

Verkefni 1

Tölvugrafík

Óskar Þórir Bjarnason, otb2@hi.is

1)

OpenGL (og WebGL) geta aðeins teiknað þríhyrninga. Útskýrið hvernig hægt væri að skilgreina fylltan Sívalning eingöngu með þríhyrningum í þrívíðu rúmi. Rissið upp mynd af sívalningi og útskýrið hvernig hann væri myndaður úr þríhyrningum. Til að fá nokkuð góða nálgun á sívalningi hversu marga þríhyrninga haldið þið að þurfi (bara stærðargráður: tugi, hundruð, þúsundir,...)? Rökstyðjið í nokkrum orðum.

SVAR:

Fyrsta spurningin sem við þurfum að spyrja okkur er hvernig við myndum tvívíðan hring úr þríhyrningum. Jafna hrings lýsir því hvernig má skilgreina hring út frá gefnum radíus r og punktastafni $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ þar sem n er fjöldi punkta. Jafna hrings með miðju (h, k) er eftirfarandi:

$$(x - h)^2 + (y - k)^2 = r^2$$

Fyrir punktinn (x, y) .

Látum miðju (h, k) vera punktinn $(0, 0)$ og fáum

$$x^2 + y^2 = r^2$$

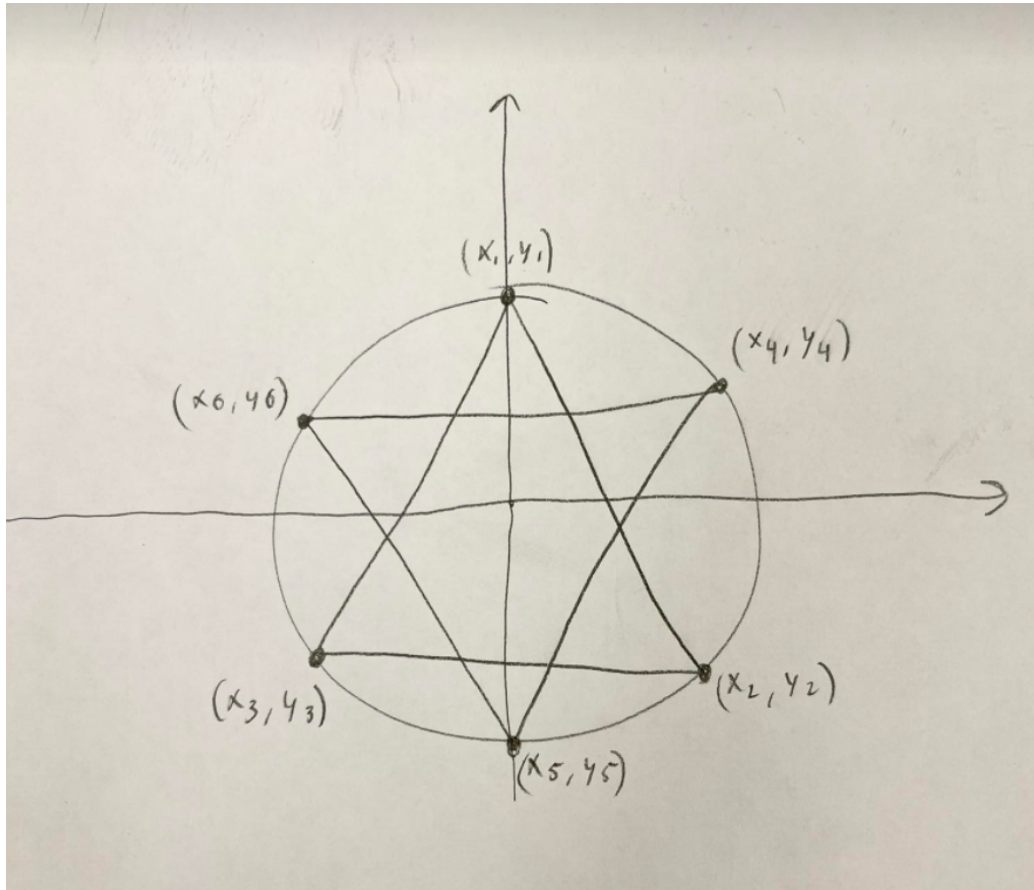
Þessi jafna lýsir sambandi punktsins (x, y) við radíus og miðju $(0, 0)$. Þ.e.a.s. punkturinn (x, y) er í fjarlægð r frá miðju $(0, 0)$. Til að mynda hring þurfum við því að mynda punktastafn sem uppfyllir þessa jöfnu, þ.e.a.s. þeir punktar sem eru í fjarlægð r frá miðju $(0, 0)$. Í WebGL er hægt að velja úr velja úr þremur þríhyrningsföllum. `gl.TRIANGLES`, `gl.TRIANGLE_STRIP` og `gl.TRIANGLE_FAN`.

`gl.TRIANGLES` fallið býr til þríhyrninga úr þremur punktum hverju sinni þ.e.a.s. Ef við værum með punktastafnið $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ þá yrði fyrsti þríhyrningurinn teiknaður af punktum $\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\}$ og sá næsti væri teiknaður af $\{(x_4, y_4), (x_5, y_5), (x_6, y_6)\}$ o.s.frv.

`gl.TRIANGLE_STRIP` fallið býr til þríhyrninga úr fyrstu þremur punktum, næsti þríhyrningur væri síðan teiknaður af öðrum, þriðja og fjórða punktinum þ.e.a.s. fyrsti þríhyrningurinn væri teiknaður af punktum $\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\}$, sá næsti væri teiknaður af $\{(x_2, y_2), (x_3, y_3), (x_4, y_4)\}$ o.s.frv.

`gl.TRIANGLE_FAN` fallið býr til svokallaðan þríhyrninga blævæng. Fyrsti þríhyrningurinn yrði myndaður af $\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\}$, sá næsti af punktum $\{(x_1, y_1), (x_3, y_3), (x_4, y_4)\}$ og síðan $\{(x_1, y_1), (x_4, y_4), (x_5, y_5)\}$ o.s.frv.

Notum `gl.TRIANGLES` í þessu dæmi. Eftirfarandi mynd sýnir punktastafn $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5), (x_6, y_6)\}$ sem uppfyllir jöfnu hrings.



Þarna eru þríhyrningarnir $\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\}$ og $\{(x_4, y_4), (x_5, y_5), (x_6, y_6)\}$ teiknaðir inn á. Með því að fjölga punktum yrði þetta safn þríhyrninga nær því að verða hringur. Þegar kemur að því að gera þennan hring að sívalningi þarf einungis að bæta við z-hniti. Jafna hings yrði þá

$$x^2 + y^2 + z^2 = 3r^2$$

Fyrir punktastafn $\{(x_1, y_1, z_1), (x_2, y_2, z_2), \dots, (x_n, y_n, z_n)\}$ með miðju $(0,0,0)$. Fyrst yrði myndaður tvívíður hringur með aðferðinni sem lýst er að ofan. Þá væri z-hnitið þeirra punkta 0. Þegar væri búið að teikna þann hring þá væri z-hnitið aukið um eitthvað ákveðið magn (sennilega örlítið) og sami hringur teiknaður. Þessu yrði ítrað þangað til að z-hnitið yrði það sama og hæð sívalningsins.

Þegar kemur að ákveða um það bil fjölda þríhyrninga til þess að leysa þetta verkefni er auðvelt að sjá að því fleiri þríhyrningar yrðu teiknaðir því nákvæmari yrði sívalningurinn. Fjöldi þríhyrninga háður hæð sívalningsins en ekki radíus hans. Til þess að fá grófa hugmynd fyrir fjöldanum skulum við setja upp einfalt dæmi. Segjum að við teiknum punkt á 1° fresti. Fyrir tvívíðan hring yrði það 360 punktar sem myndu gefa $360/3=120$ þríhyrninga. Segjum að hæðin sé og við látum z-hnitið stækka um 0,01 í hvert skipti. Það yrðu þá 100 tvívíðir hringir með 120 þríhyrninga, allt í allt þá $120 \cdot 100 = 12.000$ þríhyrningar þannig að þegar er spurt um tugi, hundruða eða þúsunda þá má með sjá af þessu dæmi að sennilega eru þessir þríhyrningar taldir í þúsundum.

2)

Kynnið ykkur eitt af öflugustu grafíkkortunum fyrir borðtölvur í dag Nvidia GeForce RTX 3090 (aðeins ítarlegri upplýsingar á Wikipedia) og finnið út eftirfarandi gildi:

A. Innri bandvidd (memory bandwidth), þ.e. flutningshraði milli grafíkminnis og GPU

SVAR: 936 GB/s

B. Litahraði skjápunkta (pixel fill rate), þ.e. hversu marga skjápunkta er hægt að lita á sek

SVAR: 156,2 GP/s

C. Ytri bandvidd (bus interface bandwidth), þ.e. bandvidd tengibrautar (PCI-E 4.0, sjá t.d. hér)

SVAR: 16 GB/s

D. Hámarks skjáupplausn (max digital resolution) og hvaða sjónvarps/skjá staðall er það?

SVAR: 8K UHD (7680 X 4320)

3)

Skoðið aðeins nýja grafíkforritasafnið Vulkan og nefnið helstu kosti þess umfram OpenGL(sjá t.d. líka á Wikipedia). Þetta eiga bara að vera 3-5 setningar, ekki ritgerð!

SVAR: Vulkan gerir tölvunni kleift að nota minna CPU en OpenGL fyrir sömu verkefni. Einnig gefur Vulkan meiri stjórn yfir GPU heldur en OpenGL. Í staðinn fyrir að þýða litarana á keyrslutíma líkt og í OpenGL þá þýðir Vulkan litarana á undan á svokallað SPIR-V form. Þetta leiðir til þess að forritið á auðveldara með að ræsa sig og hægt er að nota fleiri litara í hvert skipti.

4)

Khronos samtökin halda utan um ýmsa fleiri opna staðla sem tengjast grafík og grafík örgjörvum. Skoðið aðeins staðlana glTF og Open XR og segið stuttlega í ykkar eigin orðum frá tilgangi þeirra.

SVAR:

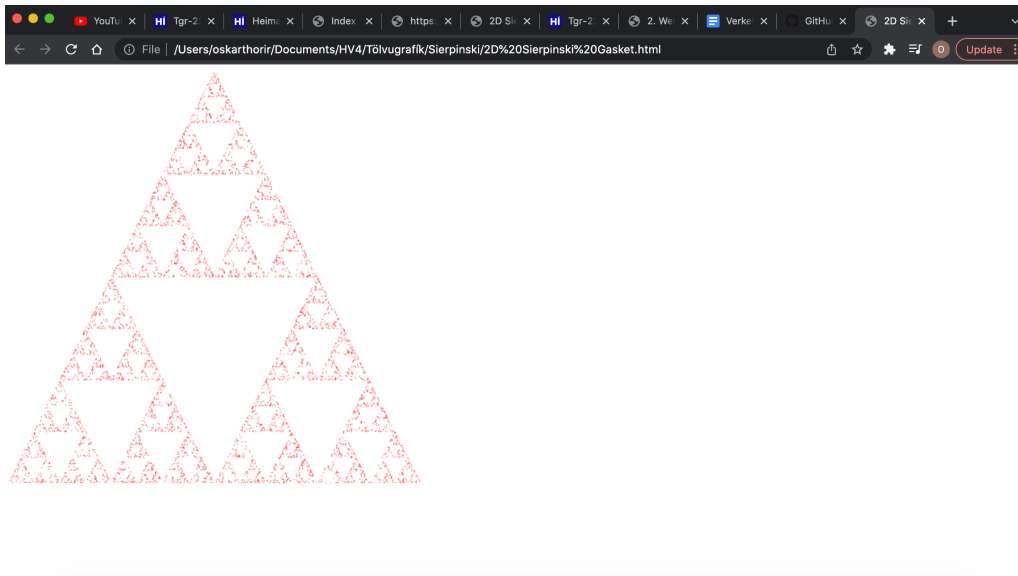
glTF er staðall sem leggur áherslu á að lágmarka stærð þrívíðra hluta og tímann sem tekur til þess að opna þá og keyra.

Open XR staðalinn snýst um að veita hágæða aðgang að sýndaveruleika (e. virtual reality) og gagnauknum veruleika (e. augmented reality).

5)

Náið í útgáfu 1 af þéttlista Sierpinskis og fáið hana til að virka á tölvunni hjá ykkur. Þið þurfið einnig að ná í Javascript skrána (gasket1.js) og skrána rwebgl-Utils.js, initShaders.js Og MV.js sem eru í möppunni Common. Fjölgið punktunum sem eru teiknaðir svo að myndin verði skarpri. Skilið mynd af útkomunni.

5.000 punktar:



100.000 punktar:

