

Gràfics i Visualització de Dades

Tema 3c: Exercicis resolts

Anna Puig

Pregunta 1

1. Quan s'apliquen textures a un objecte amb mapping directa:

a. L'ordre de les transformacions és: funció de correspondència, funció de projecció, funció de transformació del valor i aplicació del valor de la textura a alguna de les propietats que s'usen a la fórmula de Blinn-Phong

b. L'ordre de les transformacions és: funció de projecció, funció de correspondència, funció de transformació del valor i aplicació del valor de la textura a alguna de les propietats que s'usen a la fórmula de Blinn-Phong

c. No s'utilitza Blinn-Phong sinó directament el valor de la textura

d. L'ordre de les transformacions és: funció de projecció, funció de correspondència i directament l'aplicació del valor de la textura a alguna de les propietats que s'usen a la fórmula de Blinn-Phong

Justificació: Veure transparència 10 del Tema 3b

Pregunta 2

2. Quan s'apliquen textures a un objecte:

a. Si es realitza mapping directa, la funció de projecció – que passa de coordenades (u,v) a texels – permet realitzar simetries, repeticions i retallats de la textura sobre una cara de l'objecte

b. Si s'aplica bump-mapping és necessari aplicar de nou el test de visibilitat per tal de calcular bé l'eliminació de parts amagades.

c. Si s'aplica un mapping en dues fases utilitzant una esfera, cal realitzar dues transformacions en la funció de transformació del valor.

d. La tècnica de displacement-mapping modifica només el valor de la normal a cada punt amb el valor de la textura.

El Bump-mapping no altera la silueta dels objectes i per tant no cal tornar a fer el test de visibilitat dels objectes.

Justificació: Veure transparència 15 del Tema 3b. La funció de projecció, està formada per dues funcions A i A' . És la funció A' la que permet fer les simetries, repeticions i retallats de la textura quan el mode `GL_WRAP` està activat.

Pregunta 3

3. Suposa la textura següent (Figura (a)) que s'ha mapejat en un quadrat. En moure dos dels vèrtexs del quadrat, es distorsiona la textura de la forma de la Figura (b). Per a què pot ser degut?

La textura inicial no distorsionada com es veu a la figura (a)

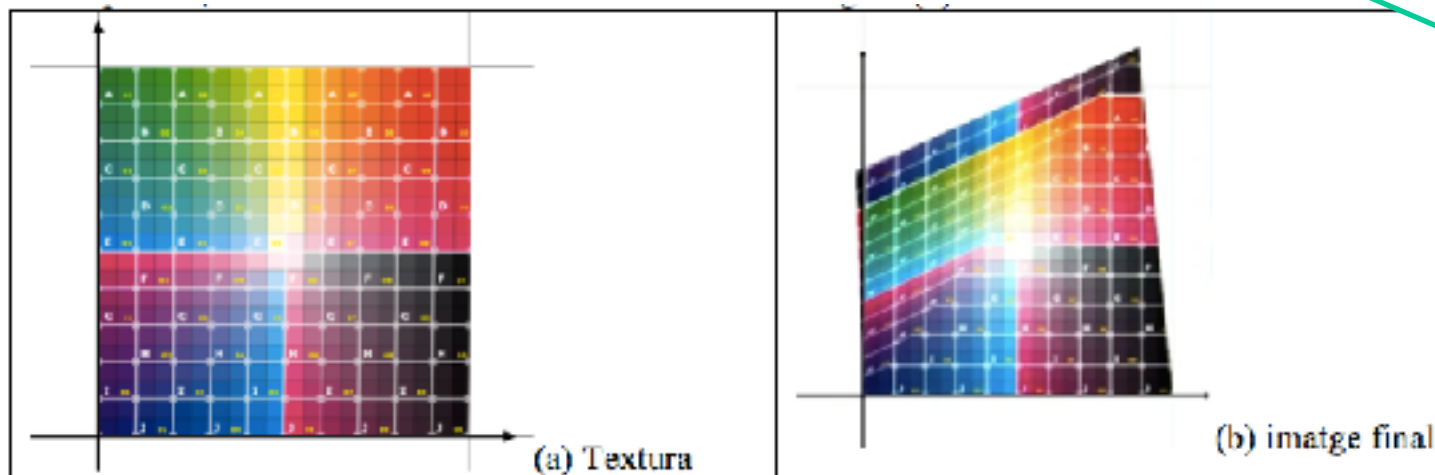
a. La textura està distorsionada ja abans de ser aplicada

b. Les coordenades de textura estan mal aplicades

c. El quadrat està subdividit en molts triangles petits i semblants entre si

De fet, la solució és un barreja de la resposta b i d. Veure transparència següent.

d. El quadrat està format per només dos triangles



Si el quadrat estés format per triangles molt petits i semblants entre si, la textura no distorsionaria globalment sino que només a la regió en hi ha el moviment

Pregunta 3

Suposa que tens un quadrat format per dos triangles on es mapeja la textura amb les coordenades de textura (s, t) definides convencionalment:



Quan es mouen alguns dels vèrtexs del quadrat de forma que els triangles queden desproporcionats, dóna aquest efecte.



Això passa per que els triangles han canviat de forma diferent en l'espai 3D sense variar les seves coordenades ST, fet que produeix la distorsió. Les transformacions afins que han patit els dos triangles son diferents, mentre que el mapeig a l'espai de textura ha quedat inalterat. Per tant, la distorsió es produïda tan per que no s'han definit bé les coordenades de textura com pel fet que són dos triangles. Si el quadrat estés format per més de dos triangles, les transformacions entre triangles es veurien més contínues i la distorsió es veuria més gradual

Pregunta 4

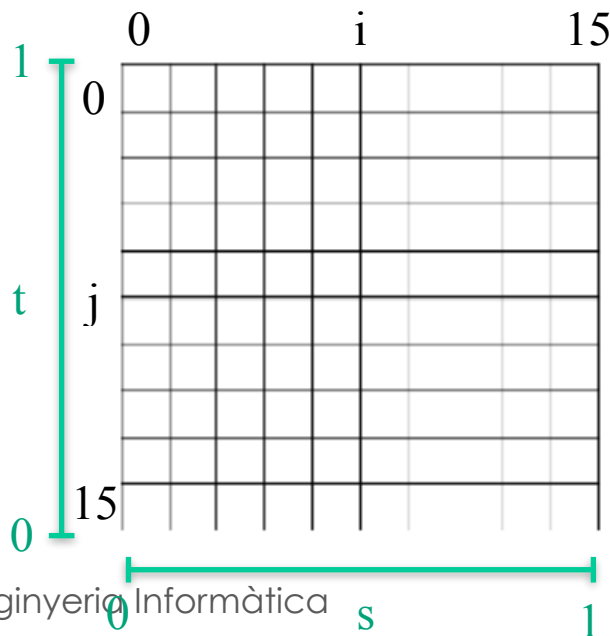
4. S'han calculat els valors (s, t) amb la funció de projecció, quins valors donaria la funció de correspondència a (i, j) corresponents al punt $(s, t) = (0.72, 0.40)$, del sistema expressat a la figura?

a. $(11, 6)$

b. $(11, 9)$

c. No es poden calcular els valors sense saber si es fa GL_NEAREST o GL_LINEAR

d. $(10, 9)$



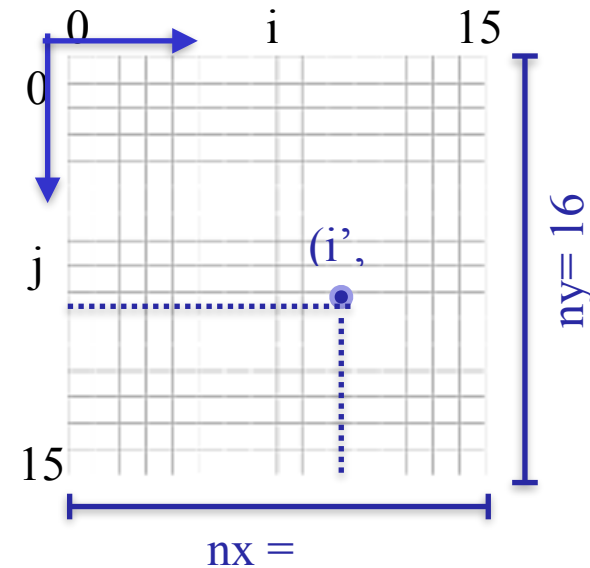
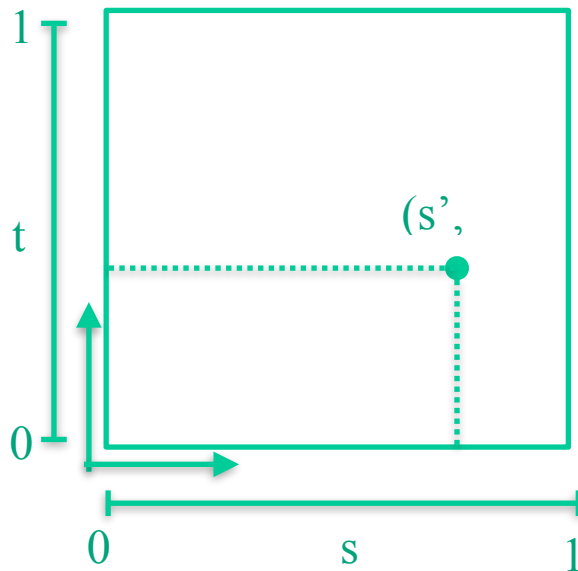
Justificació: Cal tenir en compte que l'origen del sistema (s, t) es troba a l'esquerra i l'origen del sistema (i, j) és a la part superior esquerra.

$$i: 0,72 * 16 = 11,52 \Rightarrow i = 11$$

$$j: 16 - 0,4 * 16 = 9,6 \Rightarrow j = 9$$

veure transparència següent per més explicació

Pregunta 4



La relació dels dos sistemes de coordenades és la següent:

$$\frac{s'}{1} = \frac{i'}{nx} \quad \text{d'aquí, aïllant } i', \text{ queda } i' = (\text{int})(s' * nx) \Rightarrow i' = (\text{int})(0,72 * 16) = 11$$

$$\frac{t'}{1} = \frac{ny - j'}{ny} \quad \text{d'aquí, aïllant } j', \text{ queda } j' = (\text{int})(ny - t' * ny)$$

$$\Rightarrow j' = (\text{int})(16 - 0,4 * 16) = (\text{int})(9,6) = 9$$

En el cas de $(s', t') = (1.0, 1.0)$ donaria el píxel (16, 16), que representa un téxel que no és de la imatge. En aquest cas, gràcies a les tècniques de WRAP es faria un tracta de REPEAT, MIRROR, CLAMP, etc.

De totes formes, a efectes pràctics, en treballar amb floats, quasi sempre tindrem aproximacions a 16 i no 16.0 exactament, la qual cosa, ens portarà al píxel 15.

Pregunta 5

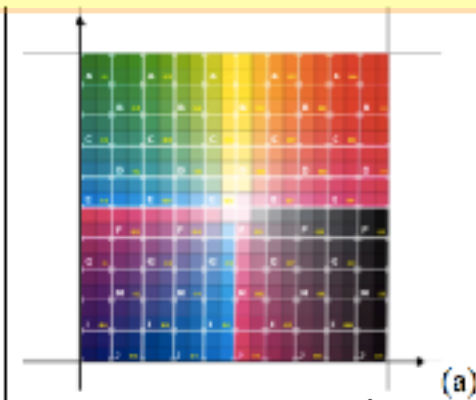
5. Com s'aconsegueix la imatge de la columna (b) partint de la textura de la columna (a)? Fixa't que es repeteix 3 cops a les dues dimensions, quines respostes són **certes** si està activat GL_REPEAT?

a. s_{max} i t_{max} són (3,3)

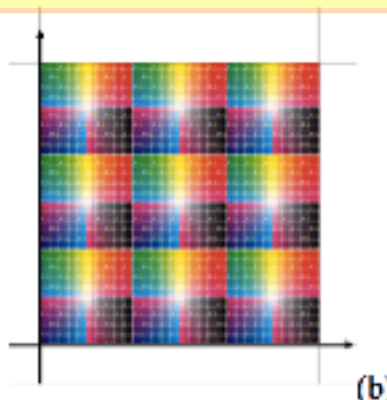
b. la funció de A' que passa de (u, v) a (s, t) es defineix com $A'(u, v) = (3*u, 3*v)$

c. la funció de A' que passa de (u, v) a (s, t) es defineix com $A'(u, v) = (0.333*u, 0.333*v)$

d. (i, j) es calculen com la part decimal de s i de t ($s\%1, t\%1$)



(a)



(b)

veure transparència següent per més explicació

Pregunta 5

Observacions inicials: Funcionament de les textures en GL en relació al mapeig de (s,t) a la textura:

- En GL, per fer la funció de correspondència de (s, t) a un téxel (i, j) de la imatge de la textura, cal que (s, t) estigui en l'interval $[(0, 0) - (1, 1))$.
- Quan els valors de (s, t) excedeixen el valor 1, si el WRAPPING està activat, es recalculen els valors de (s, t) per a portar-los a l'interval $(0, 0) - (1, 1)$.
- Concretament, quan el GL_REPEAT està activat, s'usa el la part decimal del valor de (s, t) , que és equivalent a dir $s\%1, t\%1$

Pregunta 5

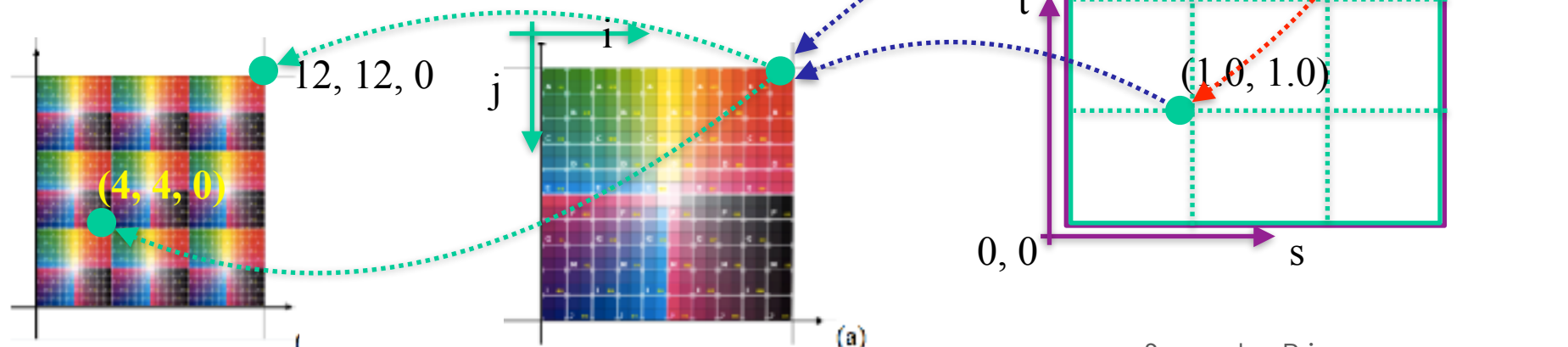
Donat un quadrat on es vol posar la textura, per exemple el quadrat de vèrtexs extrems $(0, 0, 0)$ i $(12, 12, 0)$ en coordenades de món (veure quadrat verd de la transparència)

Li associem amb la funció A les coordenades u, v entre $(0, 0)$ i $(1, 1)$.

La funció A' mapejarà els valors de (u, v) entre $(0,0)$ i $(1, 1)$ a valors (s, t) entre $(0,0)$ i $(3, 3)$. Aquí $s_{\max} = 3$ i $t_{\max} = 3$

En tenir activat el `GL_REPEAT`, quan s'accedeixi amb els (s, t) a la textura (i, j) amb la funció B , s'accedirà a la textura amb la part decimal de s i t

Proveu a comprovar com es calculen les diferents coordenades dels punts inicials del quadrat $(4, 4, 0)$ i $(12, 12, 0)$

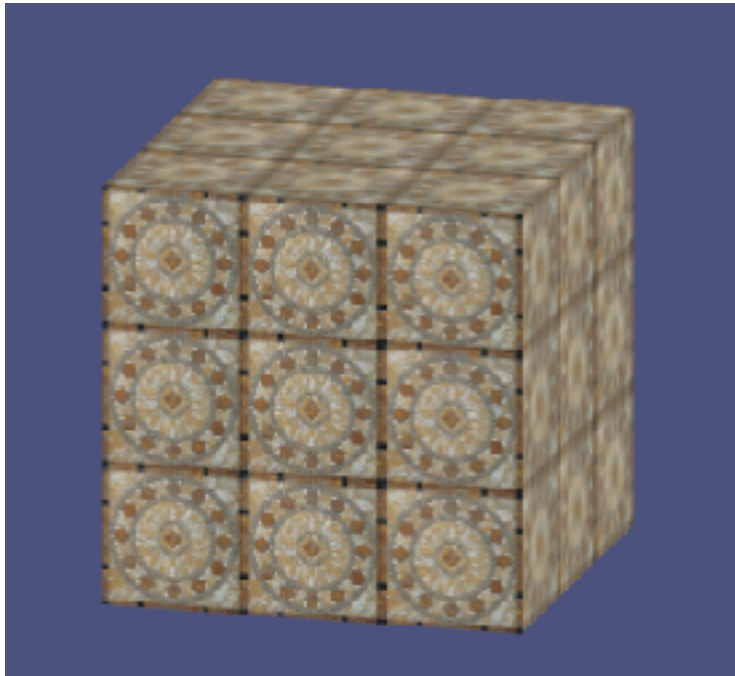


Visualització final del quadrat verd inicial

Textura

Pregunta 5

Si ho voleu comprovar podeu agafar el projecte de cubTextures i transformar les coordenades de textures multiplicant-les per 3 o per 0.33, per a veure els diferents efectes



$$A'(u, v) = (3*u, 3*v)$$



$$A'(u, v) = (0.333*u, 0.333*v)$$

Pregunta 5

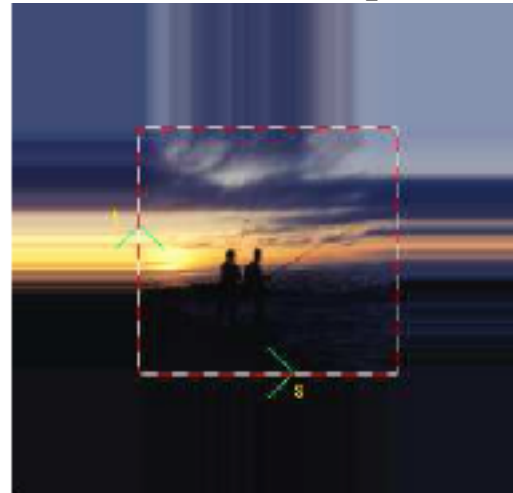
Com seria la funció A' en el cas dels altres efectes de WRAP? com ara mirall, clamp o marc?

Clamp

Mirall



Textura inicial



Marc



Pregunta 6

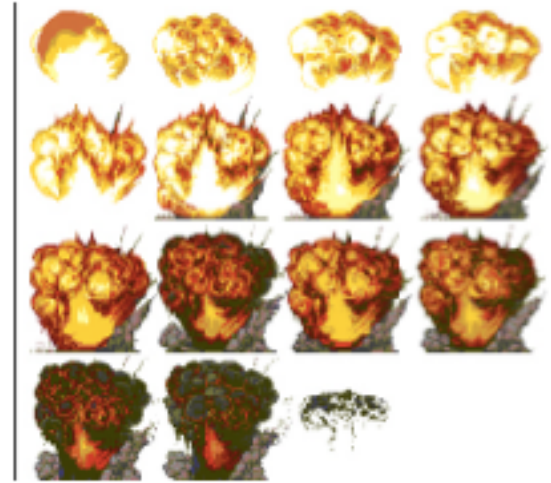
6. Es volen utilitzar textures en GL per a simular una explosió d'un projectil. Per això es visualitza un pla quadrat a la pantalla i a cada frame se li aplica un tros de textura. Les diferents imatges de l'animació de l'explosió es guarden en una única textura de de dimensió 1024x1024. Quina de les següents afirmacions és certa?

a. S'utilitzarà una funció de projecció diferent a cada frame i es passaran les noves coordenades de textures al fragment shader.

b. S'utilitzarà una funció de correspondència a cada frame per a calcular el píxel corresponent en el fragment shader.

c. Caldrà definir un mapping en dues fases per a poder tenir en compte que les textures es projecten en un pla.

d. **Només és possible** obtenir aquest efecte partint la textura en sub-textures més petites de forma que es vagin passant al fragment shader a cada frame.

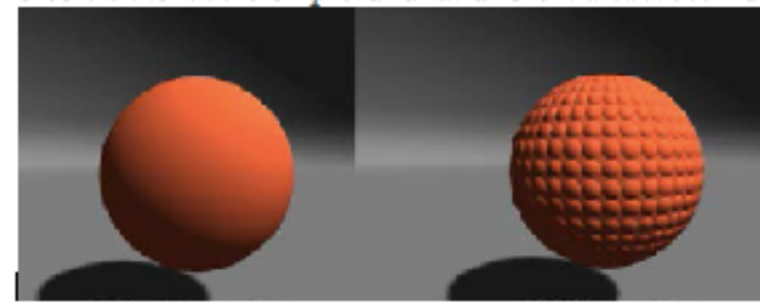


Com hem vist fins ara és la funció de projecció la que permet detallar com accedir a la textura, ja que la funció de correspondència és fixa en GL

Seria possible si no treballéssim en GL, però GL no ho permet

Es possible fer-ho així però no és la única manera

Pregunta 7



7. Es té una esfera de color vermell que s'ha visualitzat segons Phong shading en la imatge de l'esquerra. Quin mètode s'ha fet servir per a calcular la imatge de la dreta? Per què?

a. S'ha aplicat el mètode de bump mapping substituint la normal de la fórmula de Phong-Blinn per la informació guardada en una textura ja que es veu la silueta de la projecció és la mateixa en les dues visualitzacions.

b. S'ha aplicat el mètode de **displacement mapping** que permet aplicar una textura als punts del model de l'esfera ja que la silueta de la projecció és la mateixa en les dues visualitzacions.

c. S'ha aplicat el mètode de bump mapping substituint la **component difusa** del material de la fórmula de Phong-Blinn per la informació guardada en una textura ja que es veu la silueta de la projecció és la mateixa en les dues visualitzacions.

d. S'ha aplicat el mètode de **displacement mapping** que permet substituir la distància a la llum de cadascun dels punts de la superfície pel valor guardat en una textura ja que s'està obtenint un efecte d'atenuació en profunditat diferent a cadascun dels punts de la superfície.

La resposta a. descriu exactament com és el mètode de Bum Mapping (veure transparència 30 del tema 2b)

el displacement mapping canvia les siluetes i en aquest cas, a la foto la silueta de l'esfera no canvia en relació a l'esfera visualitzada amb Phong

Pregunta 8

8. Es vol visualitzar una cara formada per quatre vèrtexs de tal manera que es vegi rugosa la seva superfície, sense canviar la seva silueta i es vol visualitzar situant la direcció de l'observador perpendicular al pla de la cara. Quin mètode de textures caldria aplicar?

a. Displacement mapping.

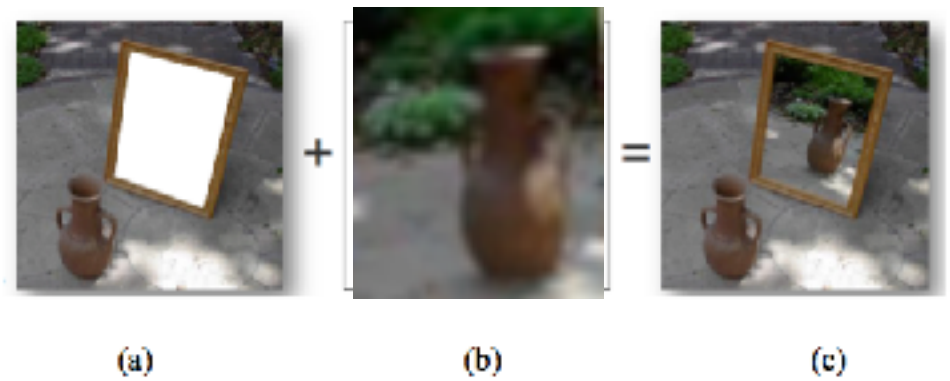
b. Bump-mapping en el vertex shader

c. Una cara de 4 vèrtexs no pot visualitzar-se amb rugositats si no es defineixen més vèrtexs.

d. Bump-mapping en el fragment shader

Per a simular rugositats sense canviar silueta és necessari fer bump mapping. Ha de ser en el fragment shader per poder fer rugositats a nivell dels píxels de la rasterització del polígon

Pregunta 9



9. S'ha obtingut una imatge via RayTracing del que reflecteix un mirall d'una escena (veure la Figura (b)). Es vol utilitzar aquesta imatge per a posar-la com a textura en el mirall (veure figura (c)) usant una visualització projectiva basada en shaders. Per això es passa la imatge (a) al fragment shader. Quina de les següents afirmacions és **falsa**?

a. Per a obtenir amb RayTracing la Figura (b) s'ha de definir l'observador al centre de l'objecte mirall per poder calcular el que veuria l'observador des del mirall.

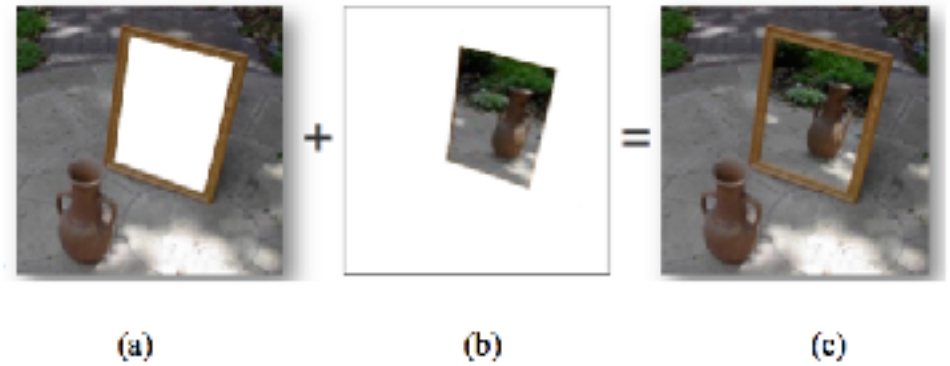
b. Per a obtenir amb RayTracing la Figura (b) cal definir el frustum de la càmera per tal que només es visualitzi la cara del mirall.

c. Les càmeres han de ser les mateixes en el RayTracing i en el Z-Buffer.

d. Cal passar la imatge obtinguda al RayTracing com si fós una textura al fragment shader i tenir les coordenades de textures normalitzades tenint en compte l'objecte que defineix el mirall.

Fixeu-vos que la imatge (b) que s'obté amb Raytracing és directament el que veu l'observador si es situés en el mirall i només es visualitza la cara del mirall. Per tant, les àmeres no poden ser les mateixes en els dos algorismes.

Pregunta 10



10. S'ha obtingut una imatge via RayTracing del que reflecteix un mirall d'una escena (veure la Figura (b)). Es vol utilitzar aquesta imatge per a posar-la com a textura en el mirall (veure figura (c)) usant una visualització projectiva basada en shaders. Per això es passa la imatge (a) al fragment shader. Quina de les següents afirmacions és **falsa**?

a. Per a obtenir amb RayTracing la Figura (b) s'ha de definir l'observador al centre de l'objecte mirall per poder calcular el que veuria l'observador des del mirall.

b. Per a obtenir amb RayTracing la Figura (b) cal només calcular el color usant Blinn-Phong en aquells píxels tals que els seus rajos primaris intersequen amb el mirall.

c. Les càmeres han de ser les mateixes en el RayTracing i en el Z-Buffer.

d. Cal passar la imatge obtinguda al RayTracing com si fós una textura al fragment shader i tenir les coordenades de textures normalitzades tenint en compte l'objecte que defineix el mirall.

Ara la imatge (b) que s'obté amb Raytracing és el tros de mirall que es veuria en el ZBuffer. En aquest cas les dues càmeres han de ser les mateixes en les dues visualitzacions i la càmera en el Raytracing ha d'estar en la mateixa posició que en el ZBuffer i no en el centre del mirall com diu l'apartat a