6 Grafika 2D.

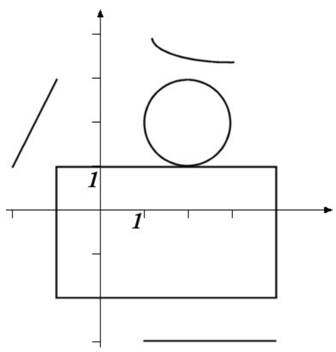
Jacek Tarasiuk

6.1 Obiekty 2D

W wektorowej grafice dwuwymiarowej obraz opisuje się jako zbiór prostych obiektów geometrycznych takich jak: odcinki, elipsy, prostokąty itp[1]. Każdy z takich obiektów

Rysunek 6.1

Przykład grafiki wektorowej



można łatwo zdefiniować podając jedynie kilka parametrów. W takiej reprezentacji obrazek z rysunku 6.1 mógłby zostać opisany na przykład jako:

```
O -2 1 -1 3          // odcinek i współrzędne końców
K 2 2 1              // koło i współrzędne środka oraz promień
P -1 1 4 -2          // prostokąt o zadanych narożnikach
E 3 4 0.5 2 180 270  // fragment elipsy o zadanym środku, długości półosi oraz
```

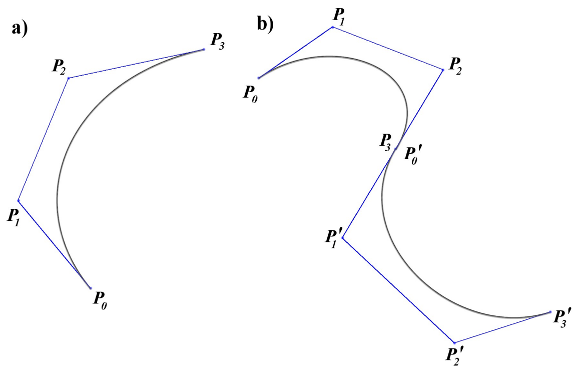
1Czasami wykorzystuje się również bardziej złożone krzywe opisywane równaniami analitycznymi lub formułami rekurencyjnymi.

```
początkowym i końcowym kącie rysowania
O 1 -3 4 -3            // odcinek i współrzędne jego końców
```

6.1.1 Krzywa Beziera

krzywa Beziera Często używanym w grafice dwuwymiarowej obiektem są krzywe Beziera. Krzywe te pozwalają w sposób gładki modelować różnego rodzaju krzywizny. Krzywa taka **łamana Beziera** definiowana jest przez zbiór punktów kontrolnych, które tworzą łamaną Beziera. Krzywa ta jest gładka i charakteryzuje się tym, że jest styczna do pierwszego i ostatniego odcinka łamanej, która ją definiuje (rys.6.2a). Własność ta jest szczególnie cenna, gdyż umożliwia sklejanie krzywych w dłuższe krzywizny. Warunkiem gładkiego sklejenia jest współliniowość ostatnich dwóch punktów kontrolnych jednej krzywej z pierwszymi dwoma punktami kontrolnymi drugiej krzywej (rys.6.2b).

Rysunek 6.2
a) krzywa i łamana Beziera
b) gładkie łączenie krzywych



Równanie parametryczne krzywej Beziera można przedstawić w postaci szeregu wielomianów Bernsteina, gdzie współczynnikami w szeregu będą punkty kontrolne krzywej Beziera:

$$Q(t)=\sum_{i=0}^{n} P_i B_i^n(u) \quad \text{, gdzie} \quad Q(t)=(x(t), y(t))$$

a zmienną u wiąże się ze zmienną t poprzez lokalna parametryzację: $t=t_0+u(t_1-t_0)$.

Wielomiany Bernsteina definiuje się jako:

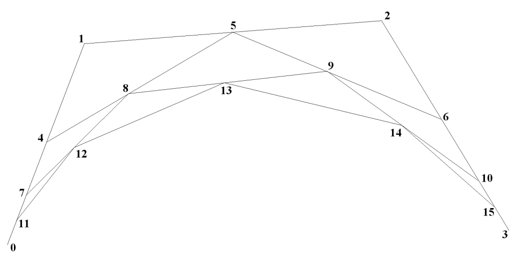
$$B_i^n(u)=\binom{n}{i}(1-u)^{n-1}\cdot u^i \quad \text{, gdzie} \quad u\in[0,1] \quad .$$

Dla najczęściej spotykanego przypadku $n=3$ krzywą Beziera można zapisać jako:

$$Q(t)=P_0(1-t)^3+3P_1(1-t)^2t+3P_2(1-t)t^2+P_3t^3 \quad .$$

Ciekawostka.

Wybierzmy cztery dowolne punkty 0 , 1 , 2 , 3 jak na poniższym rysunku:



Jeśli odcinek 0-1 podzielimy na pół dostaniemy punkt oznaczony numerem 4. Podobnie dzieląc odcinek 1-2 otrzymamy punkt 5, a dla odcinka 2-3 punkt 6. Jeżeli teraz połączymy punkty 0-4-5-6-3 otrzymamy nową łamaną. Teraz każdy z odcinków nowej łamanej możemy podzielić na pół, a otrzymane punkty połączyć. Otrzymamy łamaną 0-7-8-9-10-3. Powtarzając powyższe czynności w nieskończoność, w granicy, otrzymamy krzywą Beziera.

6.2 Reprezentacja obrazu 2D.

Reprezentacja obrazu wektorowego poprzez obiekty geometryczne jest wygodna, tylko do momentu, gdy nie zajdzie potrzeba jego transformacji i przekształcenia. Wówczas okazuje się, że znacznie wygodniej jest pracować na odcinkach. Dlatego w praktyce, często wszystkie obiekty geometryczne 2D przybliża się odcinkami. Prostokąt zamienia się na cztery połączone odcinki, okrąg na zadaną z góry liczbę odcinków (która może zależeć od jego promienia), podobnie postępujemy z elipsami i łukami. Również krzywe Beziera zapisuje się w postaci zbioru połączonych odcinków. W takiej reprezentacji opis obrazu składa się z:

- listy wierzchołków, z których każdy jest punktem $\mathbf{P_i(x_i , y_i)}$,

- lista krawędzi, z których każda zdefiniowana jest przez uporządkowaną parę wierzchołków,
- każdy obiekt jest reprezentowany przez listę tworzących go krawędzi,
- cały obraz zapisany jest w postaci listy obiektów.

Wspomniane uporządkowanie wierzchołków tworzących odcinki wykorzystywane jest na przykład do badania położenia punktu względem wielokąta. Jeżeli wszystkie wierzchołki wielokąta uporządkowane są dodatnio (to jest przeciwnie do ruchu wskazówek zegara) to jeżeli punkt leży po lewej stronie każdego z odcinków tworzących wielokąt to oznacza, że znajduje się on wewnątrz wielokąta.

6.3 Transformacje obiektów 2D.

Przekształcenie obiektu 2D sprowadza się do przekształceniu położeń wszystkich jego wierzchołków, a następnie połączenia ich. Przekształcenie wierzchołka o współrzędnych **P(x,y)** polega na znalezieniu jego nowych współrzędnych **P'(x',y')**. W tym celu każdy punkt P(x,y) zamienia się na reprezentację wektorową:

$$P(x,y) \quad \rightarrow \quad \vec{v}=\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad .$$

współrzędne jednorodne

Współrzędne punktu na płaszczyźnie wyrażone w postaci wektora o trzech współrzędnych, z których ostatnia jest równa jedynce[2] nazywamy współrzędnymi jednorodnymi. Dowolna transformacja pojedynczego punktu sprowadza się do przemnożenia reprezentującego go wektora przez odpowiednią macierz przekształcenia:

$$\vec{v}'=M\cdot\vec{v} \quad .$$

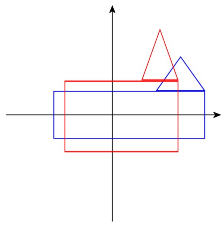
Transformacja obiektu złożonego z wielu odcinków polega na przekształceniu położeń wszystkich punktów definiujących odcinki, a następnie narysowaniu odcinków w nowych położeniach.

Poniżej podano macierze przekształceń dla najważniejszych transformacji.

2 Dopuszczalne są także wektory postaci [x,y,0]. Wektor taki reprezentuje punkt leżący w nieskończoności, a kierunek w którym znajduje się ten punkt można wyznaczyć na podstawie zależności: $tg(\alpha)=y/x$

skalowanie wzdłuż osi X o S_x oraz wzdłuż osi Y o S_y

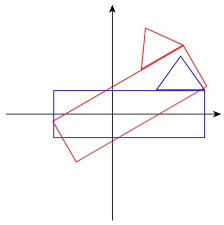
$$M = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



Jeżeli spełniony jest warunek $S_x = S_y$ proporcje obrazu zostaną zachowane, a skalowanie takie nazywamy jednorodnym, w przeciwnym przypadku proporcje ulegną zmianie, a skalowanie nazywamy niejednorodnym.

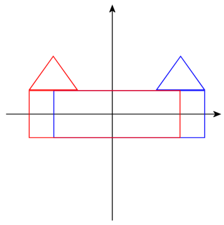
obrót o kąt α

$$M = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



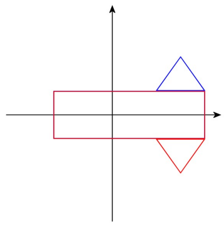
odbicie lustrzane względem osi Y

$$M = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



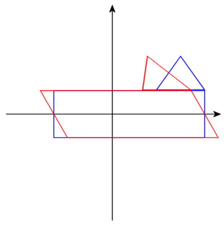
odbicie lustrzane względem osi X

$$M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



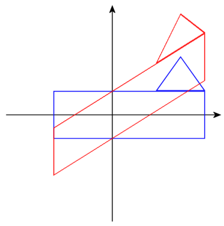
pochylenie w kierunku osi X

$$M = \begin{bmatrix} 1 & a & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



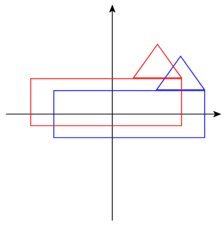
pochylenie w kierunku osi Y

$$M = \begin{bmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



przesunięcie o Δx w kierunku osi X, oraz o Δy w kierunku osi Y

$$M = \begin{bmatrix} 1 & 0 & \Delta x \\ 0 & 1 & \Delta y \\ 0 & 0 & 1 \end{bmatrix}$$



6.3.1 Składanie przekształceń.

Oczywiście transformacje obiektów 2D można ze sobą składać. Dowolnemu złożeniu kilku przekształceń odpowiada jedna macierz będąca iloczynem macierzy poszczególnych przekształceń składowych. Na przykład załóżmy, że chcemy dany obiekt najpierw zmniejszyć wzdłuż osi Y o połowę, następnie obrócić go o 30°, po czym całość jeszcze odbić symetrycznie względem osi Y, wówczas macierz przekształcenia będzie miała w przybliżeniu następująca postać:

$$\begin{bmatrix} -0{,}866 & 0{,}025 & 0{,}000 \\ 0{,}500 & 0{,}433 & 0{,}000 \\ 0{,}000 & 0{,}000 & 1{,}000 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0{,}866 & -0{,}5 & 0 \\ 0{,}5 & 0{,}866 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0{,}5 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Ze szczególnym przypadkiem złożenia przekształceń mamy do czynienia wtedy, gdy chcemy wykonać jakąś transformację (obrót, skalowanie itp.) wokół punktu $\mathbf{P(x_0,y_0)}$ nie będącym środkiem układu współrzędnych. W takiej sytuacji musimy najpierw przesunąć układ tak, aby jego środek pokrył się z punktem $\mathbf{P(x_0,y_0)}$, następnie wykonać stosowne przekształcenie, a na koniec przesunąć układ ponownie o taki sam wektor jak na początku tylko przeciwnie skierowany. Macierz przekształceń będzie wówczas miała postać:

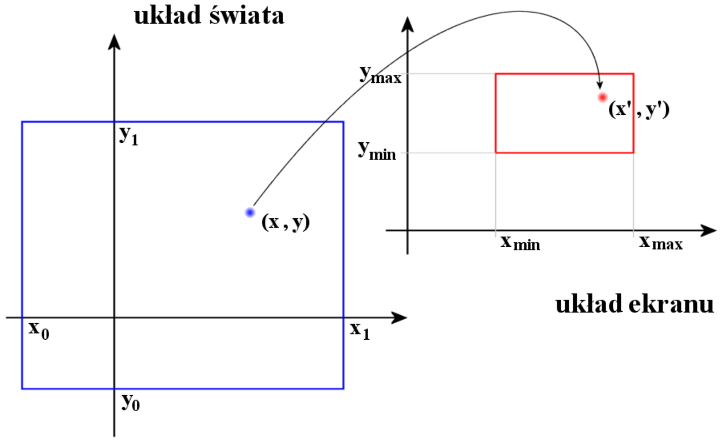
$$M=\begin{bmatrix}1 & 0 & x_0\\0 & 1 & y_0\\0 & 0 & 1\end{bmatrix}\cdot T\cdot\begin{bmatrix}1 & 0 & -x_0\\0 & 1 & -y_0\\0 & 0 & 1\end{bmatrix}$$

gdzie T jest macierzą przekształcenia wykonywanego wokół punktu $\mathbf{P(x_0,y_0)}$.

6.4 Skalowanie i okienkowanie.

Układ, w którym wyrażone są współrzędne wierzchołków obrazu nazwijmy układem świata, a układ współrzędnych, w którym mają zostać wyrażone wierzchołki obrazu w celu wyświetlenia na ekranie nazwijmy układem ekranu. Skalowanie i okienkowanie polega na takim przeliczeniu współrzędnych wierzchołków z układu świata do układu ekranu, aby prostokąt wyznaczony przez współrzędne x_1 , y_1 , x_2 , y_2 w układzie świata przekształcony został w prostokąt x_{min} , y_{max} , x_{max} , y_{min} w układzie ekranu (rysunek 6.3).

Rysunek 6.3
Układ świata i układ ekranu



Zauważmy, że w ogólności proporcje boków obu obszarów nie musza być jednakowe. Zagadnienie sprowadza się zatem do znalezienia formuły przekształcającej dowolny punkt (x,y) we współrzędnych świata w odpowiadający mu punkt (x',y') we współrzędnych ekranowych. Można to zrobić korzystając z prostych proporcji:

$$\frac{x-x_0}{x_1-x_0}=\frac{x'-x_{min}}{x_{max}-x_{min}} \qquad \frac{y-y_0}{y_1-y_0}=\frac{y'-y_{min}}{y_{max}-y_{min}} \; .$$

Po rozwiązaniu obu równań otrzymamy ostateczną formułę:

$$x' = S_x(x - x_0) + x_{min} \qquad S_x = \frac{x_{max} - x_{min}}{x_1 - x_0}$$
$$y' = S_y(y - y_0) + y_{min} \qquad \text{gdzie:} \qquad S_y = \frac{y_{max} - y_{min}}{y_1 - y_0} \quad .$$

Zauważmy, że powyższe równania można zapisać w postaci macierzowej jako złożenie trzech przekształceń:

$$\begin{bmatrix} 1 & 0 & x_{min} \\ 0 & 1 & y_{min} \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & -x_0 \\ 0 & 1 & -y_0 \\ 0 & 0 & 1 \end{bmatrix}$$

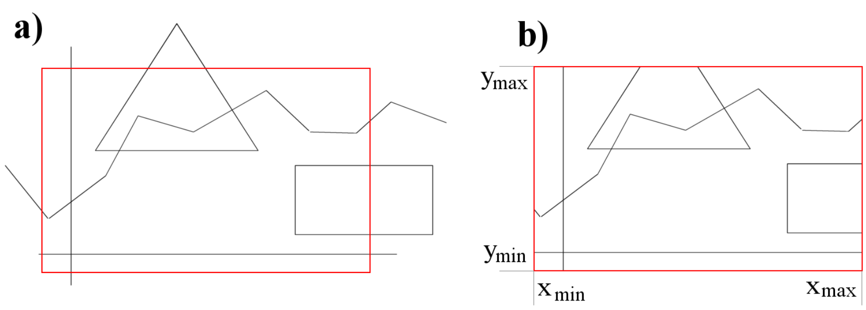
co po wymnożeniu daje ostatecznie macierz przejścia:

$$M = \begin{bmatrix} S_x & 0 & x_{min} - S_x x_0 \\ 0 & S_y & y_{min} - S_y y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad .$$

6.5 Obcinanie (cliping).

Odrębnym zagadnieniem jest obcinanie odcinków do prostokątnego obszaru ograniczonego wartościami współrzędnych $x_{min}, x_{max}, y_{min}, y_{max}$ (tzw. prostokąt obcięcia).

Rysunek 6.4
a) rysunek wektorowy
b) rysunek po obcięciu



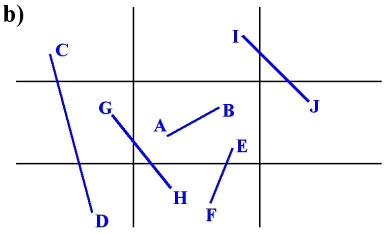
Istnieje co najmniej kilka interesujących algorytmów obcinających odcinki do zadanego prostokąta. Metoda przedstawiona poniżej (tzw. algorytm Cohena-Sutherlanda) jest jedną z prostszych i bardziej efektywnych, a do tego łatwo ją rozszerzyć na przypadek trójwymiarowy. Poszczególnym regionom wokół prostokąta obcięcia oraz samemu prostokątowi przypisuje się kody pokazane na rysunku 6.5a.

Rysunek 6.5
a) kodowanie regionów
b) położenie odcinka względem prostokąta obcięcia

a)

0101	**0100**	**0110**
0001	**0000**	**0010**
1001	**1000**	**1010**

b)



Zauważmy, że poszczególne bity kodów znajdziemy błyskawicznie sprawdzając bit znaku odpowiednich różnic:

1 bit kodu jest bitem znaku różnicy:	$y_{min}-y$
2 bit kodu jest bitem znaku różnicy:	$y-y_{max}$
3 bit kodu jest bitem znaku różnicy:	$x-x_{max}$
4 bit kodu jest bitem znaku różnicy:	$x_{min}-x$

Na rysunku 6.5b pokazano wszystkie możliwe relacje położenia odcinka względem prostokąta obcięcia. Przyjmijmy, że współrzędne początku odcinka określa punkt $P1(x_1,y_1)$, a końca odcinka punkt $P2(x_2,y_2)$. Punktom tym przyporządkujemy kody C1 i C2 w zależności od tego, w którym regionie leżą punkty P1 i P2.

Zauważmy teraz:

- jeśli $(C1=0)\wedge(C2=0)$ to odcinek leży w całości w prostokącie obcięcia i nie będzie dalej przetwarzany (odcinek AB),
- jeśli $(C1\wedge C2)\neq 0$ to odcinek w całości leży poza prostokątem obcięcia i też nie będzie dalej przetwarzany (odcinek CD).

W pozostałych przypadkach postępujemy zgodnie z poniższym algorytmem:

- jeżeli C1=0 to zamieniamy P1 z P2 i C1 z C2,

- jeżeli w C1 ustawiony jest bit 0001 to:

$$x_1{}'=x_{min} \qquad y_1{}'=y_1+\frac{y_2-y_1}{x_2-x_1}(x_{min}-x_1)$$

- jeżeli w C1 ustawiony jest bit 0010 to:

$$x_1{}'=x_{max} \qquad y_1{}'=y_1+\frac{y_2-y_1}{x_2-x_1}(x_{max}-x_{min})$$

- jeżeli w C1 ustawiony jest bit 0100 to:

$$y_1{}'=y_{min} \qquad x_1{}'=x_1+\frac{x_2-x_1}{y_2-y_1}(y_{min}-y1)$$

- jeżeli w C1 ustawiony jest bit 1000 to:

$$y_1{}'=y_{max} \qquad x_1{}'=x_1+\frac{x_2-x_1}{y_2-y_1}(y_{max}-y_1)$$

W powyższych równaniach $x_1{}'$ i $y_1{}'$ są współrzędnymi początku odcinka po jego obcięciu do prostokąta obcięcia. Powyższy algorytm powtarzamy dwukrotnie. Pierwszy przebieg obetnie odcinki GH i IJ z jednego końca. Drugi przebieg w przypadku odcinka GH obetnie drugi koniec, a w przypadku odcinka IJ stwierdzi, że odcinek leży poza prostokątem obcięcia.

6.6 Przykładowe programy.

Do skryptu dołączono następujące programy demonstracyjne:

Rysowanie odcinka 1 – przykład algorytmu EFLA rysowania odcinka (ciekawostka nie omówiona w skrypcie).

Rysowanie odcinka 2 – wykorzystanie funkcji scanline do szybkiego rysowania (nie omówione w skrypcie).

Krzywe Beziera – porównanie krzywych Beziera z standardowej biblioteki oraz liczonej „ręcznie”.

Transformacje 2D – dwuwymiarowe transformacje prostych figur geometrycznych.

Okienkowanie i obcinanie – prosty rysunek skalowany, przenoszony ze współrzędnych rzeczywistych do współrzędnych urządzenia oraz obcinanie do wybranego prostokąta.