

Computergrafieken: Kraan

Waldo Wautelet

May 27, 2022

1 Input-toetsen

g : animatie last (horizontaal)
G : animatie last (verticaal)
r : animatie roteren
p : algemeen perspectief mode
i : specifiek perspectief mode
o : orthogonaal mode
n : een extra kraan tekenen
N : een kraan verwijderen (indien meer dan 1)
x : plus 1 op x-coordinaat
X : min 1 op x-coordinaat
y : plus 1 op y-coordinaat
Y : min 1 op y-coordinaat
z : plus 1 op z-coordinaat
Z : min 1 op z-coordinaat
j : laat de x, y, z-as zien
l : laat draad zien in bezier vlakken
a : ambient wit licht aan/uit zetten
b : diffuse groen-blauw licht aan/uit zetten
c : rood speculair licht aan/uit zetten
d : geel spot aan/uit zetten
v : spothoek vergroten
V : spothoek verkleinen
w : exponent spot vergroten
W : exponent spot verkleinen
e : shininess vergroten met 5
E : shininess verkleinen met 5
h : spot hoger plaatsen (op y-as)

H : spot lager plaatsen (op y-as)
k : controlepunten bezier-vlakken aan/uit zetten
s : smooth shading aan/uit zetten
f : doorzichtigheid beziervlak aan/uit zetten
m : mist aan/uit zetten
M : exponent/linear aan zetten (mist)

default : alle nummertjes tussen 0-9 laten elk enkel een deel zien van de kraan

Ik heb bij de input een bewuste keuze gemaakt wegens een mogelijks tekort aan toetsen. Ik heb gekozen om gebruik te maken van booleans om sommige variabelen te behandelen zoals bijv. de animaties. Hierdoor moet ik maar 1 toets gebruiken om aan en uit te schakelen. Deze methode gebruik ik bij volgende variabelen:

- animatie last (zowel horizontaal als verticaal)
- animatie rotatie
- draad laten zien
- alle lichten
- controlepunten laten zien
- smooth shading
- doorzichtigheid
- alles rond mist
- alle onderdelen apart laten zien

Bij het ingeven van de displaymodussen heb ik gebruik gemaakt van 2 coördinaten, omdat de frustum methode rare coördinaten gebruikt die ik niet zomaar kon gebruiken van de andere.

2 Ontwikkelde methodes

In deze sectie wordt beschreven welke methodes er zijn en wat welke doet.

```
void init(void);
```

In de init functie zit alle standaard code in die 1 keer moet uitgevoerd worden en daarna niet meer. Hier zit een clear in van het scherm, de camera word geplaatst en alle lichten krijgen kun kleuren.

```
void winReshapeFcn(GLint w, GLint h);
```

Deze functie word opgeroepen wanneer het venster verandert van grootte, hierin word een keuze gemaakt welk perspectief we willen gebruiken.

```
void displayFcn(void)
```

Deze functie tekent alle onderdelen, plaatst de lichten (indien nodig) en tekent mist indien aangezet. Hierin wordt ook drawKraan opgeroepen (indien nodig meerdere keren met een translatie ertussen).

```
void drawKraan();
```

De drawKraan functie tekent alle delen van de kraan indien deze aan staan.

```
int main(int argc, char* argv[]);
```

De main functie bevat alle standaard initialiseringen voor OpenGL, ook word er een timer functie doorgegeven en ten slotte nog een toetsenbord functie.

```
void toetsPress(unsigned char toets, int x_muis, int y_muis);
```

Alle opties van toetsinvoer worden doorlopen en indien nodig aangepast, ik maak hier gebruik van een switch.

```
void drawBalk();
```

Deze functie tekent enkel de 4 verticale balken als een gevulde kubus en met lijntjes.

```
void drawAssen();
```

Een functie die de 3 hoofdassen tekent in elk een ander kleurtje.

```
void drawVakwerk(char vlak);
```

Deze functie bevat alle logica om 1 van de vakwerken te maken, de functie krijgt als variabelen slechts 1 character mee om te bepalen welk vakwerk het nu gaat tekenen. Voor elke mogelijke char word dan een rotatie en of translatie toegepast.

```
void drawCylinderVast();
```

Een simpele functie om de onderste vaste cylinder te tekenen.

```
void drawCylinderLos();
```

Een simpele functie om de bovenste losse cylinder te tekenen (die meedraait met het verticale gedeelte).

```
void drawTopPart();
```

Deze functie bevat alle onderdelen die aan het verticaal deel hangen en dus mee geroteerd moeten worden. Het assenstelsel word verlegt om bovenop de losse cylinder te liggen.

```
void drawArm();
```

De functie om de arm (verticaal deel) te tekenen inclusief 2 vakwerken die geroteerd zijn.

```
void drawVakwerkArm();
```

Een specifieke functie om het vakwerk van de arm te tekenen.

```
void drawBalDing();
```

Deze functie tekent de last, het touw en het ophangstukje. Hier word rekening gehouden met de animaties en rotaties die mogelijk kunnen plaatsvinden tijdens het tekenen.

```
void drawKabine();
```

De functie die alles rond de kabine tekent, inclusief het glazen voorpaneel. Ook de spot word hier gedefinieerd (wegens het makkelijk zijn van coördinaten).

```
void drawGewichten();
```

De functie om de 3 gewichten te tekenen. De voorkant van de gewichten zijn dezelfde beziervlakken als het glas van de kabine.

```
void animFunc(int delta);
```

De functie die alles behandelt van animaties, hier is een simpele if structuur genoeg om te bepalen welke variabelen met welke getallen opgeteld moeten worden.