

Übungsaufgaben

1. Gegeben ist folgende Textur. Bestimmen Sie die Ableitung der Textur in s und t -Richtung. Verwenden Sie an den Rändern ein Padding mit 0
2. Generieren Sie aus der Textur eine Normal Map, die für das Bump Mapping eingesetzt werden kann.

Original Textur

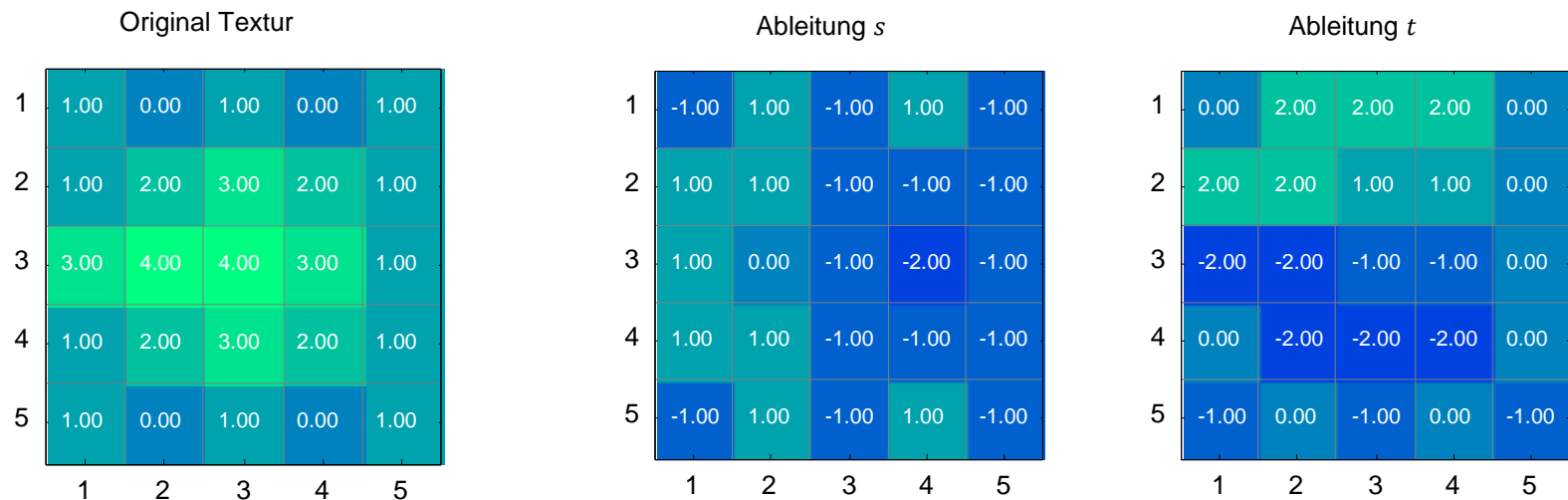
1	1.00	0.00	1.00	0.00	1.00
2	1.00	2.00	3.00	2.00	1.00
3	3.00	4.00	4.00	3.00	1.00
4	1.00	2.00	3.00	2.00	1.00
5	1.00	0.00	1.00	0.00	1.00
	1	2	3	4	5

Lösung

1) Gegeben ist folgende Textur. Bestimmen Sie die Ableitung der Textur in s und t -Richtung. Verwenden Sie an den Rändern ein Padding mit 0.

Vorgehen: Pixelweise entlang der s - (horizontal) bzw. t -Achse (vertikal) Subtraktion der Einträge bei $(s + 1)$ und s . Beispielhaft für Ableitung in s -Richtung bei $(1,1)$:

$$g(s + 1, t) - g(s, t) = 0 - 1 = -1$$



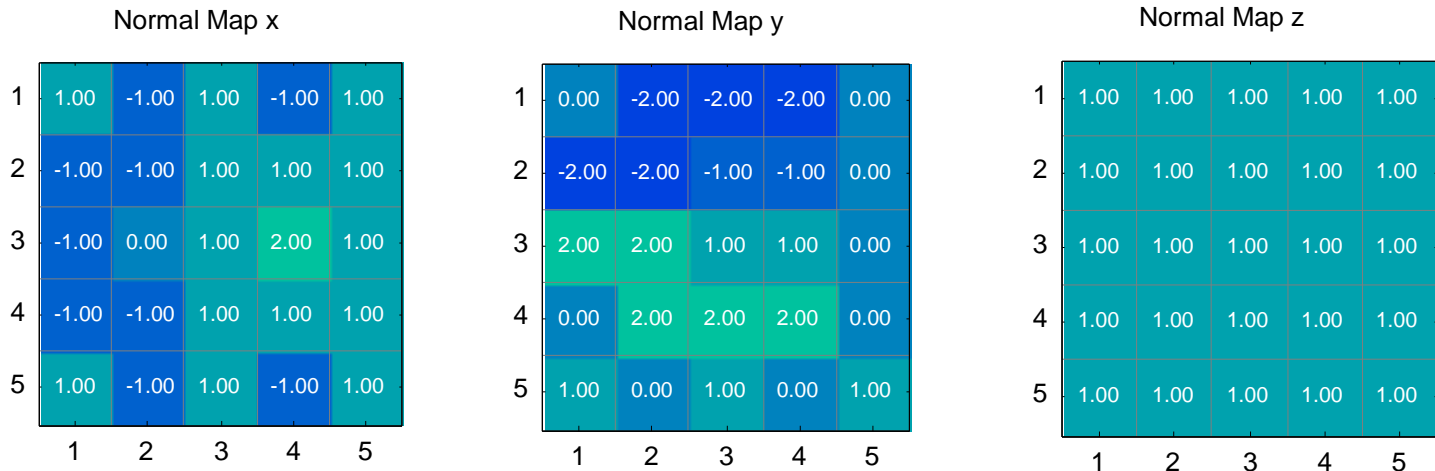
Lösung (2)

2) Generieren Sie aus der Textur eine Normal Map, die für das Bump Mapping eingesetzt werden kann.

Berechnung des Kreuzproduktes direkt aus der Textur:

$$n = \begin{pmatrix} 1 \\ 0 \\ g(s+1, t) - g(s, t) \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \\ g(s, t+1) - g(s, t) \end{pmatrix} = \begin{pmatrix} g(s, t) - g(s+1, t) \\ g(s, t) - g(s, t+1) \\ 1 \end{pmatrix}$$

→ Pixelweise Berechnung, z.B. Eintrag in Normal Map x bei 1,1: $g(s, t) - g(s+1, t) = g(1,1) - g(2,1) = 1 - 0 = 1$ wobei g der Originaltextur entspricht. Alternativ: Invertierung der bereits berechneten Ableitung bzw. Kreuzprodukt aus den berechneten Ableitungen



Lösung (3)

Normalisierung:

$$n = \frac{1}{\sqrt{(g(s,t) - g(s+1,t))^2 + (g(s,t) - g(s,t+1))^2 + 1}} \begin{pmatrix} g(s,t) - g(s+1,t) \\ g(s,t) - g(s,t+1) \\ 1 \end{pmatrix}$$

Beispiel für Normal Map x bei (1,1): $n_x = \frac{1}{\sqrt{1^2+0^2+1}} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = 0,71$

Normal Map x (normalisiert)

1	0.71	-0.41	0.41	-0.41	0.71
2	-0.41	-0.41	0.58	0.58	0.71
3	-0.41	0.00	0.58	0.82	0.71
4	-0.71	-0.41	0.41	0.41	0.71
5	0.58	-0.71	0.58	-0.71	0.58
	1	2	3	4	5

Normal Map y (normalisiert)

1	0.00	-0.82	-0.82	-0.82	0.00
2	-0.82	-0.82	-0.58	-0.58	0.00
3	0.82	0.89	0.58	0.41	0.00
4	0.00	0.82	0.82	0.82	0.00
5	0.58	0.00	0.58	0.00	0.58
	1	2	3	4	5

Normal Map z (normalisiert)

1	0.71	0.41	0.41	0.41	0.71
2	0.41	0.41	0.58	0.58	0.71
3	0.41	0.45	0.58	0.41	0.71
4	0.71	0.41	0.41	0.41	0.71
5	0.58	0.71	0.58	0.71	0.58
	1	2	3	4	5

Lösung (4)

Transformieren des Einheitsnormalenvektors in den Wertebereich $[0,1]$: $t = \frac{1}{2}(n + 1)$

Beispielhaft für den Eintrag in der Normal Map x bei $(1,1)$

$$t = \frac{1}{2}(0,71 + 1) = 0,85$$

Normal Map x (Wertebereich $[0,1]$)

1	0.85	0.30	0.70	0.30	0.85
2	0.30	0.30	0.79	0.79	0.85
3	0.30	0.50	0.79	0.91	0.85
4	0.15	0.30	0.70	0.70	0.85
5	0.79	0.15	0.79	0.15	0.79
	1	2	3	4	5

Normal Map y (Wertebereich $[0,1]$)

1	0.50	0.09	0.09	0.09	0.50
2	0.09	0.09	0.21	0.21	0.50
3	0.91	0.95	0.79	0.70	0.50
4	0.50	0.91	0.91	0.91	0.50
5	0.79	0.50	0.79	0.50	0.79
	1	2	3	4	5

Normal Map z (Wertebereich $[0,1]$)

1	0.85	0.70	0.70	0.70	0.85
2	0.70	0.70	0.79	0.79	0.85
3	0.70	0.72	0.79	0.70	0.85
4	0.85	0.70	0.70	0.70	0.85
5	0.79	0.85	0.79	0.85	0.79
	1	2	3	4	5

Übungsaufgabe

3. Gegeben sind die folgenden Werte für die Farben eines Pixels der Texturpixel g_t , des Texturhintergrundes g_h und aus der Beleuchtungsberechnung g_f . Geben Sie die resultierende Pixelfarbe g_r für die Modulationsfunktionen **GL_REPLACE**, **GL_MODULATE**, **GL_DECAL**, **GL_BLEND** und **GL_ADD** an:

$$g_t = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0,5 \end{pmatrix}, \quad g_f = \begin{pmatrix} 0,5 \\ 0,5 \\ 0,5 \\ 1 \end{pmatrix}, \quad g_h = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Gehen Sie davon aus, dass Farbwerte im Intervall $[0,1]$ liegen und gegebenenfalls abgeschnitten werden.

Lösung

3. Gegeben sind die folgenden Werte für die Farben eines Pixels der Texturpixel g_t , des Texturhintergrundes g_h und aus der Beleuchtungsberechnung g_f . Geben Sie die resultierende Pixelfarbe g_r für die Modulationsfunktionen `GL_REPLACE`, `GL_MODULATE`, `GL_DECAL`, `GL_BLEND` und `GL_ADD` an:

$$g_t = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0,5 \end{pmatrix}, \quad g_f = \begin{pmatrix} 0,5 \\ 0,5 \\ 0,5 \\ 1 \end{pmatrix}, \quad g_h = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Gehen Sie davon aus, dass Farbwerte im Intervall $[0, 1]$ liegen und gegebenenfalls abgeschnitten werden.

`GL_REPLACE`: Die Pixelfarbe entspricht der Texturfarbe.

$$g_r = g_t = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0,5 \end{pmatrix}$$

`GL_MODULATE`:

$$g_r = \begin{pmatrix} R_t \cdot R_f \\ G_t \cdot G_f \\ B_t \cdot B_f \\ A_t \cdot A_f \end{pmatrix} = \begin{pmatrix} 1 \cdot 0,5 \\ 0 \cdot 0,5 \\ 0 \cdot 0,5 \\ 0,5 \cdot 1 \end{pmatrix} = \begin{pmatrix} 0,5 \\ 0 \\ 0 \\ 0,5 \end{pmatrix}$$

Lösung (2)

GL_DECAL:

$$g_r = g_r = \begin{pmatrix} (1 - A_t)R_f + A_tR_t \\ (1 - A_t)G_f + A_tG_t \\ (1 - A_t)B_f + A_tB_t \\ A_f \end{pmatrix} = \begin{pmatrix} (1 - 0,5) \cdot 0,5 + 0,5 \cdot 1 \\ (1 - 0,5) \cdot 0,5 + 0,5 \cdot 0 \\ (1 - 0,5) \cdot 0,5 + 0,5 \cdot 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0,75 \\ 0,25 \\ 0,25 \\ 1 \end{pmatrix}$$

GL_BLEND:

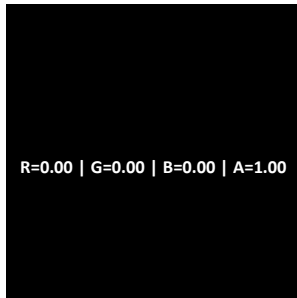
$$g_r = \begin{pmatrix} (1 - R_t)R_f + R_tR_h \\ (1 - G_t)G_f + G_tG_h \\ (1 - B_t)B_f + B_tB_h \\ A_t \cdot A_f \end{pmatrix} = \begin{pmatrix} (1 - 1)0,5 + 1 \cdot 1 \\ (1 - 0)0,5 + 0 \cdot 1 \\ (1 - 0)0,5 + 0 \cdot 1 \\ 0,5 \cdot 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0,5 \\ 0,5 \\ 0,5 \end{pmatrix}$$

GL_ADD: hier wird der Wertebereich begrenzt durch Abschneiden der Werte über 1,0. Entspricht auch der OpenGL-Konvention.

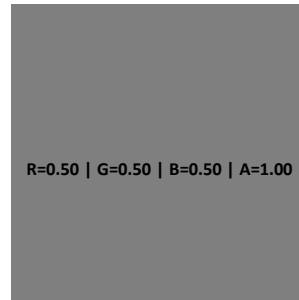
$$g_r = \begin{pmatrix} R_t + R_f \\ G_t + G_f \\ B_t + B_f \\ A_t \cdot A_f \end{pmatrix} = \begin{pmatrix} 1 + 0,5 \\ 0 + 0,5 \\ 0 + 0,5 \\ 0,5 \cdot 1 \end{pmatrix} = \begin{pmatrix} 1,5 \\ 0,5 \\ 0,5 \\ 0,5 \end{pmatrix} \Rightarrow \begin{pmatrix} 1 \\ 0,5 \\ 0,5 \\ 0,5 \end{pmatrix}$$

Lösung (3)

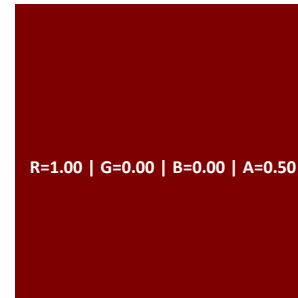
Background Color



Shading Color



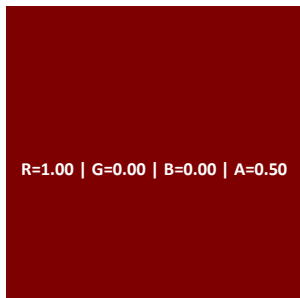
Texture Color



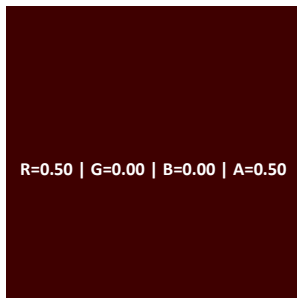
Texture Background Color

R=1.00 | G=1.00 | B=1.00 | A=1.00

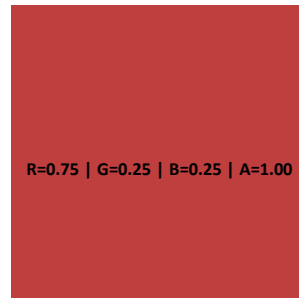
GL_REPLACE



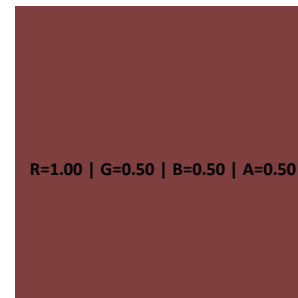
GL_MODULATE



GL_DECAL



GL_BLEND



GL_ADD (clamped)

