

# **Modelowanie i Wizualizowanie 3W grafiki. Rzutowanie i rasteryzacja**

Aleksander Denisiuk  
Uniwersytet Warmińsko-Mazurski  
Olsztyn, ul. Słoneczna 54  
[denisjuk@matman.uwm.edu.pl](mailto:denisjuk@matman.uwm.edu.pl)

# Rzutowanie i rasteryzacja

Rzutowanie

Rasteryzacja

Najnowsza wersja tego dokumentu dostępna jest pod adresem

<http://wmii.uwm.edu.pl/~denisjuk/uwm>

## Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- Rzutowanie równoległe
- Rzutowanie perspektywiczne
- Macierze w modelowaniu

## Rasteryzacja

# Rzutowanie

#### Rzutowanie

- Dwa typy rzutowania

- Ukryte powierzchnie
- Rzutowanie równoległe
- Rzutowanie perspektywiczne
- Macierze w modelowaniu

#### Rasteryzacja

## Dwa typy rzutowania

- Równoległe
- Perspektywiczne

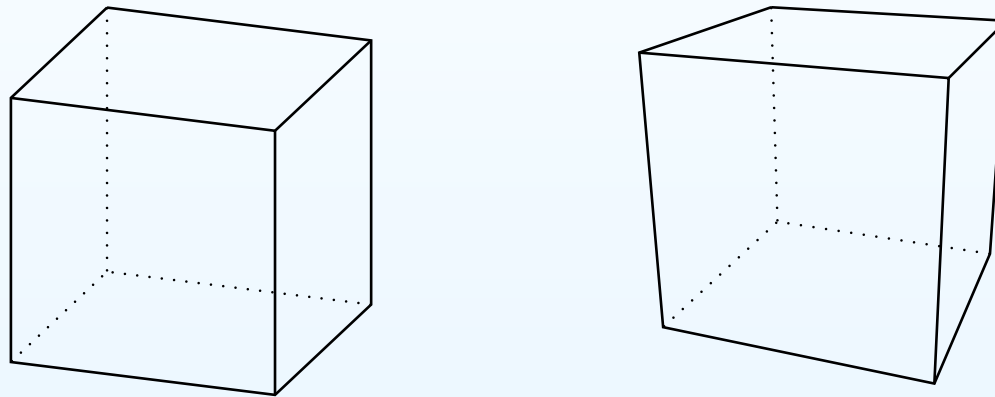


Figure II.18: The cube on the left is rendered with an orthographic projection. The one on the right with a perspective transformation. With the orthographic projection, the rendered size of a face of the cube is independent of its distance from the viewer; compare, for example, the front and back faces. Under a perspective transformation, the closer a face is, the larger it is rendered.

#### Rzutowanie

- Dwa typy rzutowania
- **Ukryte powierzchnie**
- Rzutowanie równoległe
- Rzutowanie perspektywiczne
- Macierze w modelowaniu

#### Rasteryzacja

## Ukryte powierzchnie

- Algorytm malarza
- Algorytm buforu głębokości

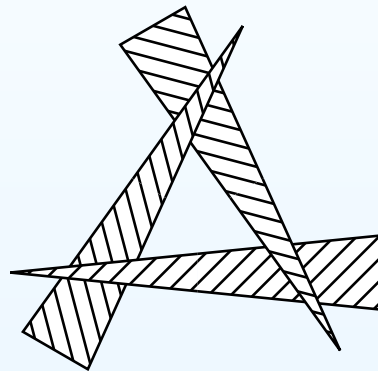


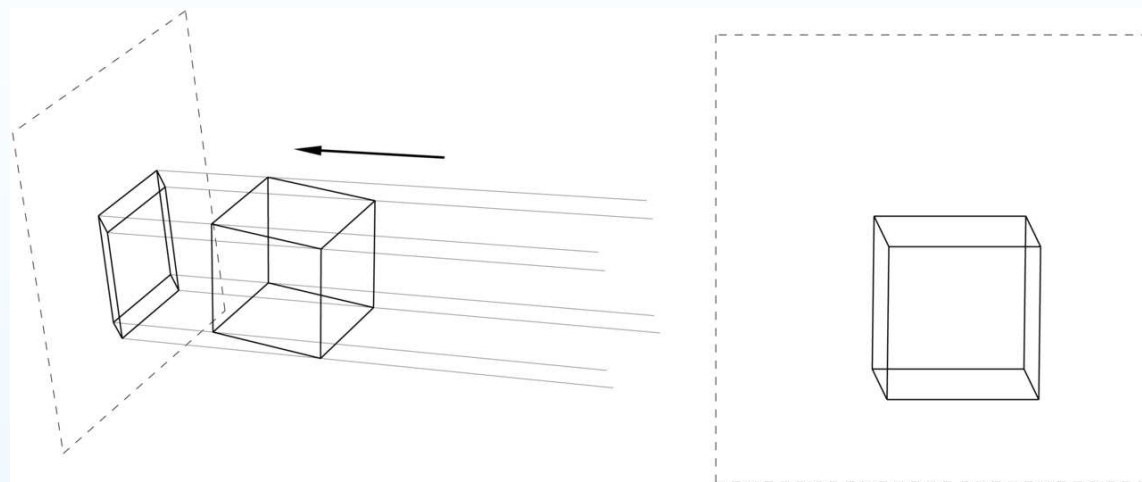
Figure I.12: Three triangles. The triangles are turned obliquely to the viewer so that the top portions of each triangle is in front of the base portion of another.

# Rzutowanie równoległe

## Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- **Rzutowanie równoległe**
- Rzutowanie perspektywiczne
- Macierze w modelowaniu

## Rasteryzacja



# Macierz rzutowania równoległego

## Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- **Rzutowanie równoległe**
- Rzutowanie perspektywiczne
- Macierze w modelowaniu

## Rasteryzacja

$$l \leq x \leq r, \quad \text{left, right}$$

$$b \leq y \leq t, \quad \text{bottom, top}$$

$$n \leq -z \leq f, \quad \text{near, far}$$

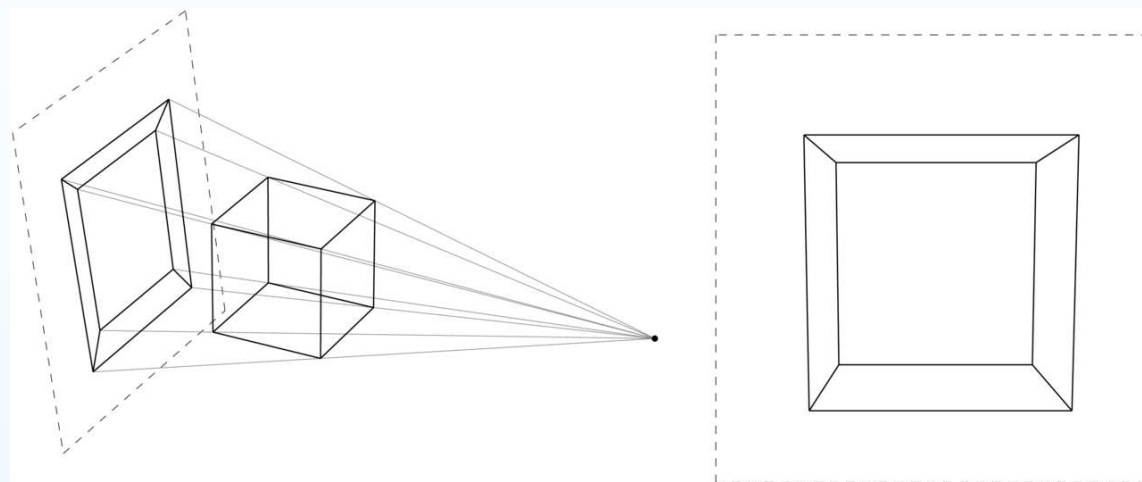
$$\begin{pmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & -\frac{2}{f-n} & -\frac{f+n}{f-n} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

# Rzutowanie perspektywiczne

## Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- Rzutowanie równoległe
- **Rzutowanie perspektywiczne**
- Macierze w modelowaniu

## Rasteryzacja





# Rzutowanie perspektywiczne

## Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- Rzutowanie równoległe
- **Rzutowanie perspektywiczne**
- Macierze w modelowaniu

## Rasteryzacja

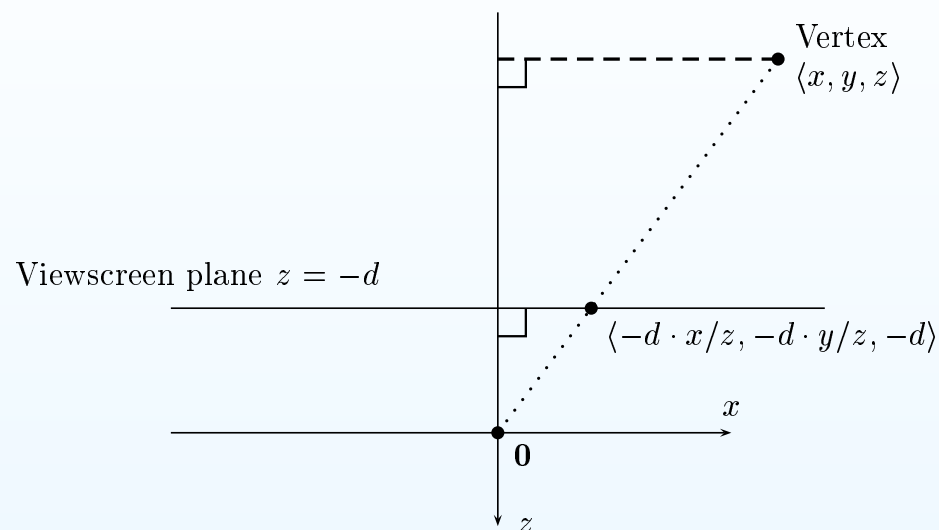


Figure II.19: Perspective projection onto a view screen at distance  $d$ . The viewer is at the origin, looking in the direction of the negative  $z$  axis. The point  $\langle x, y, z \rangle$  is perspective projected onto the plane  $z = -d$ , which is at distance  $d$  in front of the viewer at the origin.

# Funkcja głębokości

## Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- Rzutowanie równoległe
- **Rzutowanie perspektywiczne**
- Macierze w modelowaniu

## Rasteryzacja

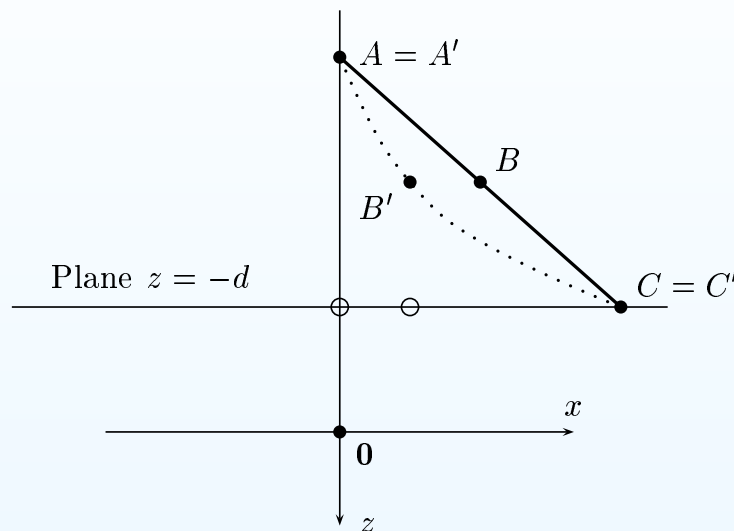


Figure II.20: The undesirable transformation of a line to a curve. The mapping used is  $\langle x, y, z \rangle \mapsto \langle -d \cdot x/z, -d \cdot y/z, z \rangle$ . The points  $A$  and  $C$  are fixed by the transformation and  $B$  is mapped to  $B'$ . The dotted curve is the image of the line segment  $AC$ . (The small unlabeled circles show the images of  $A$  and  $B$  under the mapping of figure II.19.)

- $\text{głębokość}(z) = A + \frac{B}{z}$

#### Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- Rzutowanie równoległe
- **Rzutowanie perspektywiczne**
- Macierze w modelowaniu

#### Rasteryzacja

## Macierz rzutowania

- $(x, y, z) \mapsto \left(-\frac{dx}{z}, -\frac{dy}{z}, A + \frac{B}{z}\right)$
- we współrzędnych jednorodnych  
 $(x : y : z : 1) \mapsto (dx : dy : -Az - B : -z)$
- macierz:

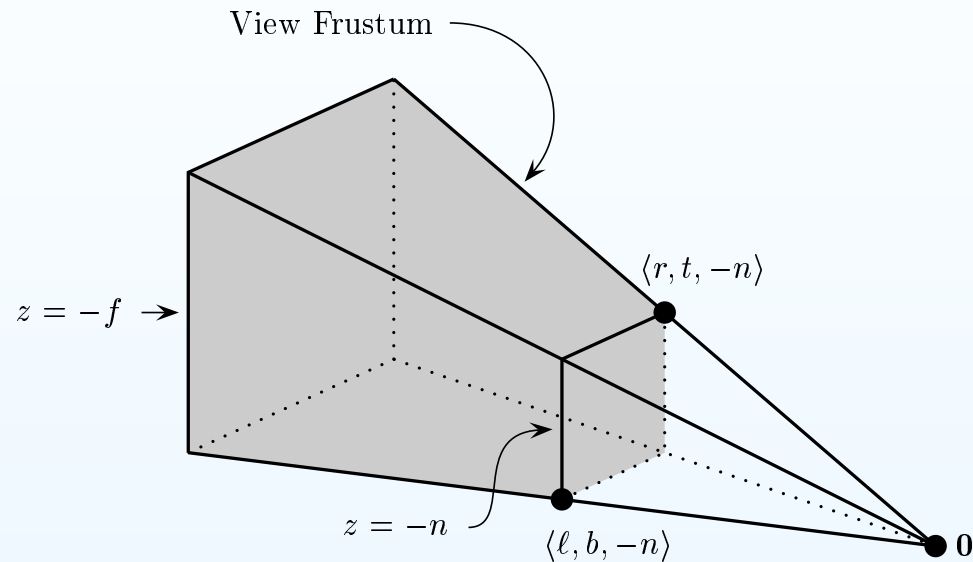
$$\begin{pmatrix} d & 0 & 0 & 0 \\ 0 & d & 0 & 0 \\ 0 & 0 & -A & -B \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

# Bryła widzenia

## Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- Rzutowanie równoległe
- **Rzutowanie perspektywiczne**
- Macierze w modelowaniu

## Rasteryzacja



- głębokość( $z$ ) =  $A + \frac{B}{z}$ , gdzie  $A = \frac{f+n}{f-n}$ ,  $B = \frac{2fn}{f-n}$ .

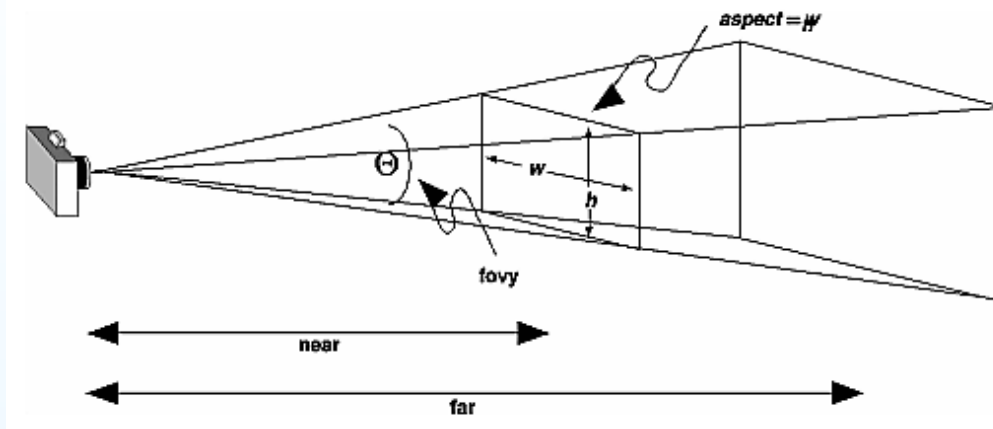
- macierz: 
$$\begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & -\frac{f+n}{f-n} & -\frac{2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

# Macierz rzutowania

## Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- Rzutowanie równoległe
- **Rzutowanie perspektywiczne**
- Macierze w modelowaniu

## Rasteryzacja



- $\theta$  — kąt widzenia
- $a = \frac{w}{h}$  — *aspect ratio*, format obrazu
- macierz:

$$\begin{pmatrix} \frac{1}{a} \operatorname{ctg}(\theta/2) & 0 & 0 & 0 \\ 0 & \operatorname{ctg}(\theta/2) & 0 & 0 \\ 0 & 0 & -\frac{f+n}{f-n} & -\frac{2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

## Zastosowanie rzutowania: cień

### Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- Rzutowanie równoległe
- **Rzutowanie perspektywiczne**
- Macierze w modelowaniu

### Rasteryzacja

$$(x, y, z) \mapsto \left( \frac{x}{1 - y/y_0}, 0, \frac{z}{1 - y/y_0} \right)$$

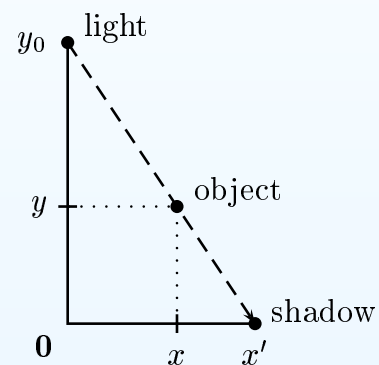


Figure II.22: A light is positioned at  $\langle 0, y_0, 0 \rangle$ . An object is positioned at  $\langle x, y, z \rangle$ . The shadow of the point is projected to the point  $\langle x', 0, z' \rangle$ , where  $x' = x/(1 - y/y_0)$  and  $z' = z/(1 - y/y_0)$ .

# Cień

## Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- Rzutowanie równoległe
- **Rzutowanie perspektywiczne**
- Macierze w modelowaniu

## Rasteryzacja

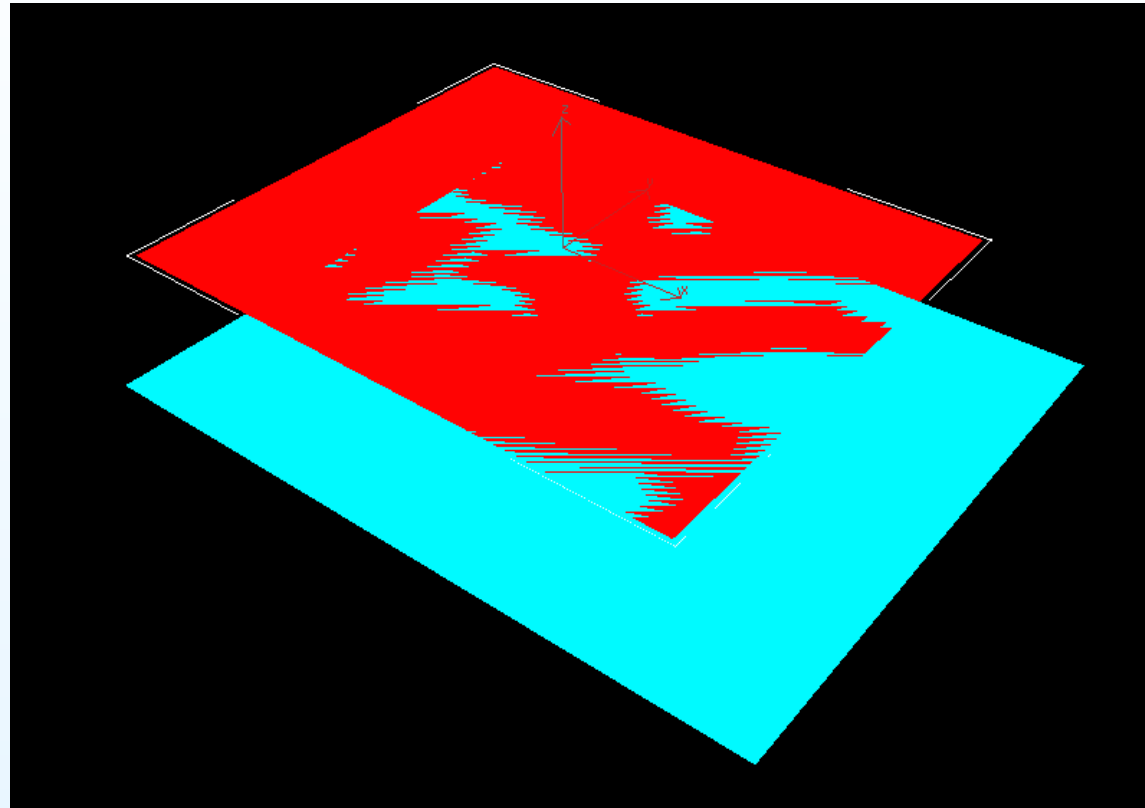
$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -\frac{1}{y_0} & 0 & 1 \end{pmatrix}$$

# Z-fighting

## Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- Rzutowanie równoległe
- **Rzutowanie perspektywiczne**
- Macierze w modelowaniu

## Rasteryzacja





# Z-fighting. Przykład w blenderze

## Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- Rzutowanie równoległe
- Rzutowanie perspektywiczne
- Macierze w modelowaniu

## Rasteryzacja

## Przykład w blenderze

#### Rzutowanie

- Dwa typy rzutowania
- Ukryte powierzchnie
- Rzutowanie równoległe
- Rzutowanie perspektywiczne

#### • **Macierze w modelowaniu**

#### Rasteryzacja

## Macierze w modelowaniu

- Macierz `ModelMatrix`
- Macierz `ViewMatrix`
- Macierz `ProjectionMatrix`

Rzutowanie

## Rasteryzacja

- Rasteryzacja odcinka
- Algorytm Bresenhama
- Interpolacja głębokości, etc.
- Interpolacja w wielokątach
- Błędy zaokrąglenia

# Rasteryzacja

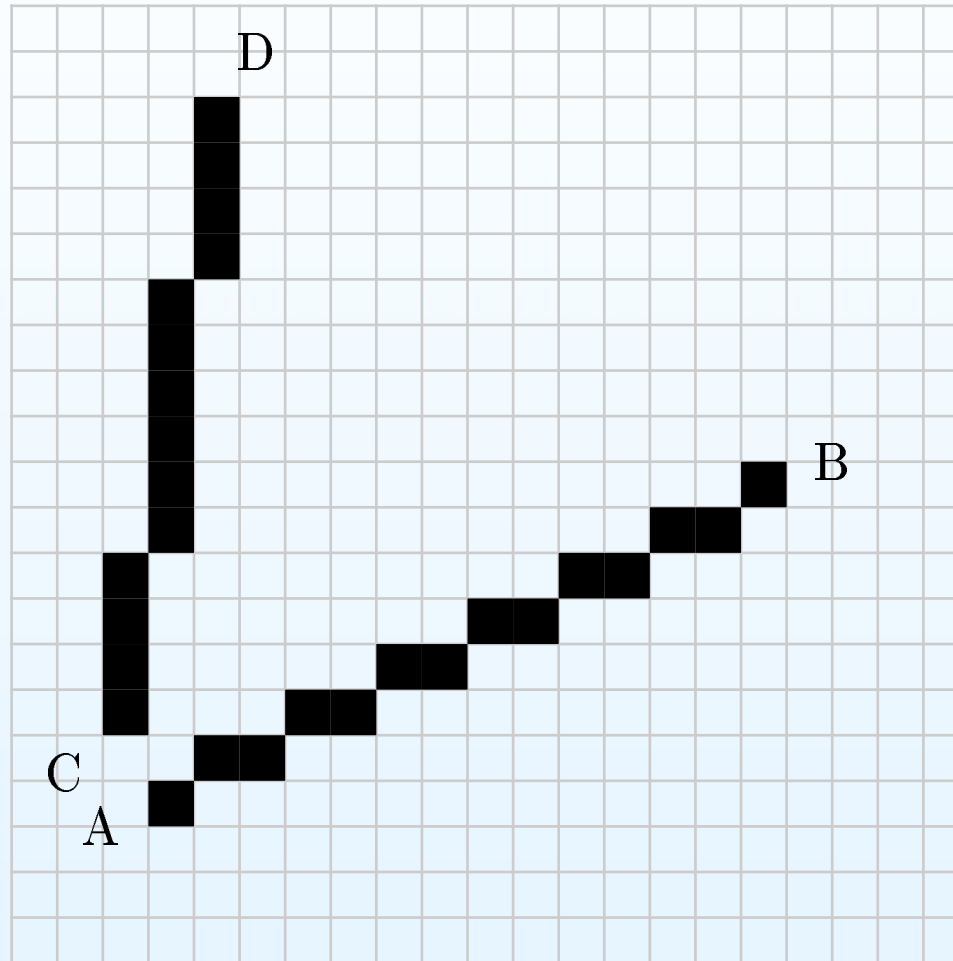
# Rasteryzacja odcinka

## Rzutowanie

## Rasteryzacja

### ● Rasteryzacja odcinka

- Algorytm Bresenhama
- Interpolacja głębokości, etc.
- Interpolacja w wielokątach
- Błędy zaokrąglenia



Rzutowanie

Rasteryzacja

● Rasteryzacja odcinka

● Algorytm

Bresenhama

● Interpolacja

głębokości, etc.

● Interpolacja

w wielokątach

● Błędy zaokrąglania

## Założenia

- $(x, y)$  — współrzędne „abstrakcyjne”, liczby rzeczywiste
- $(i, j)$  — współrzędne ekranowe, liczby całkowite
- $x_2 > x_1, y_2 \geq y_1$
- $y_2 - y_1 \leq x_2 - x_1$
- Zaokrąglenie:  $i_1 = \text{round}(x_1)$ ,  $i_2 = \text{round}(x_2)$ ,  
 $j_1 = \text{round}(y_1)$ ,  $j_2 = \text{round}(y_2)$

## Algorytm

- $y(i) = i_1 + \frac{i-i_1}{i_2-i_1}(j_2 - j_1)$
- $j = \text{round}(y)$
- Kod:

**Wejście:**  $(i_1, j_1)$  — początek odcinka,  $(i_2, j_2)$  — koniec odcinka,  $i_2 > i_1, j_2 \geq j_1, j_2 - j_1 \leq i_2 - i_1$

**Wynik:** Odcinek został wyświetlony

$$m \leftarrow \frac{j_2 - j_1}{i_2 - i_1}$$

writePixel( $i_1, j_1$ )

$$y \leftarrow j_1$$

**for**  $i = i_1 + 1$  **to**  $i_2$  **do**

$$y \leftarrow y + m$$

$$j \leftarrow \text{round}(y)$$

writePixel( $i, j$ )

**end for**

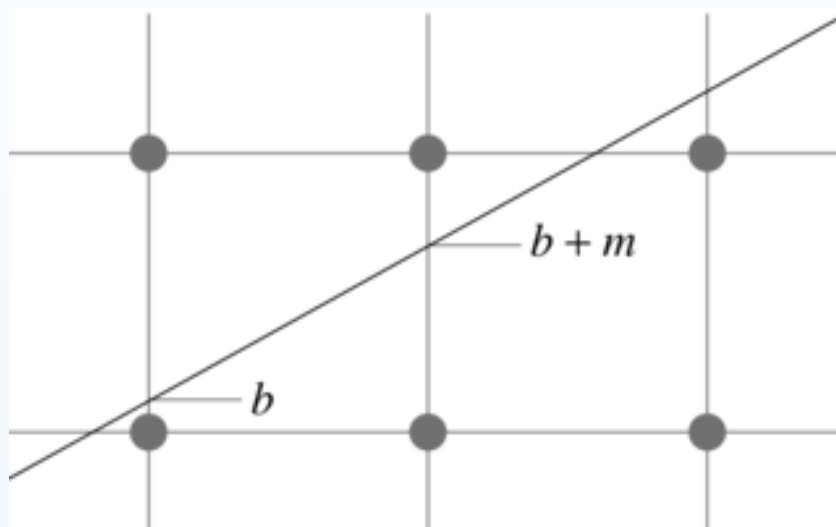
## Kumulacja przyrostu $y$

Rzutowanie

Rasteryzacja

● Rasteryzacja odcinka

- Algorytm Bresenhama
- Interpolacja głębokości, etc.
- Interpolacja w wielokątach
- Błędy zaokrąglania



- na każdym kroku do przyrostu  $y$  dodaje się  $m$
- przechodzimy o jeden piksel w górę, jeżeli przyrost przekroczy  $1/2$

- Algorytm Bresenhama
- Interpolacja głębości, etc.
- Interpolacja w wielokątach
- Błędy zaokrąglenia

## Algorytm 2

**Wejście:**  $(i_1, j_1)$  — początek odcinka,  $(i_2, j_2)$  — koniec odcinka,  
 $i_2 > i_1, j_2 \geq j_1, j_2 - j_1 \leq i_2 - i_1$

**Wynik:** Odcinek został wyświetlony

$$m \leftarrow \frac{j_2 - j_1}{i_2 - i_1}$$

$$b \leftarrow 0$$

writePixel( $i_1, j_1$ )

$$j \leftarrow j_1$$

**for**  $i = i_1 + 1$  **to**  $i_2$  **do**

$$b \leftarrow b + m$$

**if**  $b > \frac{1}{2}$  **then**

$$j \leftarrow j + 1$$

$$b \leftarrow b - 1$$

**end if**

writePixel( $i, j$ )

**end for**



## Eliminacja liczb rzeczywistych

- przyrost jest wielokrotnością  $m = \frac{j_2 - j_1}{i_2 - i_1}$ :
  - $b = k \frac{j_2 - j_1}{i_2 - i_1}$
- $b < \frac{1}{2} \iff 2k(j_2 - j_1) < i_2 - i_1$ 
  - zamieniamy przyrost na przyrost całkowity
  - przyrost całkowity na każdym kroku zwiększa się
    - $2\Delta j = 2(j_2 - j_1)$
  - przechodzimy na wyższy poziom w  $j$ , jeżeli przyrost całkowity przekroczy  $(i_2 - i_1)$

## Algorytm Bresenhama

**Wejście:**  $(i_1, j_1)$  — początek odcinka,  $(i_2, j_2)$  — koniec odcinka,

$$i_2 > i_1, j_2 \geq j_1, j_2 - j_1 \leq i_2 - i_1$$

**Wynik:** Odcinek został wyświetlony

$$m \leftarrow 2(j_2 - j_1)$$

$$b \leftarrow 0$$

**writePixel** $(i_1, j_1)$

$$j \leftarrow j_1$$

$$P \leftarrow i_2 - i_1$$

**for**  $i = i_1 + 1$  **to**  $i_2$  **do**

$$b \leftarrow b + m$$

**if**  $b > P$  **then**

$$j \leftarrow j + 1$$

$$b \leftarrow b - 2P$$

**end if**

**writePixel** $(i, j)$

**end for**

#### Rzutowanie

#### Rasteryzacja

- Rasteryzacja odcinka

- Algorytm

Bresenhama

- Interpolacja  
głębokości, etc.

- Interpolacja  
w wielokątach

- Błędy zaokrąglania

## Interpolacja głębokości, etc.

- $z_1$  jest początkową głębokością,  $z_2$  — końcową
- $z_2 - z_1 > 0$ , nie koniecznie musi być mniejsze, niż  $x_2 - x_1$
- $z$  na każdym kroku będzie się powiększało o  $\frac{\Delta z}{\Delta x}$
- wprowadźmy:
  - przyrost  $q = \text{int} \left( \frac{\Delta z}{\Delta x} \right)$  — część całkowita
  - przyrost błędu  $m = \Delta z - q\Delta x$

- Rasteryzacja odcinka
- Algorytm Bresenhama
- **Interpolacja głębokości, etc.**
- Interpolacja w wielokątach
- Błędy zaokrąglania

## Interpolacja głębokości, etc.

**Wejście:**  $(i_1, z_1)$  — początek odcinka,  $(i_2, z_2)$  — koniec odcinka,

$$i_2 > i_1, z_2 \geq z_1,$$

**Wynik:** Odcinek został wyświetlony

$$q \leftarrow \text{int} \left( \frac{z_2 - z_1}{x_2 - x_1} \right); b \leftarrow 0$$

$$m \leftarrow 2((z_2 - z_1) - q(x_2 - x_1))$$

writePixel( $i_1, z_1$ )

$$z \leftarrow z_1; P \leftarrow i_2 - i_1$$

**for**  $i = i_1 + 1$  **to**  $i_2$  **do**

$$z \leftarrow z + q$$

$$b \leftarrow b + m$$

**if**  $b > P$  **then**

$$z \leftarrow z + 1$$

$$b \leftarrow b - 2P$$

**end if**

writePixel( $i, z$ )

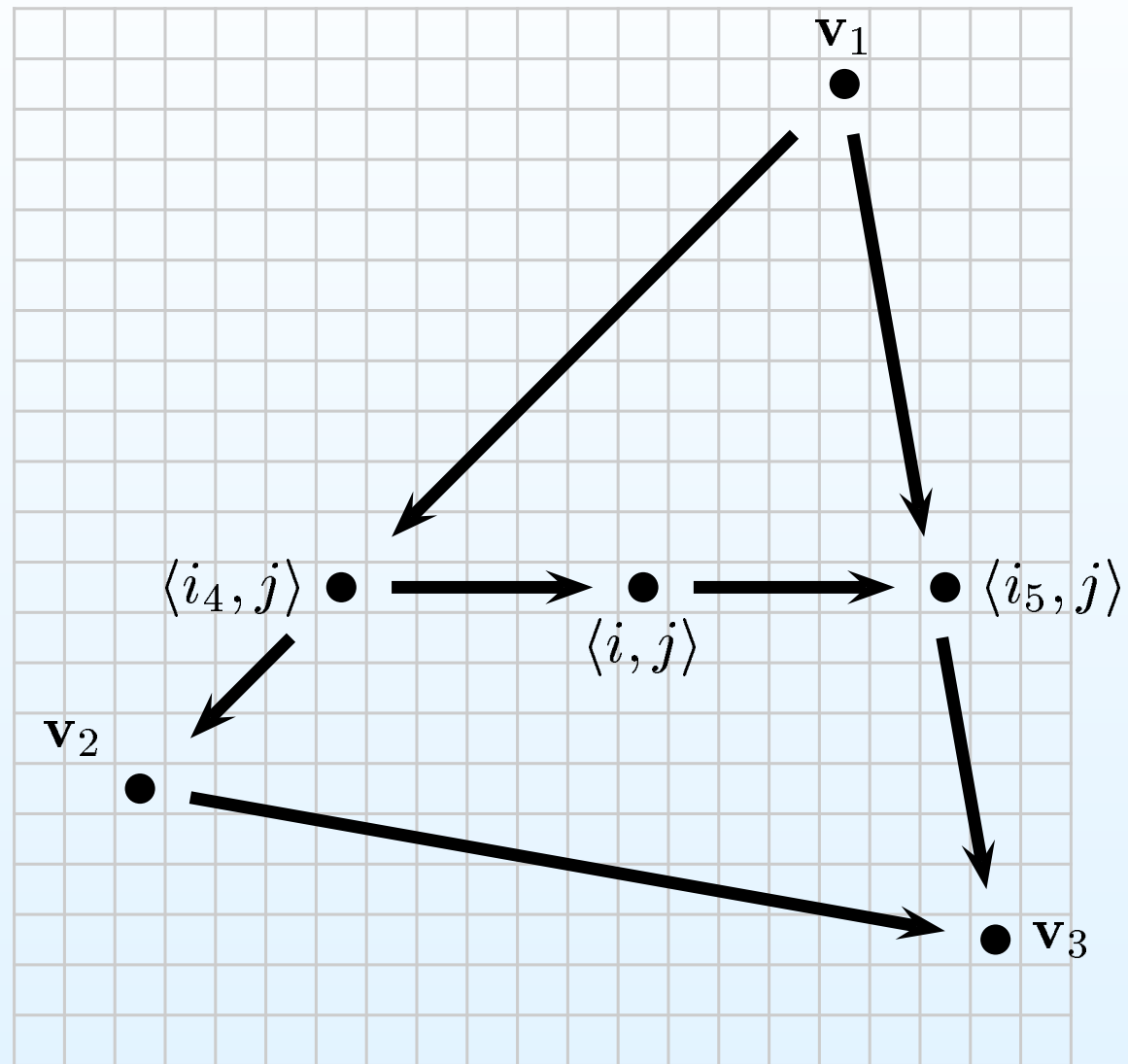
**end for**

# Interpolacja w trójkątach

## Rzutowanie

## Rasteryzacja

- Rasteryzacja odcinka
- Algorytm Bresenhama
- Interpolacja głębokości, etc.
- Interpolacja w wielokątach
- Błędy zaokrąglenia

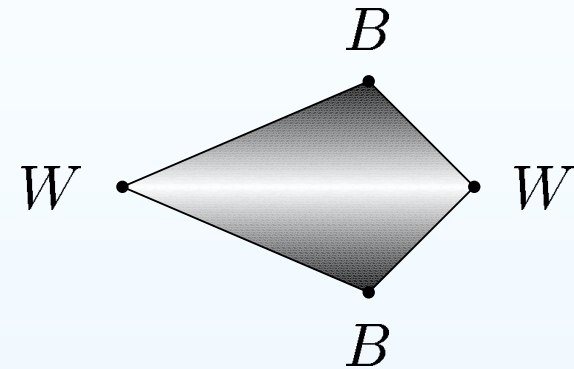
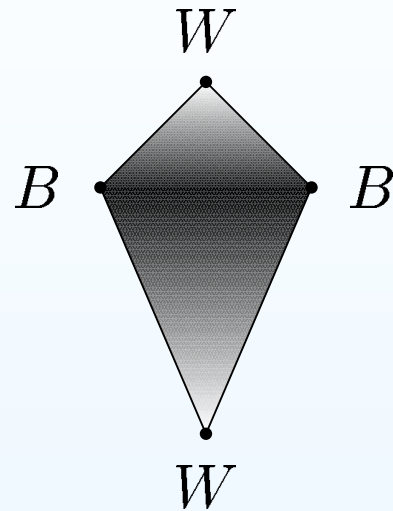


# Interpolacja w wielokątach

## Rzutowanie

## Rasteryzacja

- Rasteryzacja odcinka
- Algorytm Bresenhama
- Interpolacja głębokości, etc.
- **Interpolacja w wielokątach**
- Błędy zaokrąglenia

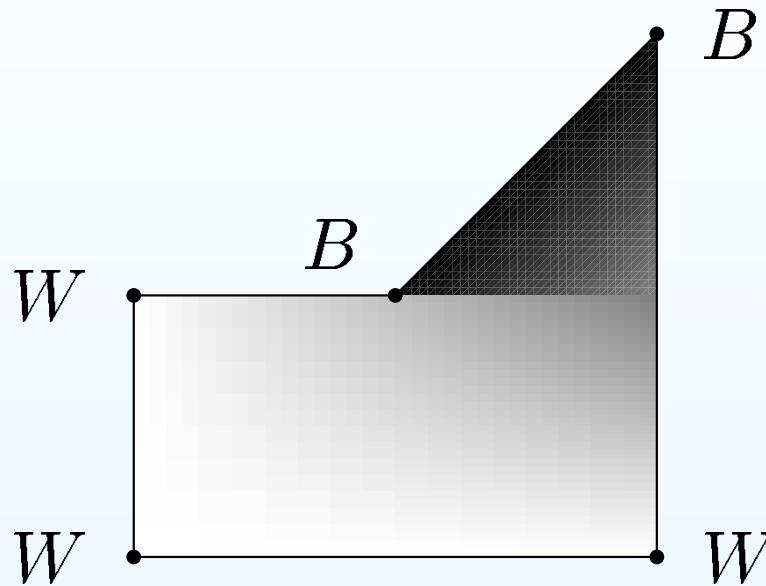


# Wielokąt niewypukły

## Rzutowanie

## Rasteryzacja

- Rasteryzacja odcinka
- Algorytm Bresenhama
- Interpolacja głębokości, etc.
- **Interpolacja w wielokątach**
- Błędy zaokrąglenia

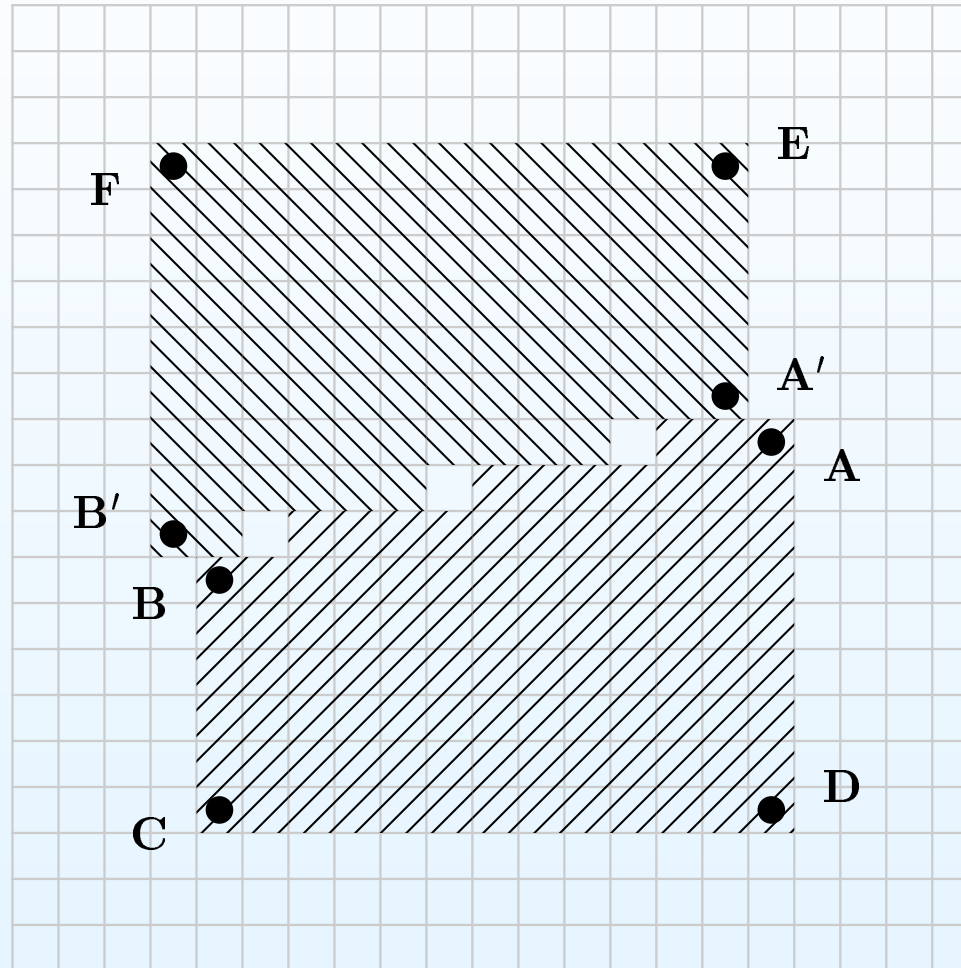


# Niebezpieczeństwa związane z zaokrągleniem

## Rzutowanie

## Rasteryzacja

- Rasteryzacja odcinka
- Algorytm Bresenhama
- Interpolacja głębokości, etc.
- Interpolacja w wielokątach
- Błędy zaokrąglenia





# Niedobre przymykanie

## Rzutowanie

## Rasteryzacja

- Rasteryzacja odcinka
- Algorytm Bresenhama
- Interpolacja głębokości, etc.
- Interpolacja w wielokątach
- Błędy zaokrąglenia

