



Prof. Dr. Andreas Butz

Dipl.-Medieninf. Hendrik Richter

Dipl.-Medieninf. Raphael Wimmer

Computergrafik 1 Übung

Wiederholung Lineare Algebra:

Vektoren, Matrizen, Transformationen in 2D und 3D



Organisatorisches

Termin und Ort:

- 5 Wöchentliche Übungstermine:
 - Freitags, 12:15 – 13:45 und 14:15 – 15:45
 - Montags, 12:15 – 13:45 und 14:15 – 15:45 und 16:15 – 17:45
- Ort: Theresienstr. 39, Raum B 134
- Zum Ende des Semesters wird es eine Klausur geben (26.07.2010)



Ziele der Übung

Erlernen von grundlegenden Konzepten der 3D-Programmierung:

- Transformationen
- Licht und Texturen
- Interaktion

Erwerb von Kenntnissen in C/C++, OpenGL und QT:

- Wiederholung algebraischer Grundlagen
- Grundlage der C/C++ Programmierung
- Grundlegende 3D-Programmierung mit OpenGL
- Grundlagen der GUI-Programmierung mit QT

Verständnis der Mathematik für 3D-Grafik:

- Transformations- und Projektionsmatrizen



Formalia

Ablauf der Übung:

- Es wird wöchentliche Übungsblätter geben
 - Donnerstag 14:00 – Neues Aufgabenblatt.
 - Montag 12:00 – Abgabe der Aufgaben.
- Abgabe über UniWorX
- erlaubte Formate sind nur **Plain Text (UTF-8)** und **PDF**
- Die ersten Übungsblätter werden alleine bearbeitet und abgegeben
- im weiteren Verlauf der Übung werden von den Übungsleitern feste Gruppen zu 4 Personen eingeteilt
- **Alle** Aufgaben der späteren Übungsblätter müssen als **ein *.zip File** von **einem** Mitglied der Gruppe unter Angabe **aller anderen Mitglieder** erfolgen.



Bonuspunkte

- Abgaben müssen
 - Pünktlich (vor Montag, 12 Uhr **s.t.**)
 - Korrekt (kompilieren mit einem Klick)
 - Lauffähig
 - Und nicht abgeschrieben sein (für Abschreiber und Vorlagenlieferant keine Bonuspunkte)
- Jedes Blatt wird korrigiert, die relative Anzahl der erreichten Punkte wird umgerechnet auf 0 - 15 % der in der Klausur erreichbaren Punkte.
- **Nur** wenn **alle** Kriterien erfüllt sind, gehen die erworbenen Punkte als Bonuspunkte in die Klausur ein.
- Punkte sind Gruppenpunkte, d.h. jeder Teilnehmer einer Gruppe erhält die gleiche Anzahl Punkte.



Wiederholung Lineare Algebra: Vektoren



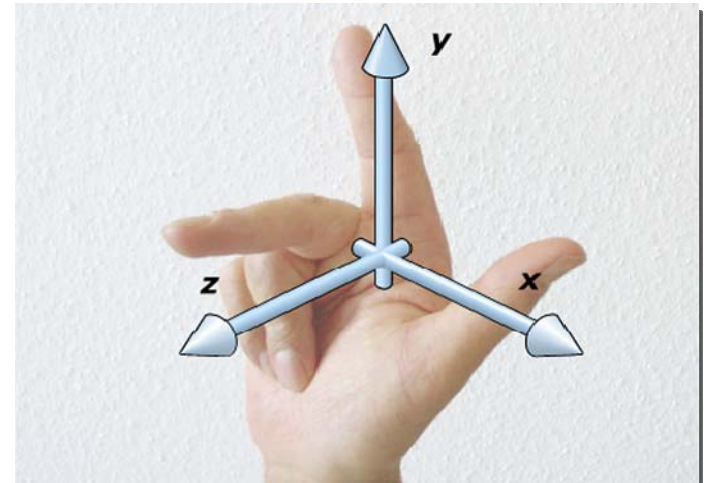
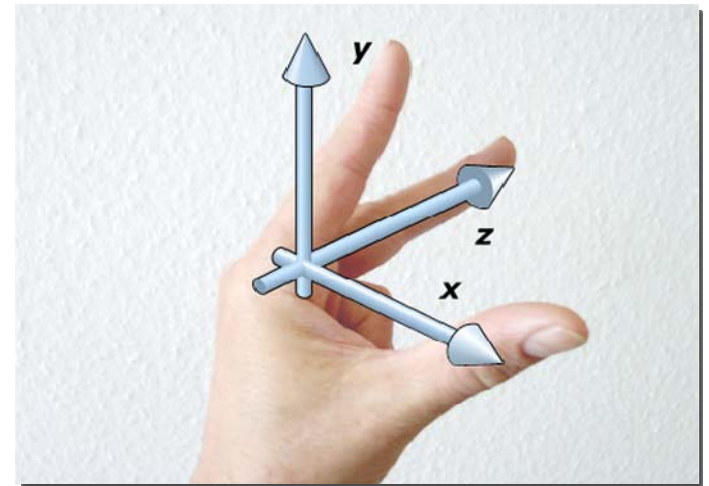
Koordinatensysteme

X – Richtung des Daumens

Y – Zeigefinger

Z – Mittelfinger

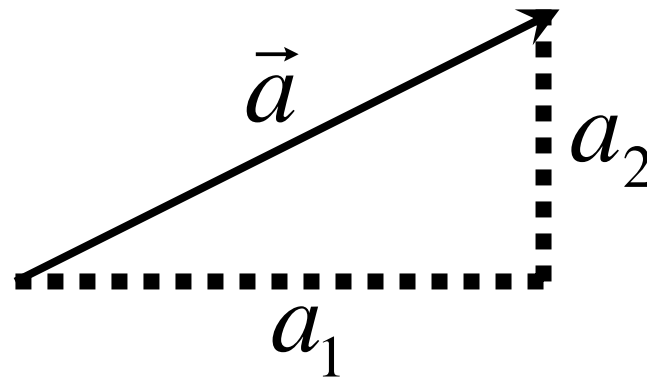
Die beiden Koordinatensysteme sind
spiegelbildlich
und nicht durch Drehung ineinander
zu überführen.



Quelle: Skript zur Vorlesung Computergrafik 1
im SS09 – Prof. Dr. Ing. Axel Hoppe – LMU
München



Vektoren - Grundbegriffe



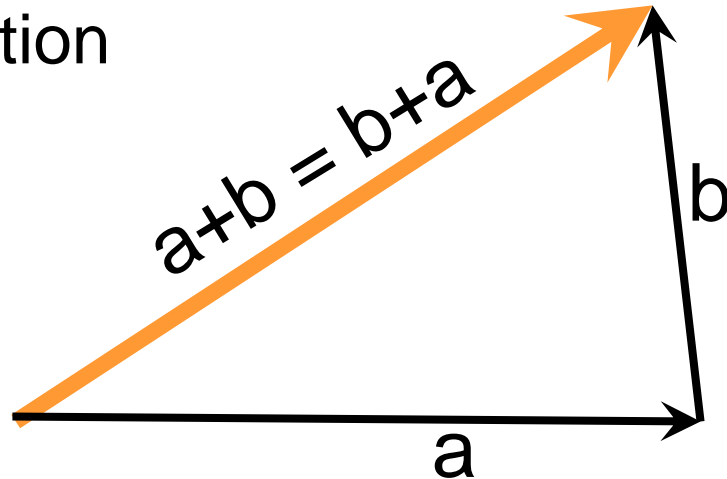
- Elemente eines Vektorraums
- Länge und Richtung = Verschiebung
- keine Aussage über absolute Position

- Schreibweise: $\vec{a} := \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}$

- Betrag/ Norm/ Länge eines Vektors: $\|\vec{a}\| = \sqrt{\vec{a} \cdot \vec{a}} = \sqrt{a_1^2 + a_2^2 + \dots + a_n^2}$



Vektoren - Addition



- geometrisch: Anfang des zweiten Pfeils an Spitze des ersten Pfeils
- Addition ist kommutativ
- im kartesischen Koordinatensystem:

$$\vec{a} + \vec{b} = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} + \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} x_1 + x_2 \\ y_1 + y_2 \end{pmatrix}$$



Vektoren – Multiplikation 1: Skalarmultiplikation

Multiplikation mit einem Skalar/ Skalarmultiplikation:

$$\vec{a} * \alpha = \begin{pmatrix} x \\ y \\ z \end{pmatrix} * \alpha = \begin{pmatrix} x * \alpha \\ y * \alpha \\ z * \alpha \end{pmatrix}$$

- **Veränderung der Vektorlänge um Faktor des Skalars**
- **das Ergebnis ist wieder ein Vektor**
- **bei Multiplikation mit negativem Skalar dreht sich die Richtung des Vektors**



Vektoren – Multiplikation 2: Skalarprodukt

Multiplikation eines Vektors mit einem Vektor:

$$\langle a, b \rangle \Leftrightarrow \vec{a} \cdot \vec{b} = \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} \cdot \begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix} = x_1x_2 + y_1y_2 + z_1z_2$$

- **das Ergebnis ist ein Skalar**
- **wenn Skalarprodukt = 0, stehen die Vektoren senkrecht aufeinander bzw sind orthogonal**
- **Verwendung: Winkel zwischen Vektoren ausrechnen**

$$\cos \lambda = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \cdot \|\vec{b}\|}$$

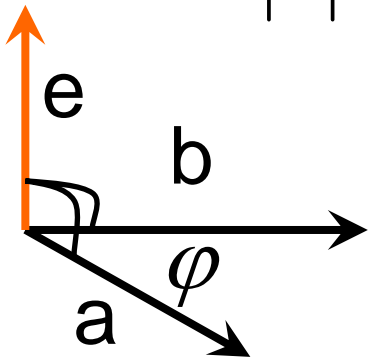


Vektoren – Multiplikation 3: Kreuzprodukt/ Vektorprodukt

**Ziel: Zwei Vektoren zu einem Normalenvektor der Ebene verbinden.
Welcher Vektor steht auf beiden Vektoren senkrecht?**

**Die Länge dieses Vektors ist die Flächengröße des Parallelogramms
mit den Seiten \vec{a} und \vec{b} .**

$$\vec{a} \times \vec{b} = |\vec{a}| \cdot |\vec{b}| \cdot \sin(\varphi) \cdot \vec{e}$$



Phi = Winkel, eingeschlossen von a und b

e = Einheitsvektor, zu beiden Vektoren orthogonal

$$\vec{a} \times \vec{b} = \begin{pmatrix} a_2 \cdot b_3 - a_3 \cdot b_2 \\ a_3 \cdot b_1 - a_1 \cdot b_3 \\ a_1 \cdot b_2 - a_2 \cdot b_1 \end{pmatrix}$$



Vektoren – Lineare Abhängigkeit

Mehrere Vektoren sind linear abhängig, wenn

- sie parallel (2D) oder koplanar (3D) zueinander sind ODER
- man einen aus den anderen erstellen kann ODER
- man mit ihnen den Nullvektor ausdrücken kann, d.h.
- es $\lambda_1, \dots, \lambda_n$ gibt, nicht alle Null, so daß gilt:

$$\lambda_1 \vec{v}_1 + \dots + \lambda_n \vec{v}_n = \mathbf{0}$$



Geraden im \mathbb{R}^3

Geraden im \mathbb{R}^3 haben folgende Form:

$$x = \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \lambda \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

Punkt Geraden-
parameter Richtungs-
vektor

Zur Berechnung eines Geradenschnittpunkts:

- Beide Geradengleichungen gleichsetzen
- eine Gleichung für jede Koordinate
- Geradenparameter berechnen
- Einsetzen



http://commons.wikimedia.org/wiki/File:Casio_fx-991ES_Calculator_New.jpg

Stefan-Xp

GNU FDL



Wiederholung Lineare Algebra: Matrizen

<http://matrixcookbook.com/>



Matrizen - Grundbegriffe

- Anordnung von Elementen (m x n = m Zeilen, n Spalten)
- Verwendung zur Transformation von Punkten und Vektoren
 - Translation, Rotation, Scheren, Skalieren
- Addition und Multiplikation mit einem Skalar: Element für Element

$$\begin{pmatrix} 1 & 3 \\ 5 & 2 \\ 0 & 4 \end{pmatrix}$$



Matrizen – Multiplikation mit einer Matrix

- **Spalten in Matrix 1 = Zeilen in Matrix 2**
- **Element (i,j) im Produkt ist Produkt der Zeile i der Matrix 1 und Spalte j der Matrix 2**

$$\begin{pmatrix} 1 & 3 \\ 5 & 2 \\ 0 & 4 \end{pmatrix} \begin{pmatrix} 3 & 6 & 9 & 4 \\ 2 & 7 & 8 & 3 \end{pmatrix} = \begin{pmatrix} 9 & 27 & 33 & 13 \\ 19 & 44 & 61 & 26 \\ 8 & 28 & 32 & 12 \end{pmatrix}$$

- **nicht kommutativ: $AB \neq BA$**
- **assoziativ: $A*(B*C) = (A*B)*C$**
- **distributiv: $A*(B+C) = AB + AC$**



Matrizen – Multiplikation mit einer Matrix (Falkesches Schema)

- **Spalten in Matrix 1 = Zeilen in Matrix 2**
- **Element (i,j) im Produkt ist Produkt der Zeile i der Matrix 1 und Spalte j der Matrix 2**

						Spalte j
			1	2	3	4
			3	6	9	4
Zeile i			2	7	8	3
1	1	3	9	27	33	13
2	5	2	19	44	61	26
3	0	4	8	28	32	12

$$\begin{pmatrix} 1 & 3 \\ 5 & 2 \\ 0 & 4 \end{pmatrix} \begin{pmatrix} 3 & 6 & 9 & 4 \\ 2 & 7 & 8 & 3 \end{pmatrix}$$



Matrizen – Multiplikation mit einem Vektor

- **Vektor als einspaltige Matrix (m x 1)**

$$\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} -x \\ y \end{pmatrix}$$



Matrizen – Transponieren einer Matrix

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix}^T = \begin{pmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{pmatrix}$$



Matrizen – Einheitsmatrix

$$I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Zwei Matrizen, deren Produkt bei der Matrizenmultiplikation die Einheitsmatrix ist, sind zueinander invers.

$$AA^{-1} = A^{-1}A = I$$

$$(AB)^{-1} = B^{-1}A^{-1}$$



Matrizen – Inverse Matrix

Zwei Matrizen, deren Produkt bei der Matrizenmultiplikation die Einheitsmatrix ist, sind zueinander invers.

Elementare Zeilenumformungen

1. Vertauschung zweier Zeilen
2. Multiplikation einer Zeile mit einer konstanten Zahl $\neq 0$
3. Addition des beliebigen Vielfachen einer Zeile zu einer anderen Zeile

$$\begin{pmatrix} 0 & -1 & 0 \\ 1 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

1. Matrix zur Einheitsmatrix umformen
2. gleiche Schritte auf Einheitsmatrix anwenden
3. umgeformte Einheitsmatrix ist Inverse der Quellmatrix



http://commons.wikimedia.org/wiki/File:Casio_fx-991ES_Calculator_New.jpg

Stefan-Xp

GNU FDL



Wiederholung Lineare Algebra: Transformationen



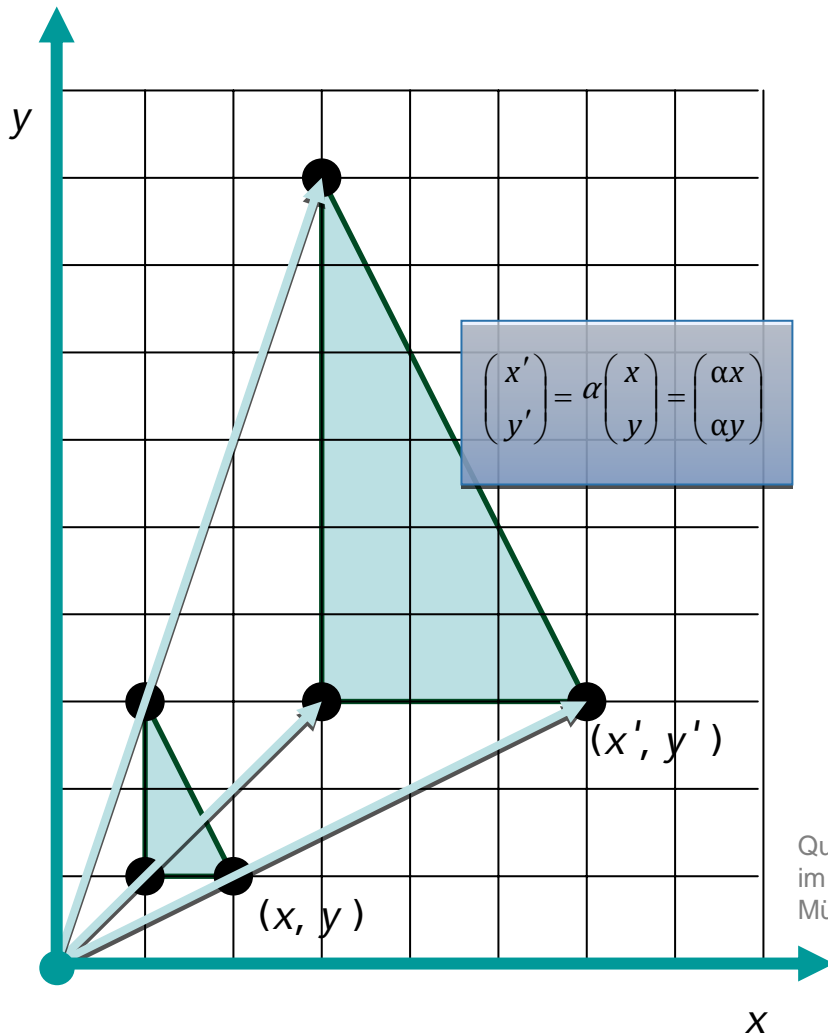
Transformationen in 2D - Grundbegriffe

Bewegungen = Transformationen

- Veränderung der Position von Punkten
- Verschiebung = Translation
- Größenveränderungen = Skalierung
- Drehung = Rotation
- Weitere affine (linienerhaltende) Transformationen:
 - Spiegelung
 - Scherung



Uniforme Skalierung

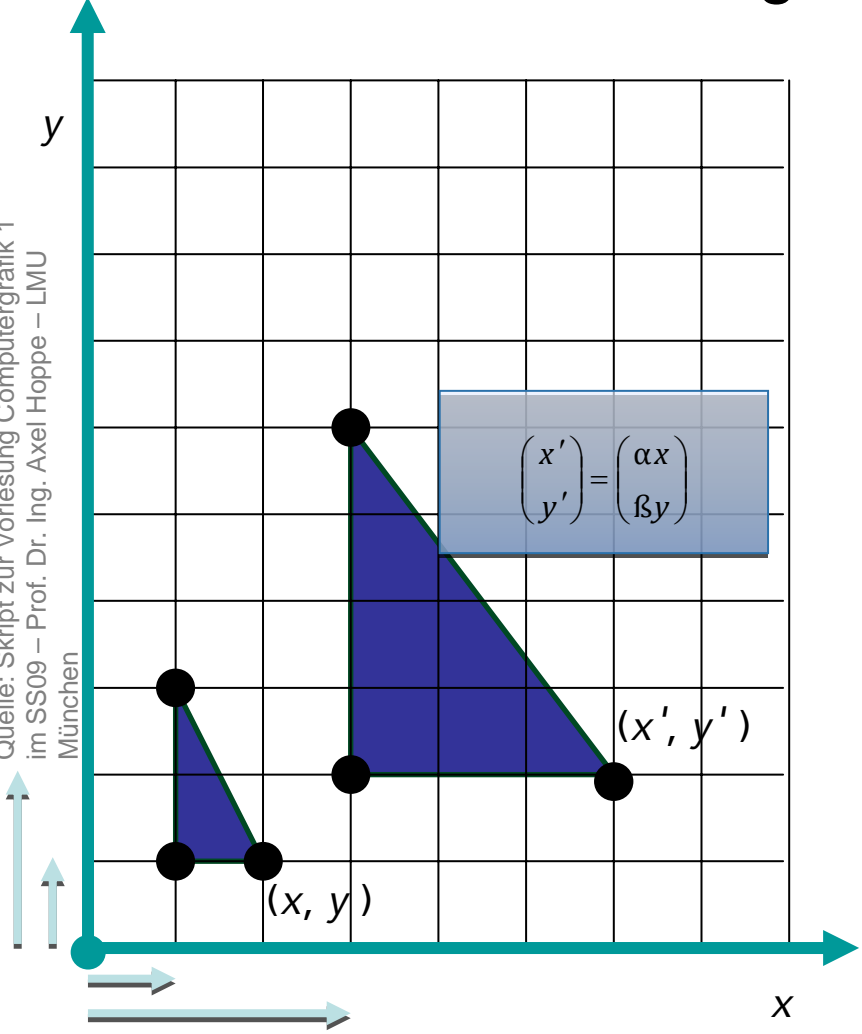


- Zentrum der Skalierung ist 0, Skalierung erfolgt in alle Richtungen uniform mit dem skalaren Faktor α
- Multiplikation mit dem Skalierungsfaktor
- Ortsvektor zu (x, y) wird auf das α -fache verlängert, um (x', y') zu erhalten

Quelle: Skript zur Vorlesung Computergrafik 1
im SS09 – Prof. Dr. Ing. Axel Hoppe – LMU
München



Nicht-uniforme Skalierungen

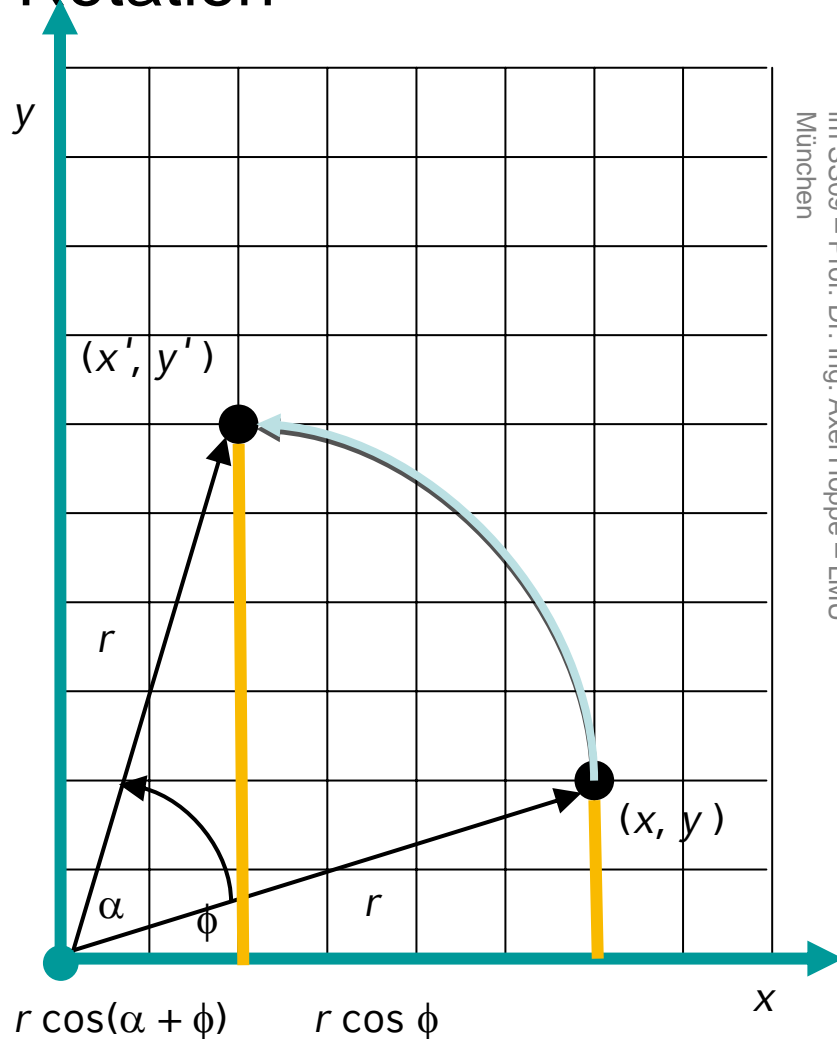


- Zentrum der Skalierung ist 0, Skalierung erfolgt in x-Richtung mit dem Faktor α , in y-Richtung mit β (Skalierungsvektor $(\alpha, \beta)^T$)
- Multiplikation mit entsprechenden Skalierungsfaktoren
- Ortsvektor zu (x, y) wird auf das α -fache in x-Richtung und das β -fache in y-Richtung verlängert.
- eine Darstellung als Matrizenmultiplikation ist möglich:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \alpha x \\ \beta y \end{pmatrix}$$



Rotation



- Rotationszentrum ist der Nullpunkt.
- Positive Werte von α ergeben eine Drehung entgegen dem Uhrzeigersinn.
- Punkt (x, y) wird um den Winkel α um den Nullpunkt gedreht, so dass sich der Punkt (x', y') ergibt.



Rotation: Berechnungs-Vorschrift

Rotationen um negative Winkel erfolgen mit dem Uhrzeigersinn; ausnutzen:

$$\cos(-\alpha) = \cos(\alpha) \text{ und} \\ \sin(-\alpha) = -\sin(\alpha)$$

die Berechnungs-Vorschrift

$$x' = x \cos \alpha - y \sin \alpha$$

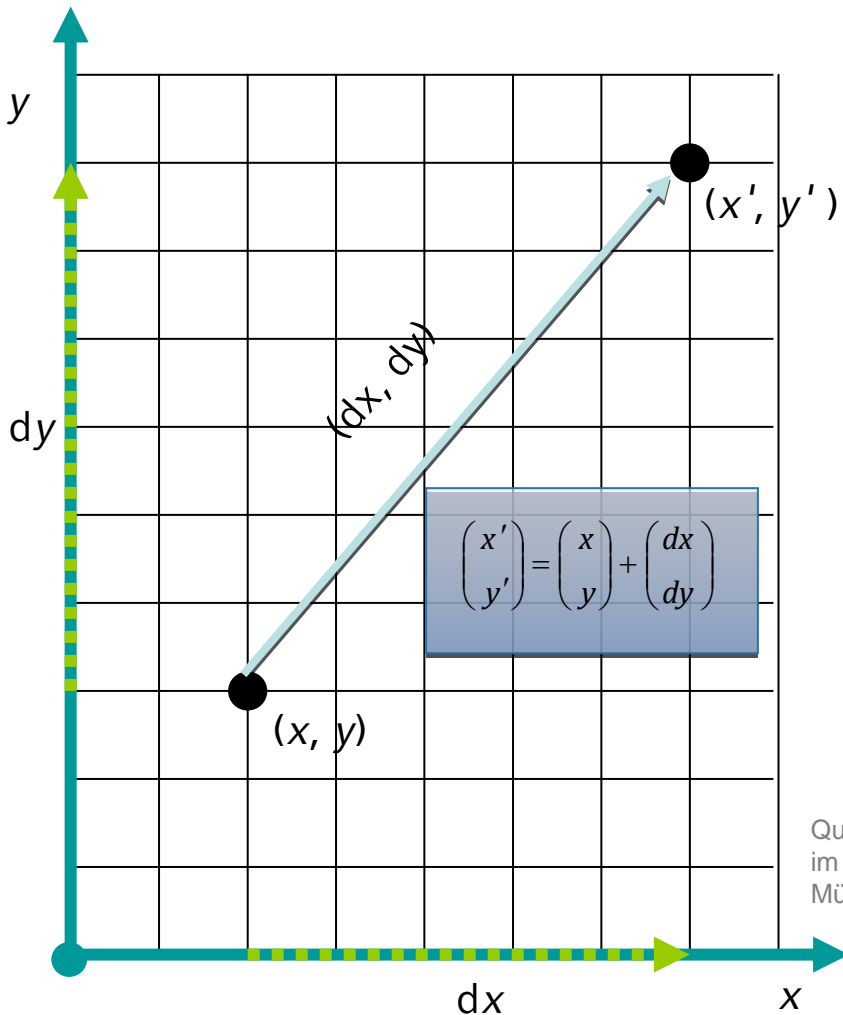
$$y' = x \sin \alpha + y \cos \alpha$$

kann als Matrix-Vektormultiplikation ausgedrückt werden:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$



Translation



Quelle: Skript zur Vorlesung Computergrafik 1
im SS09 – Prof. Dr. Ing. Axel Hoppe – LMU
München

- Punkt (x, y) wird auf gerade Linie nach (x', y') verschoben.
- Addition des Verschiebungsvektors
- Beschreibung der Translation durch einen Vektor (dx, dy) , der die Verschiebungsweite in x - und y -Richtung angibt



Zusammenfassung

Skalierung = **Multiplikation** mit Vektor/ Skalierungsmatrix

Rotation = **Multiplikation** mit orthonormaler Rotationsmatrix

Translation = **Addition** eines Translationsvektors

Matrizenmultiplikation ist assoziativ*, lange Ketten von Transformationsmatrizen könnten also zusammengefasst werden. Wie lässt sich die Translation als Matrizenmultiplikation ausdrücken?

→ Homogene Koordinaten

→ Translation als Matrixmultiplikation:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & \alpha \\ 0 & 1 & \beta \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} \alpha + x \\ \beta + y \\ 1 \end{pmatrix}$$

VORTEIL:

Repräsentation aller Punkte in homogenen Koordinaten ermöglicht einheitliche Behandlung der Transformationen, Geschwindigkeitsgewinn

* Matrizen A, B und Vektor x : $A*(B*x) = (A*B)*x$



Transformationen in 3D

- Vorgehensweise analog zu 2D
- Arbeit mit homogenen Koordinaten
- homogene Koordinaten sind jetzt vierdimensional
- Transformationsmatrizen demzufolge 4×4 -Matrizen
- Anwendung wie in 2D

$$T = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Translation

$$S = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Skalierung

$$R_x = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$R_y = \begin{pmatrix} \cos \alpha & 0 & \sin \alpha & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \alpha & 0 & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$R_z = \begin{pmatrix} \cos \alpha & -\sin \alpha & 0 & 0 \\ \sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Rotation



http://commons.wikimedia.org/wiki/File:Casio_fx-991ES_Calculator_New.jpg

Stefan-Xp

GNU FDL



Vielen Dank!

Nächstes Mal:

C++, Qt