Depth/Z-buffer

Algoritmo di occlusion culling. Durante la rasterizzazione si calcola per ogni pixel la distanza dall'osservatore, e si scrive il frammento corrispondente nel buffer solo se non c'è già un frammento associato ad una distanza minore.

Si può implementare nella pipeline nell'*output combiner*, e aggiungendo il depth buffer (della stessa dimensione dello schermo). Il fragment shader può modificare il depth buffer, ma è inefficiente: se non lo fa è possibile scartare i frammenti prima di eseguirne la shader. È preferibile renderizzare a partire dalle primitive più vicine, visto che modificare la profondità memorizzata è più costoso di mantenerla.

Z-fighting

I valori nel depth buffer sono numeri a virgola *fissa* (equidistanziati con incrementi di $1/(2^{32}-1)$, oggi $n = 24$ o 32 bit) tra 0 e 1 (configurabile).

Si ha Z-fighting quando due poligoni producono frammenti con valori di profondità uguali. Visto che le coordinate in view space sono in virgola mobile (densità di valori rappresentabili maggiore intorno a 0), frammenti che sono a profondità diverse in view space posso non esserlo nel depth buffer.

Applicando la matrice di proiezione in prospettiva, si ha che le profondità in clip space non sono distribuite uniformemente, ma hanno l'andamento di $1/z$ – la stessa distanza viene schiacciata di più se è più vicina al far plane, e l'80% dei valori rappresenta il 30% più vicino della profondità. Ciò significa che gli oggetti più lontani soo più suscettibili a z-fighting.

Per ridurre la possibilità di Z-fighting è necessario scegliere valori opportuni per *near* (non troppo vicino) e *far*. Per renderizzare grandi profondità si lavora a blocchi, incrementando near e far (che devono essere piani di separazione).

Polygon offset

Permette di cambiare i valori di profondità per far vincere una primitiva in caso di collisioni. L'offset è:

$$\text{factor} \cdot \text{slope} + \text{units} \cdot r,$$

dove r è la distanza tra due valori di profondità vicini. Si tiene in considerazione la pendenza della primitiva. In genere, factor = units = 1 dà un buon risultato.