



### 1. Binarizacja.

**Segmentacja** to wyodrębnianie obiektów z tła. Segmentacja może wyróżniać obiekty, na przykład na podstawie: **konturów**, **barwy**, **typu tekstury** (faktury) powierzchni. **Binarizacja** jest rodzajem segmentacji wykonywanej na podstawie **jasności pikseli** obrazów szarych. Binarizacja polega na przetwarzaniu obrazów szarych na czarno-białe i stosuje się ją do redukowania ilości informacji nieistotnych z punktu widzenia analizy obrazu. Przekształcenie obrazu szarego w obraz binarny wiąże się z ustaleniem pewnego algorytmu przypisania punktom obrazu szarego wartości 0 lub 1.

Ze względu na zastosowany algorytm, wyróżniamy następujące typy binaryzacji:

- o binaryzacja z dolnym progiem,
- o binaryzacja z górnym progiem,
- o binaryzacja z dwoma progami,
- o binaryzacja wielokryterialna,
- o binaryzacja z histerezą.

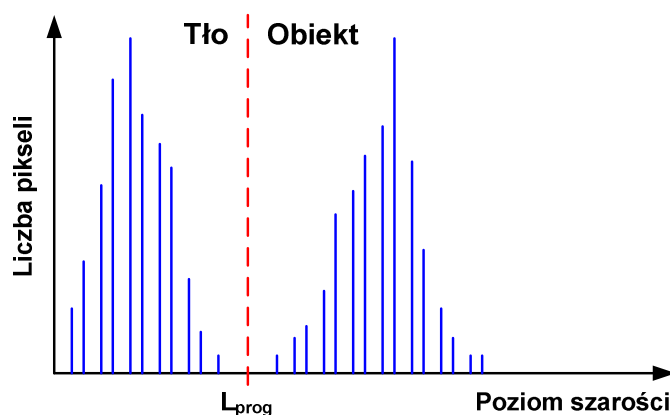
Funkcję binaryzacyjną z **dolnym progiem** ( $L_{pd}$ ) można zapisać w postaci:

$$B(w, k) = \begin{cases} 0 & \text{dla } A(w, k) \leq L_{pd} \\ 1 & \text{dla } A(w, k) > L_{pd} \end{cases}$$

W binaryzacji z dolnym progiem, wszystkie punkty o stopniu szarości poniżej progu stają się czarne (0), a powyżej niego białe (1). W Matlabie, binaryzację z dolnym progiem obrazu szarego **A** na czarno-biały **B** realizuje polecenie:

```
>> B = A > Lpd;
```

Wynik przetwarzania będzie zależał od wartości przyjętego progu, który można dobrać na podstawie obserwacji histogramu obrazu. Jeśli ów histogram ma "dolinę", to jej położenie określa granicę między jasnymi i ciemnymi obszarami i bywa przyjmowane właśnie jako wartość **progu binaryzacji** (rys. 1).



Rys. 1. Określanie wartości progu binaryzacji dla przykładowego obrazu

Funkcję binaryzacyjną z **górnym progiem** ( $L_{pg}$ ) można zapisać w postaci:

$$B(w, k) = \begin{cases} 0 & \text{dla } A(w, k) \geq L_{pg} \\ 1 & \text{dla } A(w, k) < L_{pg} \end{cases}$$

W binaryzacji z górnym progiem, wszystkie punkty o stopniu szarości poniżej progu stają się białe (1), a powyżej niego czarne (0). W Matlabie, binaryzację z górnym progiem obrazu szarego **A** na czarno-biały **B** realizuje polecenie:

```
>> B = A < Lpg;
```



Funkcję binaryzacyjną z dwoma progami ( $L_{pd}$ ,  $L_{pg}$ ) można zapisać w postaci:

$$B(w, k) = \begin{cases} 0 & \text{dla } A(w, k) \leq L_{pd} \\ 1 & \text{dla } L_{pd} < A(w, k) \leq L_{pg} \\ 0 & \text{dla } A(w, k) > L_{pg} \end{cases}$$

W binaryzacji z dwoma progami, wszystkie punkty o stopniu szarości pomiędzy progami stają się białe (1), a pozostałe stają się czarne (0). W Matlabie, binaryzację z dwoma progami obrazu szarego **A** na czarno-biały **B** realizuje polecenie:

```
>> B = (A > Lpd) & (A <= Lpg);
```

### ♦ Ćwiczenie 1.

Plik graficzny **czerniak.jpg** przedstawia zdjęcie czerniaka złośliwego skóry. Na podstawie tego obrazu wykonaj poniższe operacje.

- Dokonaj konwersji obrazu do skali szarości.
- Wyświetl histogram obrazu szarego i na jego podstawie dobierz wartość progu binaryzacji.
- Dokonaj binaryzacji obrazu z dolnym progami. W razie potrzeby skoryguj nieco wartość progu tak, aby uzyskać wyraźne granice guza.
- Zapisz obraz binarny pod nazwą **czerniakBW.bmp**.

## 2. Funkcje przetwarzania morfologicznego obrazów.

Pojęcie **morfologia** oznacza, m.in. badanie kształtu i struktury. W przetwarzaniu obrazów wykorzystuje się morfologię matematyczną. Jest to dział teorii zbiorów wykorzystywany, m.in. do analizy cech geometrycznych wyróżnionych obiektów obrazu. Metody morfologii matematycznej pozwalają rozpoznawać budowę obiektów, a także przetwarzać ich kształt poprzez analizę badanego obrazu za pomocą specjalnych obiektów nazywanych elementami strukturalnymi.

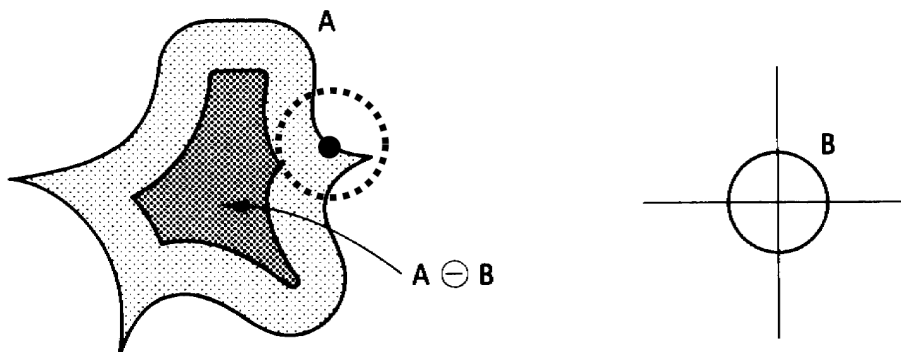
W dalszej części podano definicje popularnych działań morfologicznych, które są podstawą bardziej złożonych operacji. W działaniach tych przyjmujemy, że zbiór **A** jest obiektem przetwarzanym, a zbiór **B** jest elementem **strukturalnym**, za pomocą którego wykonujemy poszczególne operacje morfologiczne. W obrazach binarnych, przez **obiekt** rozumie się spójny zbiór punktów o wartościach **HIGH**. Wartości **HIGH** są równe 1 dla tablic obrazów klasy **double** lub 255 dla klasy **uint8**.

## 3. Erozja.

Niech **A** i **B** będą zbiorami w  $\mathbb{R}^2$ . **Erozja A przez B** (gdzie **B** jest elementem strukturalnym) jest działaniem postaci:

$$A \ominus B = \bigcap_{b \in B} (A + b)$$

Czyli  $A \ominus B$  jest wspólną częścią wszystkich translacji **A** przez wszystkie elementy **B** o wektor **b** będący elementem zbioru **B**, dla każdego wektora  $b \in B$  (rys. 2).



Rys. 2. Erozja zbioru **A** elementem **B**



### ♦ Ćwiczenie 2.

Wykonaj poniższą sekwencję poleceń ilustrujących działanie erozji, a następnie porównaj obrazy binarne **BW1** oraz **BW2**.

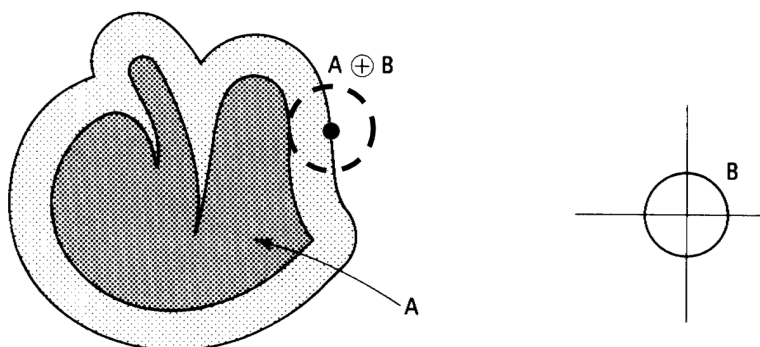
```
BW1 = imread('kontur.bmp');  
SE = ones(4, 4);  
BW2 = imerode(BW1, SE);  
imshow(BW1)  
figure, imshow(BW2)
```

### 4. Dylatacja.

Niech **A** i **B** będą zbiorami w  $\mathbb{R}^2$ . **Dylatacja A przez B** (gdzie **B** jest elementem strukturalnym) jest działaniem postaci:

$$A \oplus B = \bigcup_{b \in B} (A + b)$$

Innymi słowy,  $A \oplus B$  jest wynikiem sumy zbiorów powstałych w wyniku translacji zbioru **A** o wektor **b** będący elementem zbioru **B**, dla każdego wektora  $b \in B$  (rys. 3).



Rys. 3. Dylatacja zbioru **A** elementem **B**

### ♦ Ćwiczenie 3.

Wykonaj poniższą sekwencję poleceń ilustrujących działanie dylatacji, a następnie porównaj obrazy binarne **BW1** oraz **BW2**.

```
BW1 = imread('kontur.bmp');  
SE = ones(4, 4);  
BW2 = imdilate(BW1, SE);  
imshow(BW1)  
figure, imshow(BW2)
```

### 5. Otwieranie obiektu.

**Otwieranie obiektu A elementem B** definiuje działanie morfologiczne:

$$A \circ B = (A \ominus B) \oplus B,$$

**czyli** jest to ciąg operacji, w którym najpierw wykonuje się **erozję** obiektu **A** elementem strukturalnym **B**, a następnie uzyskany zbiór poddaje się **dylatacji** za pomocą tego samego elementu strukturalnego (rys. 4). W wyniku zastosowania operacji otwierania uzyskuje się **wygładzenie krawędzi** i **usuwanie przewężeń** obiektu (np. otwieranie zamkniętych konturów).

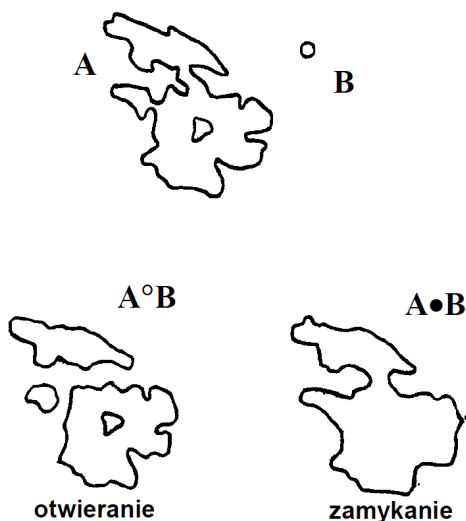
### ♦ Ćwiczenie 4.

Wykonaj poniższą sekwencję poleceń ilustrujących działanie otwierania obiektu, a następnie porównaj obrazy binarne **BW1** oraz **BW3**.

```
BW1 = imread('powierzchnia.bmp');  
SE = ones(20, 20);  
BW2 = imerode(BW1, SE);
```



```
BW3 = imdilate(BW2, SE);  
imshow(BW1)  
figure, imshow(BW3)
```



Rys. 4. Ilustracja działania operacji morfologicznych otwierania i zamykania

### 6. Zamykanie obiektu.

Zamykanie obiektu **A** elementem **B** definiuje działanie morfologiczne:

$$A \bullet B = (A \oplus B) \ominus B,$$

czyli jest to ciąg operacji, w którym najpierw wykonuje się **dylatację** obiektu **A** elementem **B**, a następnie uzyskany zbiór poddaje się **erozji** za pomocą tego samego elementu strukturalnego. Operacja zamykania **wygładza kontur** obiektu oraz **zamyka wąskie wgłębienia** lub **otwory** w obiekcie (rys. 4).

#### ♦ Ćwiczenie 5.

Wykonaj poniższą sekwencję poleceń ilustrujących zamykanie otwierania obiektu, a następnie porównaj obrazy binarne **BW1** oraz **BW3**.

```
BW1 = imread('powierzchnia.bmp');  
SE = ones(50, 50);  
BW2 = imdilate(BW1, SE);  
BW3 = imerode(BW2, SE);  
imshow(BW1)  
figure, imshow(BW3)
```

### 7. Inne funkcje do przetwarzania obrazów binarnych z biblioteki IPT.

W bibliotece IPT znajduje się funkcja **bwmorph**, za pomocą której można wykonywać liczne operacje morfologiczne na obrazach binarnych, np. erozję, dylatację, otwieranie, zamykanie, wyznaczanie szkieletu obiektu, itp. Oprócz **bwmorph**, w bibliotece IPT znajduje się także wiele innych funkcji do przetwarzania obrazów binarnych, należą do nich, m.in.:

- o **bwperim** – wyznaczanie brzegu obiektu w obrazie;
- o **bwfill** – wypełnianie zamkniętych konturów;
- o **bwlabel** – etykietowanie obiektów;

#### ♦ Ćwiczenie 6.

Wykonaj poniższą sekwencję poleceń ilustrujących działanie funkcji **bwperim**, następnie oceń wyniki przetwarzania.

```
A = imread('powierzchnia.bmp');  
imshow(A);
```



```
B = bwperim(A, 8);  
figure, imshow(B)
```

### ♦ Ćwiczenie 7.

Wykonaj poniższą sekwencję poleceń ilustrujących działanie funkcji `bwfill`, następnie oceń wyniki przetwarzania.

```
I = imread('czerniakBW.bmp');  
BW3 = ~im2bw(I);  
BW4 = bwfill(BW3, 'holes');  
imshow(BW3)  
figure, imshow(BW4)
```

### ♦ Ćwiczenie 8.

Wykonaj poniższą sekwencję poleceń ilustrujących działanie funkcji `bwlabel`, następnie oceń wyniki przetwarzania.

```
BW = [1 1 1 0 0 0 0 0  
1 1 1 0 1 1 0 0  
1 1 1 0 1 1 0 0  
1 1 1 0 0 0 1 0  
1 1 1 0 0 0 1 0  
1 1 1 0 0 0 1 0  
1 1 1 0 0 1 1 0  
1 1 1 0 0 0 0 0];  
L = bwlabel(BW, 4)
```

## 8. Zadania

### Zadanie 1.

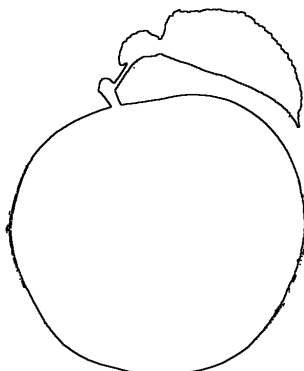
Wykonaj wybrane operacje morfologiczne na obrazie `kontur.bmp`. Zastosuj funkcję `bwmorph` z opcjami: `remove`, `thin` i `shrink`.

### Zadanie 2.

Wykorzystując obraz `jablko.bmp`, wykonaj odpowiednie operacje tak, aby otrzymać czarny obiekt jabłka na białym tle.

### Zadanie 3.

Na podstawie obrazu uzyskanego w zadaniu nr 2, uzyskaj wyraźny kontur obiektu (rys. 5).



Rys. 5. Wynik przetwarzania uzyskany w zadaniu nr 3

### Zadanie 4.

Przeprowadź binaryzację obrazu `czaple.bmp`. Po wydzieleniu obiektów prezentujących czaple, zastosuj odpowiednią metodę, w celu wygładzenia ich konturów.



### Zadanie 5.

Zastosuj binaryzację z dwoma progami dla obrazu [okret.jpg](#). Dobierz wartości progów tak, aby z jednej strony uzyskać jak najlepszą jakość obiektu przedstawiającego okręt, z drugiej zaś - minimalną powierzchnię obszarów odpowiadających niebu i wodzie.

### Zadanie 6.

Dokonaj binaryzacji obrazu [tygrys.jpeg](#). W celu usunięcia z obrazu izolowanych pikseli, zastosuj funkcję [bwmorph](#) z opcją [majority](#). Uzyskany efekt powinien być zbliżony do przedstawionego na rys. 6.



Rys. 6. Wynik przetwarzania uzyskany w zadaniu nr 6

### Zadanie 7.

Przeprowadź operacje przetwarzania obrazu otrzymanego w zadaniu nr 5, aby uzyskany efekt był taki, jak na rys. 7.



Rys. 7. Wynik przetwarzania uzyskany w zadaniu nr 7