Sutherland-Hodgman algoritmus

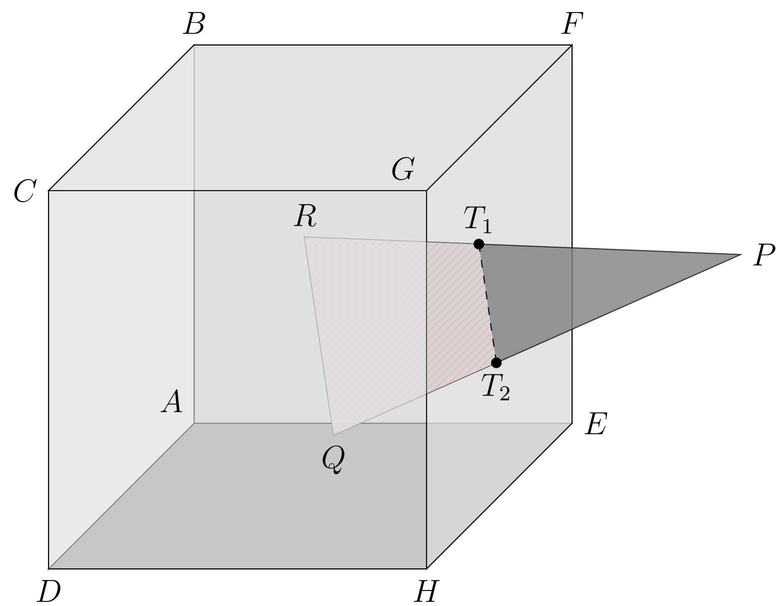
Pintér Bálint

2025. november 29.

Tartalomjegyzék

1. Bevezetés

A Sutherland-Hodgman algoritmus egy clipping algoritmus. Feladata, hogy levágja az alakzatok látómezőn kívül eső részeit. A teljesen kívül eső alakzatokat elveti, a vágósíkokat metsző alakzatokat pedig úgy vágja meg, hogy csak a látható rész maradjon. Fontos megjegyezni, hogy csak konvex vágótérrel működik helyesen. A vágott sokszög lehet konkáv, de ilyenkor plusz feldolgozást igényelhet.



1. Ábra:
Sutherland-Hodgman algoritmus szemléltetése
$ABCDEFGH$ vágótér
PQR Levágandó sokszög
RT_1T_2Q Megvágott sokszög

2. Az algoritmus

2.1. Vágótér (clip space)

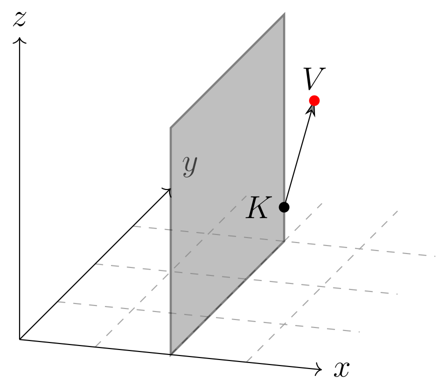
A levágást perspektív mátrixszal való szorzás után, de a w-vel való osztás (perspective divide) előtt végezzük homogén koordinátákkal. Ez az úgynevezett clip space. A perspektív mátrixszal való szorzás előtt a látótér egy csonkagúla (viewing frustum). A perspektív mátrixszal való szorzás után a vágási feltételek homogén koordinátákkal egy kocka határait adják meg. Így a vágás feltételei egyszerűen vizsgálhatóak. Clip space-ben végezzük a levágást, mert kameratérben a látótér egy csonkagúla, amire nehéz lenne vágni. A perspective divide után a kamera mögötti pontok (melyekre $w < 0$) helytelenül vetítődnek a képernyőre, ezért a vágást a perspective divide előtt kell elvégezni.

2.2. A vágás lépései

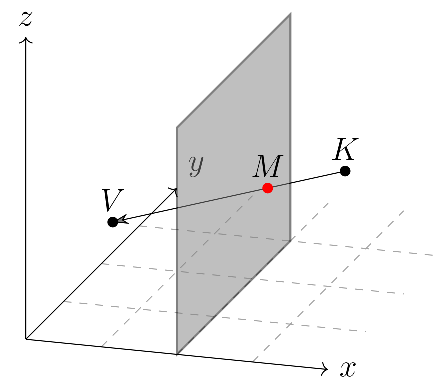
Egy sokszöget a pontjai megadott sorrendje határozza meg. Az n csúcsú sokszög oldalait a következőképpen képezzük a csúcsaiból:

$$P_nP_1, P_1P_2, P_2P_3, ..., P_{n-1}P_n$$

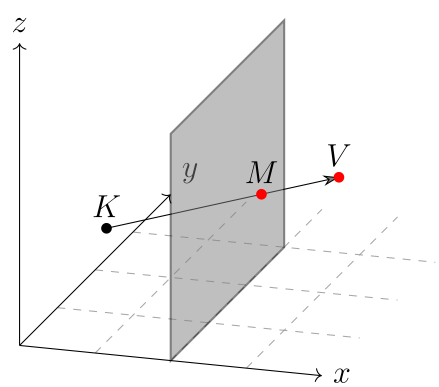
Az algoritmusunk vesz egy pontokból álló bemenetet (először a sokszög pontjai a bemenet), amelyet vág egy síkhoz, majd ennek a folyamatnak a kimenete lesz a következő síkhoz való vágásnak a bemenete. A sokszög oldalait teszteljük a síkhoz képest. Az oldalnak négy lehetséges elhelyezkedése van a síkhoz képest:



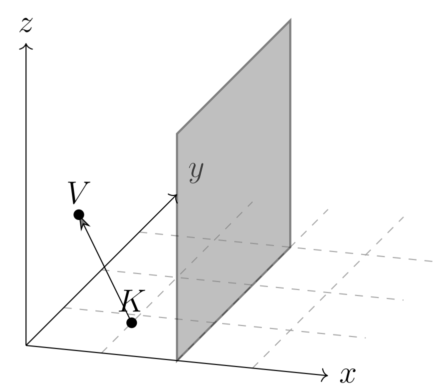
1. (BE→BE) Mindkét pontja belül van.
Csak a végpontot adjuk a kimenethez. (A kezdeti pontja már hozzá lett/lesz adva, amikor végpontként vizsgáljuk/vizsgáltuk.)



2. (BE→KI) Kezdeti pontja belül van, de a végpontja kívül van.
Az oldal és a sík metszéspontját adjuk a kimenethez.



3. (KI→BE) Kezdeti pontja kívül van, de a végpontja belül van.
A körüljárási irány megtartása érdekében először az oldal és a sík metszéspontját, majd a végpontot adjuk hozzá a kimenethez.



4. (KI→KI) Mindkét pontja kívül van.
Nem adunk semmit a kimenethez.

2. Ábra
K - Kezdőpont
V - Végpont
M -Metszéspont

A pirossal jelölt pontok lesznek hozzáadva a kimenethez

A pontokról clip space-ben egyszerű elsőfokú egyenlőtlenséggel el lehet dönteni, hogy a sík melyik oldalán van (mivel egy kocka). A perspektív mátrixunk úgy lett megírva, hogy $x, y \in [-1; 1]$ és $z \in [0; 1]$. A perspective divide előtt (w-vel osztás) a koordinátáink:

$$
\begin{aligned}
-w \leq x \leq w \\
-w \leq y \leq w \\
0 \leq z \leq w
\end{aligned}
$$

Így a feltételeink a kocka különböző oldalaira:

Jobb oldali sík: $x \leq w$, $0 \leq w - x$	**Felső oldal síkja:** $y \leq w$, $0 \leq w - y$	**Távoli oldal síkja:** $z \leq w$, $0 \leq w - z$
Bal oldali sík: $-w \leq x$, $0 \leq x + w$	**Alsó oldal síkja:** $-w \leq y$, $0 \leq y + w$	**Közeli oldal síkja:** $0 \leq z$

Ezek a feltételek egyben megadják a síktól való távolságot is, amikkel ki tudjuk számolni a metszéspontokat a két pont interpolálásával.

3. Pszeudokód

Az alap Sutherland-Hodgman algoritmus:

Algoritmus 1 Sutherland-Hodgman algoritmus

```
kimenetiLista:=eredetiSokszog
Ciklus minden vagasiEl ∈ vagoSokszog csináld
    bemenetiLista:=kimenetiLista
    kimenetiLista:=∅
    Ciklus i:=1 tól bemenetiListaHossz ig csináld
        aktualisPont:=bemenetiLista[i]
        elozoPont:=bemenetiLista[(i − 2) mod bemenetiListaHossz + 1]
        metszesPont:=SzamolMetszespont(elozoPont, aktualisPont, vagasiEl)
        Ha aktualisPont belül van vagasiEl akkor
            Ha elozoPont nincs belül vagasiEl akkor
                kimenetiLista.hozzaad(metszesPont)
            Elágazás vége
            kimenetiLista.hozzaad(aktualisPont)
        Különben Ha elozoPont belül van vagasiEl akkor
            kimenetiLista.hozzaad(metszesPont)
        Elágazás vége
    Ciklus vége
Ciklus vége
```

Források

Ivan E. Sutherland and Gary W. Hodgman. Reentrant polygon clipping. *Commun. ACM*, 17(1): 32–42, January 1974. ISSN 0001-0782. doi: 10.1145/360767.360802. URL https://doi.org/10.1145/360767.360802.