Edge Function

Pintér Bálint

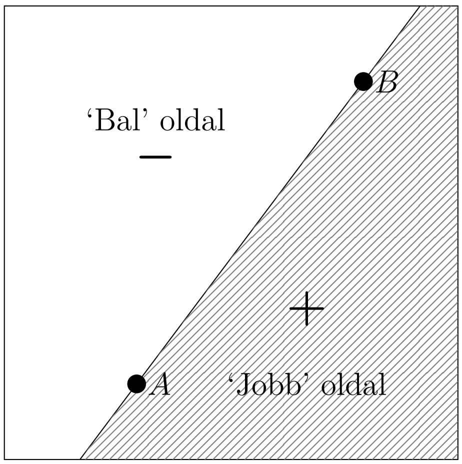
November 25, 2025

Tartalomjegyzék

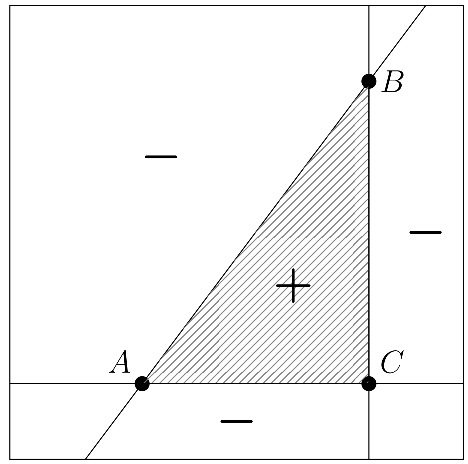
1 Bevezetés

Az edge function egy lineáris függvény, amellyel egy egyenessel kettéosztott síkon lévő pontokat lehet három régióba osztani:

- Az egyenestől 'jobbra' lévő pontok (ahol a függvény értéke pozitív).
- Az egyenestől 'balra' lévő pontok (ahol a függvény értéke negatív).
- Az egyenesre illeszkedő pontok (ahol a függvény értéke nulla).



1. Ábra
Sík felosztása A-n és B-n átmenő egyenessel



2. Ábra
Az AB, BC és CA szakaszok által meghatározott egyenesek 'jobb' oldalainak metszete egy háromszög belseje

A 2. ábra szemlélteti, hogy a háromszög 'belseje' a három oldalhoz tartozó megfelelő irányú edge function előjelei alapján meghatározható. Ezt használjuk 3D renderelésben, hogy a pixelt (pontot) tartalmazza-e a háromszög.

2 Edge function levezetése

- Legyen a szakasz kezdőpontja $P_0 = (X, Y)$
- Legyen a szakasz végpontja $P_1 = (X + dX, Y + dY)$
  - Ekkor a $\vec{P_0P_1} = \vec{v} = ((X + dX) - X, (Y + dY) - Y) = (dX, dY)$
- Legyen a vizsgált pont $P = (x, y)$
  - Ekkor a $\vec{P_0P} = \vec{u} = (x - X, y - Y)$

Az edge function lényegében a két vektor által alkotott 2×2-es mátrix determinánsa.

$$\det \begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix} = u_x v_y - u_y v_x \quad / \ \det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ad - bc$$

Helyettesítsük be az értékeinket:

$$\begin{aligned} u_x &= x - X \\ u_y &= y - Y \\ v_x &= dX \\ v_y &= dY \end{aligned}$$

A végeredmény:

$$\begin{aligned} E(x, y) &= (x - X)dY - (y - Y)dX \\ E_{P_0P_1}(P) &= (x - X)dY - (y - Y)dX \end{aligned}$$

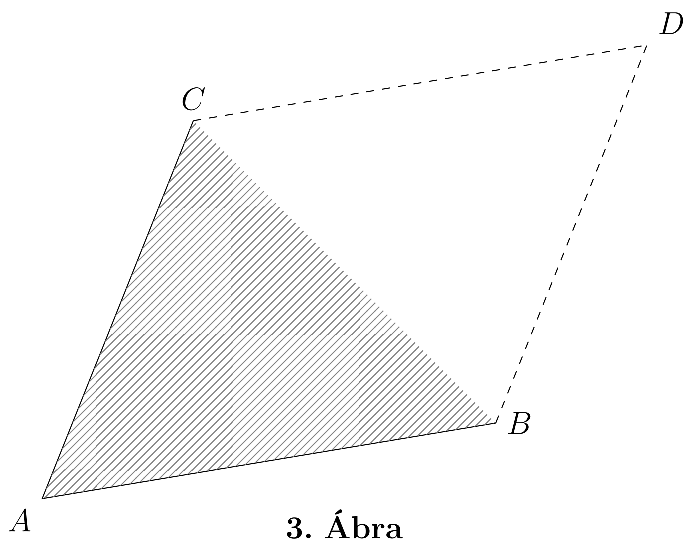
3 Edge function működési elve

3.1 Edge function geometriai jelentése

A determináns előjele 2×2-es mátrix esetén:

- Pozitív esetben a vektorok jobbsodrású (pozitív irányítású) rendszert alkotnak.
- Negatív esetben a vektorok balsodrású (negatív irányítású) rendszert alkotnak.
- Nulla esetén párhuzamosak.

A determináns abszolút értéke 2×2-es mátrix esetén a mátrix sorvektorai által kifeszített paralelogramma területét jelenti. Így, ha ezt elosztjuk kettővel, akkor megkapjuk a háromszög területét.

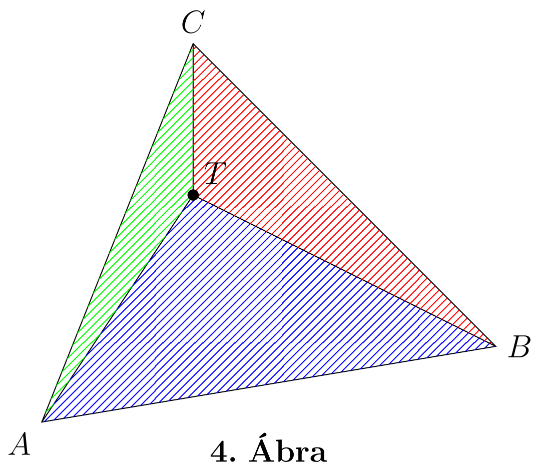


3. Ábra
Paralelogramma területe

$$E_{AC}(B) = T_{ABCD_{\varparallelogram}}$$
$$\frac{1}{2}E_{AC}(B) = T_{ABC_{\triangle}}$$

3.1.1 Baricentrikus koordináták

Az előbb kiszámolt területet felhasználjuk a háromszögön belüli pont (pixel) baricentrikus koordinátáinak kiszámolására (A baricentrikus koordinátákkal tudjuk interpolálni a háromszög egy belső pontjára a háromszög csúcsaihoz vett értékeket).



4. Ábra
A háromszög egy belső
ponttal felosztva.

$$\lambda_A = \frac{T_{TBC_\triangle}}{T_{ABC_\triangle}} \qquad / \; T_{TBC_\triangle} = \frac{1}{2} E_{CB}(T)$$

$$\lambda_B = \frac{T_{ATC_\triangle}}{T_{ABC_\triangle}} \qquad / \; T_{ATC_\triangle} = \frac{1}{2} E_{AC}(T)$$

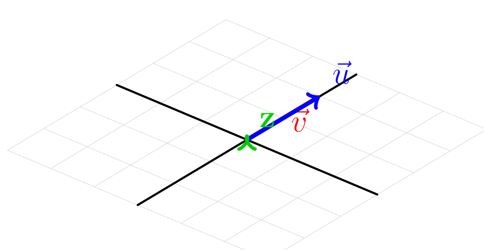
$$\lambda_C = \frac{T_{ATB_\triangle}}{T_{ABC_\triangle}} \qquad / \; T_{ATB_\triangle} = \frac{1}{2} E_{BA}(T)$$

Fontos tulajdonságuk:

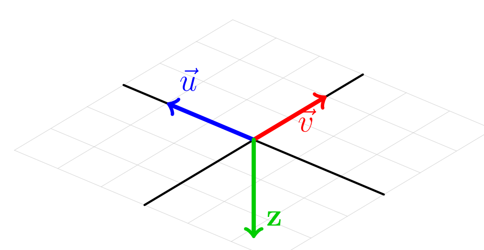
$$\lambda_A + \lambda_B + \lambda_C = 1 \qquad / \text{ Területszámítás axiómáiból is következik}$$

3.2 Vektoriális szorzatos magyarázat

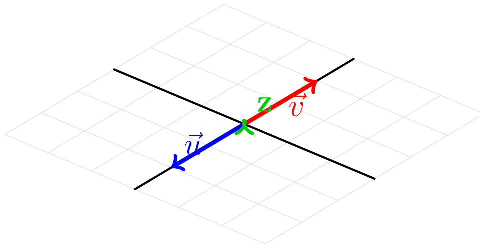
A vektoriális szorzat definíció szerint csak háromdimenziós vektorokra van értelmezve. Ha azonban a $\vec{u}$ és $\vec{v}$ vektorainkat kiegészítjük $z = 0$ koordinátával, akkor az így kapott két vektor vektoriális szorzatának a z koordinátája az edge function értéke.



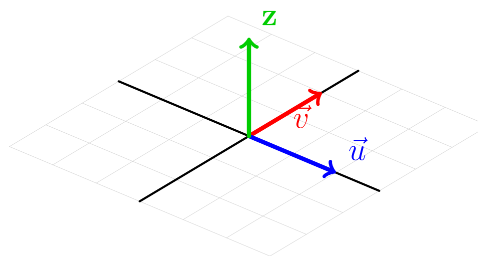
$z = 0 \Rightarrow$
Párhuzamos a két vektor
A pont a vonalon van



$z < 0 \Rightarrow$
A pont ‘balra’ van a vonaltól



$z = 0 \Rightarrow$
Párhuzamos a két vektor
A pont a vonalon van



$z > 0 \Rightarrow$
A pont ‘jobbra’ van a vonaltól

5. Ábra
Vektoriális szorzattal szemléltetve

4 Inkrementalitás

Mivel az edge function lineáris számolhatjuk inkrementálisan. Elég egyszer kiszámolni mindegyik oldalra magát az edge functiont:

$$E(x, y) = (x - X)dY - (y - Y)dX$$

Utána egy összeadással kiszámolhatjuk a többi pontokra (pixelekre)
A következő egyenletek képernyő térbeli koordináta-rendszerben vannak x jobbra nő, y lefele nő:

$$\begin{aligned} E(x+1, y) &= E(x, y) + dY \\ E(x-1, y) &= E(x, y) - dY \\ E(x, y+1) &= E(x, y) - dX \\ E(x, y-1) &= E(x, y) + dX \end{aligned}$$

Az edge function 5 műveletjét (3 kivonás, 2 szorzás), így lecsökkentjük egy műveletre (1 összeadás). Ez egy 1920×1080 felbontású képnél például jelentős mennyiségű műveletet jelent. Használhatjuk egy adott L távol lévő pixelre is.

$$\begin{aligned} E(x+L, y) &= E(x, y) + L \times dY \\ E(x-L, y) &= E(x, y) - L \times dY \\ E(x, y+L) &= E(x, y) - L \times dX \\ E(x, y-L) &= E(x, y) + L \times dX \end{aligned}$$

Bizonyítása egyik esetre:

$$\begin{aligned} &E(x+L, y) = ((x+L) - X)dY - (y - Y)dX \\ &E(x+L, y) = (x - X + L)dY - (y - Y)dX \\ &E(x+L, y) = (x - X)dY + L \times dY - (y - Y)dX \\ &E(x+L, y) = (x - X)dY - (y - Y)dX + L \times dY \quad / \; E(x, y) = (x - X)dY - (y - Y)dX \\ &E(x+L, y) = E(x, y) + L \times dY \end{aligned}$$

A többi esetet is hasonlóan lehet belátni.

Források

Juan Pineda. A parallel algorithm for polygon rasterization. *SIGGRAPH Comput. Graph.*, 22(4):17–20, June 1988a. ISSN 0097-8930. doi: 10.1145/378456.378457. URL https://doi.org/10.1145/378456.378457.

Juan Pineda. A parallel algorithm for polygon rasterization. In *Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '88, page 17–20, New York, NY, USA, 1988b. Association for Computing Machinery. ISBN 0897912756. doi: 10.1145/54852.378457. URL https://doi.org/10.1145/54852.378457.