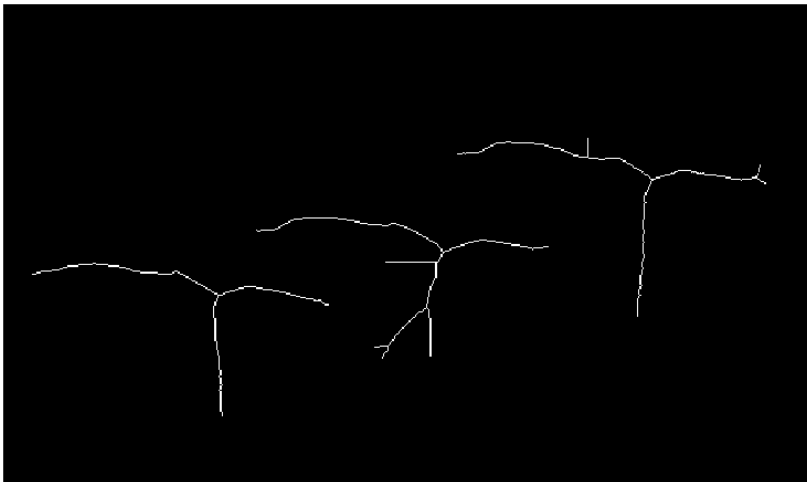


*Rysunek 11. Obrazek po wykonaniu maksymalnej ilości powtórzeń operacji shrink*

- `bwmorph(im, 'thin')` – operacja redukująca obiekty do linii. Daje podobny efekt do operacji skel



*Rysunek 12. Obrazek po wykonaniu trzech powtórzeń operacji thin*



*Rysunek 13. Obrazek po wykonaniu maksymalnej ilości powtórzeń operacji thin*

- `bwmorph(im, 'thicken', inf)` – operacja przeciwna do `thin`, podobna do dylatacji. Wynikiem jej działania są pogrubione obiekty z zachowaniem podziału między poszczególnymi obiektami



*Rysunek 14. Obrazek po wykonaniu dziesięciu powtórzeń operacji `thicken`*



*Rysunek 15. Obrazek po wykonaniu maksymalnej ilości powtórzeń operacji `thicken`*

Po omówieniu opcji polecenia `bwmorph` poznaliśmy polecenie `bwlabel`, które pozwala na nadanie etykiety poszczególnym obiektom. Polecenie to umożliwia nam dokonanie segmentacji.



*Rysunek 16. Efekt działania polecenia `bwlabel`*

Następnie wyodrębniliśmy jedną kaczkę i wyświetliliśmy ją na ekranie:



*Rysunek 17. Wyodrębniony segment z etykietą nr 3 za pomocą `bwLabel`*

Łącząc poprzednią operację z wynikiem operacji `thicken` możemy wyświetlić kaczkę wraz z jej otoczeniem.



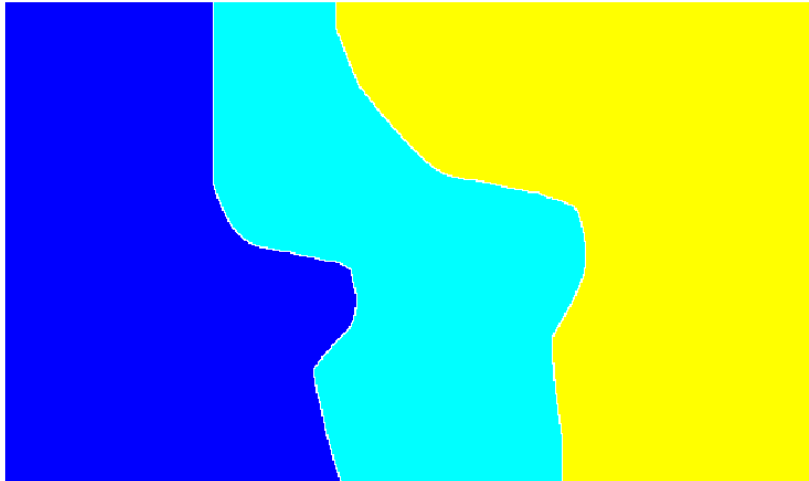
*Rysunek 18. Wyodrębniony segment z etykietą nr 2 wraz z otoczeniem*

Kolejną wykonaną przez nas operacją była transformata odległościowa. Posłużyło nam do niej polecenie `bwdist`, które zwraca odległość każdego piksela od najbliższego obiektu.



*Rysunek 19. Wynik polecenia `bwdist`*

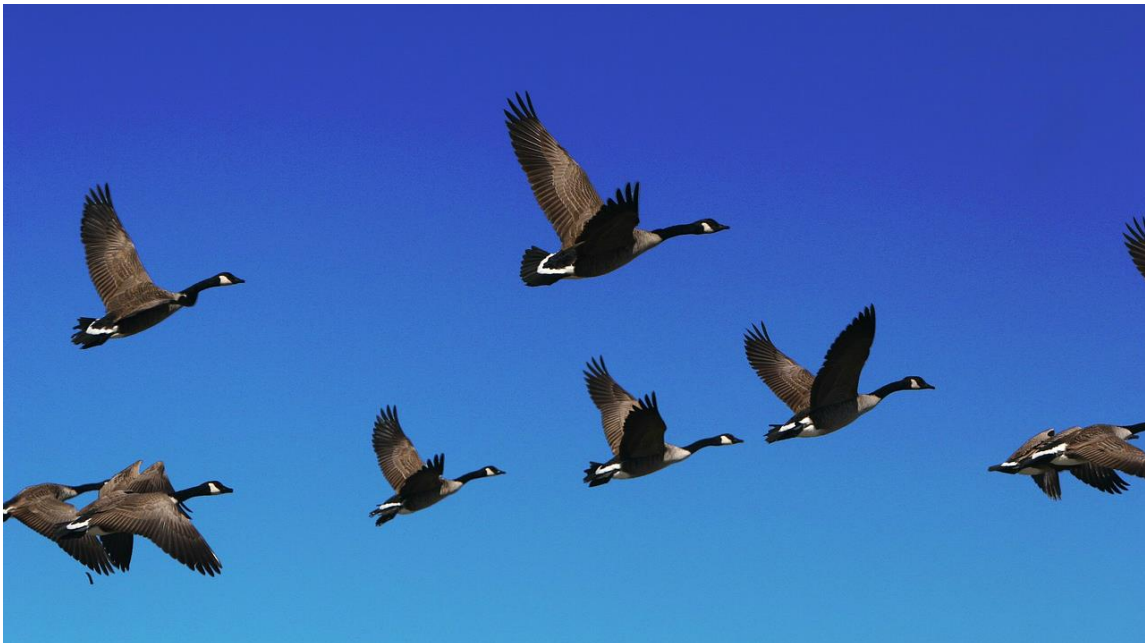
Ostatnią wykonaną przez nas operacją była segmentacja wododziałowa. Skorzystaliśmy z wcześniej otrzymanego obrazka oraz polecenia watershed.



*Rysunek 19. Segmentacja wododziałowa*

## 2. Laboratoria 6.

Na kolejnych zajęciach laboratoryjnych poznaliśmy współczynniki geometryczne obrazu. Tym razem korzystaliśmy z poniższego obrazka:



*Rysunek 20. Oryginalna wersja obrazka wykorzystywanego w trakcie zajęć*

Początkowo próbowaliśmy przekształcić obrazek w binarny podobnymi metodami jak na poprzednich zajęciach, lecz okazały się nieskuteczne dla tego obrazka. Nie potrafiliśmy za ich pomocą oddzielić ptaków od tła.

Dowiedzieliśmy się, że lepszy rezultat możemy osiągnąć używając poszczególnych warstw RGB. Zignorowaliśmy warstwę niebieską, zaś zaprzeczenie czerwonej oraz zielonej połączyliśmy za pomocą funkcji logicznej lub. Otrzymaliśmy następujący efekt:





*Rysunek 21. Binarizacja przy użyciu kanałów R i B*

Następnie poznaliśmy funkcję `regionprops`, która po przekazaniu do niej obrazu binarnego zwraca jego cechy, takie jak:

- `Area` – ilość pikseli w danym regionie,
- `Centroid` – punkt centralny obiektu ,
- `BoundingBox` – określa rozmiar i położenie najmniejszego prostokąta (zgodnego z układem odniesienia) w którym można umieścić obiekt,
- `MajorAxisLength`, `MinorAxisLength` – długość głównej i mniejszej osi elipsy przybliżającej krawędzie obiektu,
- `Eccentricity` – środek masy obiektu
- `Orientation` – kąt pomiędzy osią X układu a główną osią,
- `Circularity` – miara podobieństwa obiektu do koła. Dla samego koła wartość ta wynosi 1,
- `Image` – macierz obrazu przyciętego do `BoundingBox`,



*Rysunek 22. Image dla jednego z obiektów*

- `EulerNumber` – wartość liczby obiektów (nie licząc ilości otworów w tych obiektach),
- `Perimeter` – obwód obiektu.

Później poznaliśmy współczynniki geometryczne, które określały w różny sposób podobieństwo danego obiektu do koła:

- współczynnik Blair-Bliss'a – stosunek średniej odległości pikseli od środka obiektu do średniej odległości pikseli od środka koła o tym samym polu,
- współczynnik Danielsson'a – stosunek średniej odległości pikseli od krawędzi obiektu do średniej odległości pikseli od krawędzi koła o tym samym polu,

- współczynnik Feret'a – stosunek rozpiętości w poziomie i pionie,
- współczynnik Haralick'a – stosunek średniej odległości pikseli na krawędzi od środka obiektu do średniej odległości pikseli na krawędzi od środka koła o tym samym polu,
- współczynnik Malinowskiej – stosunek obwodu obiektu do pierwiastka z  $4\pi S$  pomniejszonego o 1, gdzie  $S$  jest powierzchnią badanego obiektu.
- CircularityS – promień koła o tym samym polu co obiekt,
- CircularityL – promień koła o tym samym obwodzie co obiekt,
- Shape – odwrotność współczynnika Circularity.

Następnie wyznaczyliśmy wszystkie powyższe współczynniki dla każdego z obiektów i zapisaliśmy je w macierzy.

Na sam koniec zajęć, wspomagając się współczynnikami oddzieliliśmy nietypowe obiekty, dla których wartości tych współczynników odbiegały od reszty o wartość  $1,8\sigma$ .

### 3. Laboratoria 7.

Na ostatnich zajęciach laboratoryjnych zajmowaliśmy się klasyfikacją obiektów przy pomocy sieci neuronowych. Korzystaliśmy ponownie z obrazka wykorzystywanego na poprzednich zajęciach oraz poniższego zdjęcia:



*Rysunek 23. Oryginalna wersja zdjęcia*

Rozpoczęliśmy ponownie od binaryzacji zdjęcia. Tym razem skorzystaliśmy tylko z kanału niebieskiego. Otrzymaliśmy następujący efekt:



*Rysunek 24. Obrazek po binaryzacji*

Następnie powtórzyliśmy operację z poprzednich zajęć – wyznaczyliśmy macierz współczynników określających podobieństwo obiektów do koła. Kolejnym krokiem było oddzielenie obiektów, które są przycięte przez to, że leżą na brzegach. Ustaliliśmy, że będą to obiekty o rozmiarze poniżej 700 pikseli.

Na koniec poznaliśmy pojęcie sieci neuronowych i ich działanie. Przetworzyliśmy za ich pomocą trzy zestawy danych – współczynniki uzyskane w trakcie poprzednich zajęć, współczynniki nietypowych obiektów z obecnych zajęć oraz współczynniki pozostałych obiektów.

### Neural Network

### Algorithms

Data Division: Random (dividerand)  
 Training: Levenberg-Marquardt (trainlm)  
 Performance: Mean Squared Error (mse)  
 Calculations: MEX

### Progress

Epoch:	0	8 iterations	1000
Time:		0:00:01	
Performance:	0.681	2.79e-19	0.00
Gradient:	0.851	3.59e-10	1.00e-07
Mu:	0.00100	1.00e-11	1.00e+10
Validation Checks:	0	4	6

### Plots

Performance

(plotperform)

Training State

(plottrainstate)

Error Histogram

(ploterrhist)

Regression

(plotregression)

Plot Interval:

1 epochs

✓

Opening Performance Plot

Stop Training

Cancel

Rysunek 24. Okno sieci neuronowej