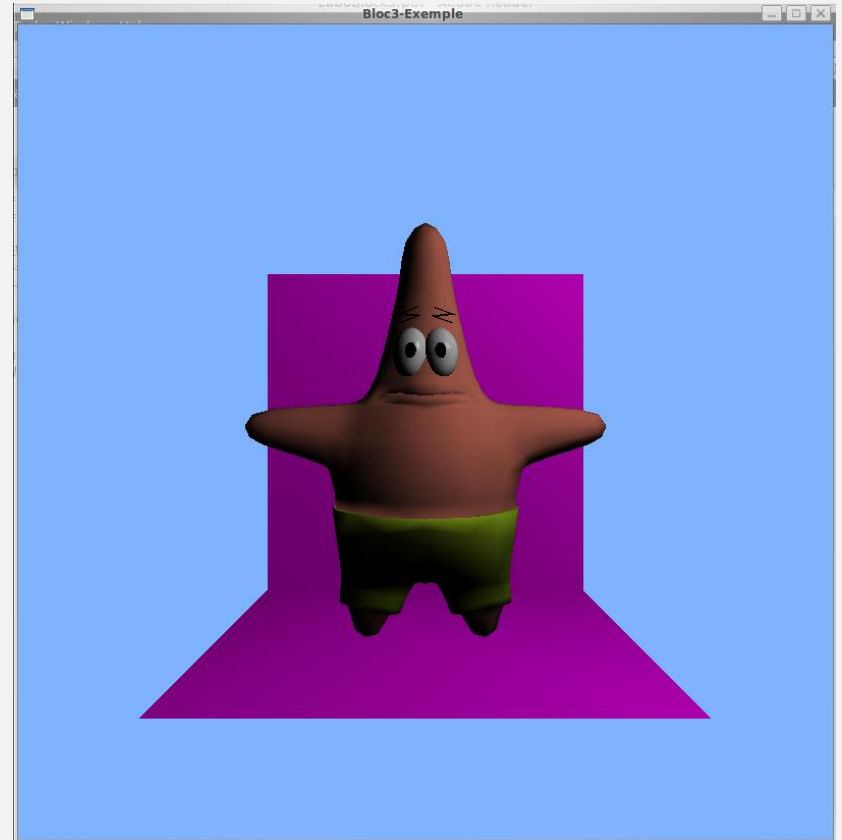
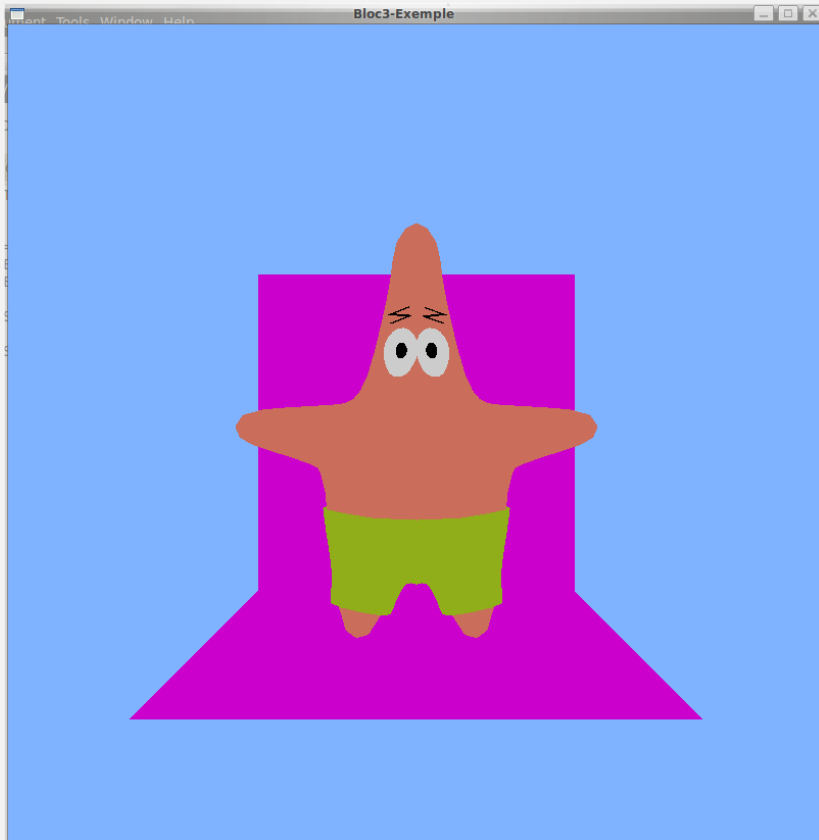


Laboratori OpenGL – Sessió 3.2

Realisme - Il·luminació:



Posició del focus de llum

Relativa a:

- L'escena – la posició del focus en SCA
 - Posició fixa del focus respecte a l'escena
 - Multiplicar `posFocus` per `view Matrix` per tenir-la en SCO
- La càmera – la posició del focus en SCO
 - Posició fixa respecte a la càmera
 - `posFocus` ja està en SCO directament
- Un model – la posició del focus en SCM
 - Posició fixa respecte al model d'un objecte
 - Multiplicar `posFocus` per $(view * TG)$ igual que al model

Cal tenir en compte

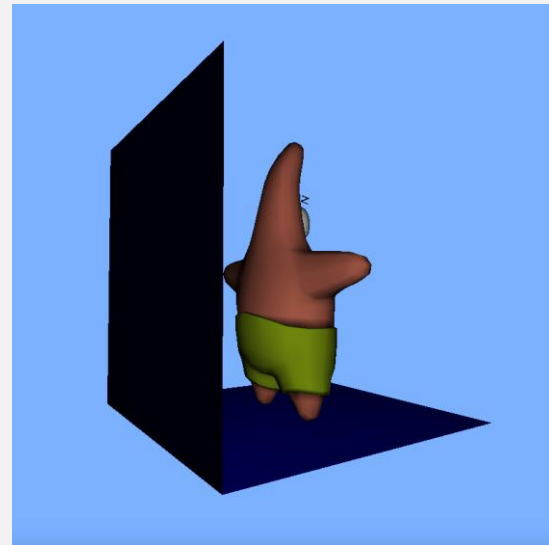
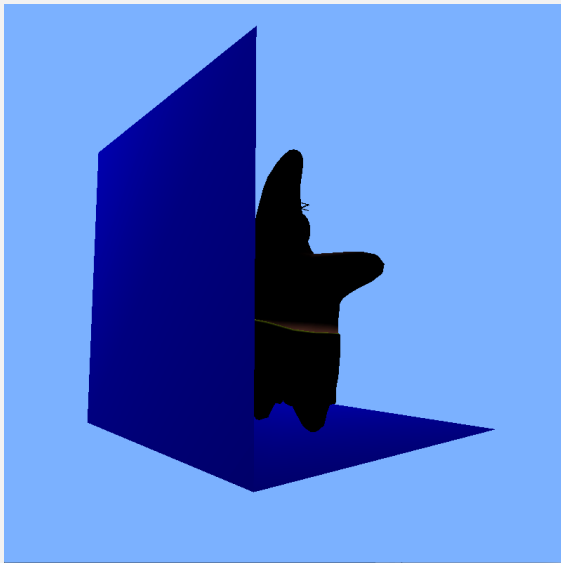
Quan els càlculs es fan en el MyGLWidget:

- Cada cop que es **modifica la viewMatrix**:
 - **Recalcular posFocus** si va multiplicada per viewMatrix
 - **Recalcular NormalMatrix** si es té calculada en MyGLWidget
- Cada cop que es **modifica la modelMatrix (TG)**:
 - **Recalcular NormalMatrix** si es té calculada en MyGLWidget
 - **Recalcular posFocus** si va multiplicada per TG

Exercici 1

Fer que el focus de llum sigui un focus de càmera:

- Que sempre estigui en la posició donada i que sigui relativa al SCO
- Provar-ho fent girar la càmera i observant les diferències amb un focus d'escena



Procés de visualització: Phong Shading i càlcul color en FS

Tenim definit TG, VM, PM com a uniforms, ara el que

Uniforms: TG, VM, PM

Vertex Shader

$V_M, N_M,$
 $k_{a\lambda}, k_{d\lambda}, k_{s\lambda}, n$



- Càlcul coordenades de clipping
- Càlcul de N_o i V_o

V_c $k_{a\lambda}, k_{d\lambda}, k_{s\lambda}, n$
 N_o, V_o

El que normalitzem al FS, quan s'interpolava al arribar al

Uniforms: Info Llums (pos en SCO, color)

Fragment Shader

- Clipping
- Perspective Division
- Device Transform
- Back-Face Culling

Rasterització
interpolació

V_D $k_{a\lambda}, k_{d\lambda}, k_{s\lambda}, n$
 N_o, V_o

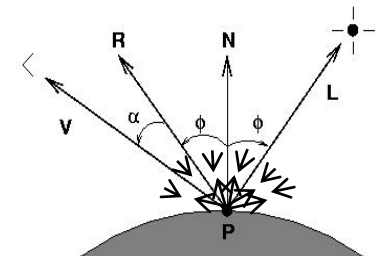
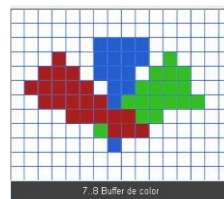
$$cf = I_{a\lambda} k_{a\lambda} + \sum_i (I_{f_i\lambda} k_{d\lambda} \cos(\Phi_i)) + \sum_i (I_{f_i\lambda} k_{s\lambda} \cos^n(\alpha_i))$$

FragColor= cf

$\{(x_f, y_f, z_f, k_{a\lambda f}, k_{d\lambda f}, k_{s\lambda f}, n_f, N_{of}, V_{of})\}$

$\{(x_f, y_f, z_f, \text{FragColor})\}$

Z-buffer



Tot ha d'estar normalitzat i en SCO!!!

Exercici 2

Càlcul color en el Fragment Shader:

- Passar les funcions `Ambient`, `Difus` i `Especular` al FS
- Fer que hi arribin les dades necessàries des del VS:
 - Posició del vèrtex en SCO
 - Normal al vèrtex en SCO
 - Propietats del material (`matamb`, `matdiff`, `matspec`, `matshin`)
- Uniforms amb les dades de llum ambient i focus de llum al FS

Els vectors normalitzats en el VS no arriben normalitzats al FS
(després de la interpolació)

Recordeu que els atributs no es poden modificar en el shader

Exercici 3

Implementa canvi entre focus de càmera i focus d'escena:

- Focus d'escena a la posició (1,1,1) en SCA
- Focus de càmera a la posició exacta de la càmera
- Usa la tecla 'F' per a fer el canvi entre els dos tipus de focus

Executables de mostra:

Tecla 'O':

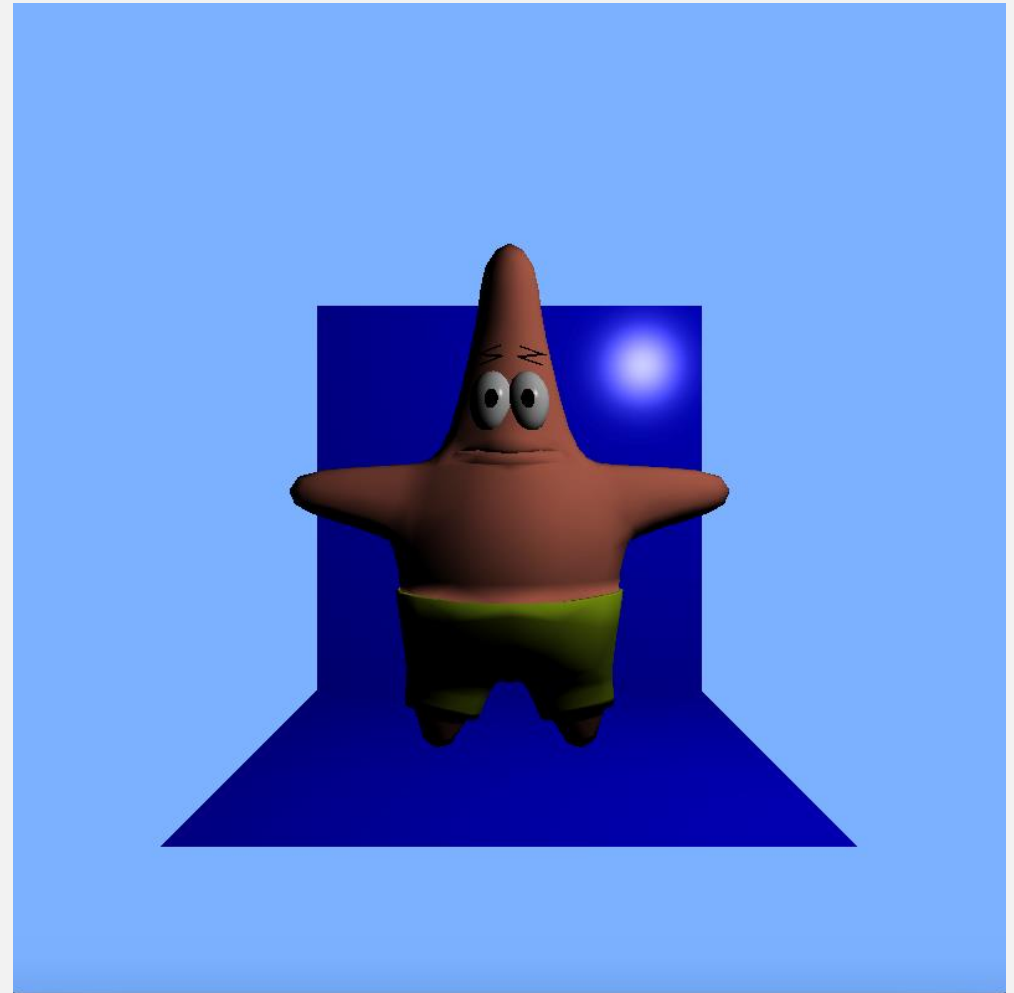
Canvia d'òptica

Tecla 'F':

Canvia de focus
de posició (1,1,1) en SCA
a focus de càmera.

Tecles 'K' i 'L':

Mouen focus sobre eix X



➤ ~/assig/idi/bin/Bloc3_Sol_3.2.1 (exercici 1 de sessió 3.2)

➤ ~/assig/idi/bin/Bloc3_Sol_3.2.2 (exercici 2 de sessió 3.2)