

Selected Topics Augmented and Virtual Reality

Einführung VR Umgebung - 2D Bereiche

Prof. Dr. Gerald Pirkel

<g.pirkel@oth-aw.de>

Amberg, 26. Oktober 2023

- Bitte beachten Sie das *Urheberrecht!*
- *Alle Materialien* dieser Vorlesung sind – auch wenn sie nicht ausdrücklich gekennzeichnet sind – *urheberrechtlich geschützt*.
- Sie dienen *ausschließlich* Ihrem *persönlichen Gebrauch* im Rahmen dieser Vorlesung.
- Die Materialien dürfen insbesondere *nicht weiter verbreitet* werden.
- *Eigene Aufzeichnungen* (Video, Foto, Ton) der Vorlesung sind *nicht gestattet*.

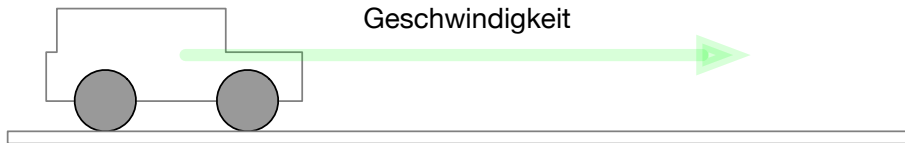
- Grundlagen Bewegungsmodelle und deren Benutzerkopplung
- Signale
- Kollider, Rigidbody
- Scripting
- Timer
- Erste Schritte in Godot
- 2D Umgebungen in Godot
- Knoten

[Godot 2D Tutorial](#) [Art Work - Tutorial Godot](#)

Knoten und Szenen: Organisationselement, um Inhalte und Logik zu kombinieren und zu verknüpfen.

- Animation, Bilder, Darstellungen, Modelle (2D und 3D)
- Audiokomponenten
- Beleuchtung und Kameras, visuelle Anzeigen
- Hierarchien (Kindknoten → lokales Koordinatensystem / relative Koordinaten)

Das an das Objekt geknüpfte Skript enthält das Bewegungsmodell.



Modell im 2D: $\mathbf{p} = (x, y)$ und $\mathbf{v} = (v_x, v_y)$

- Messung der Zeit zwischen 2 Updates (bei 30 FPS: $dt = 1s/30$) oder über Realzeitmessung (besser!)
- Bewegung seit vorherigem Update: $\mathbf{o}_c = \mathbf{v}/3600 * dt$
- neue Position $\mathbf{p}_c = \mathbf{p}_p + \mathbf{o}_c$

Verwendet: Vektoraddition und Skalierung

Zugrundeliegend: Bewegungsmodell (Position + Bewegungsrichtung + Geschwindigkeit, Beschleunigung, Reibung,....)

Benutzereingaben können verschiedene Parameter eines Spielelements beeinflussen:

- Koordinaten (direkt), $(\delta x, \delta y)$
- Geschwindigkeit $(x = x_{alt} + v_x, y = y_{alt} + v_y)$

Benutzereingaben bis jetzt: Tastatur und Mauseingaben

Eventbasierte Eingaben (*pressed, release, while_pressed*)

Godot: `Input.is_action_pressed(WERT)`

A full-page view of a blank sheet of graph paper. The grid consists of thin, light gray horizontal and vertical lines forming small squares across the entire page. There are no margins, text, or other markings on the paper.

- Schwerkraft
- Reibung
- Elastische Stöße
- Impulserhaltung
- Bewegungsmodelle (Beschleunigung u.Ä.)
- Flugbahnen
- (gedämpfte) Schwingungen
- Magnetismus
- Optik (Laser u.Ä.)
- Hebel
- Particle System (Fluide und Gase o.Ä.)

Meistens: Berechnung von Kräften, die auf ein Objekt wirken.

Differential Gleichungen

Knoten, der Physik implementiert / simuliert. Interaktion mit Umgebung und Spielelementen

- Physikalische Simulation (Bewegung, Rotation, Kollisionen, Externe Kräfte)
- 2D Umgebung / 3D Umgebung
- **Eigenschaften:** Masse, Reibung, Flexibilität (Restitution), linear / winkelabhängige Dämpfung, Gravitation
- Erkennung von Kollisionen (Hüllkörper und Signale)
- Simulation Einwirkung von Kräften (Gravitation, Reibung, Luft), Einfluss auf Geschwindigkeit, Impuls und Kontinuierliche Einwirkung

Alternative: staticbodies

Ausgangslage: Position des Körpers (x,y,z) , z ist der Abstand zwischen Boden und Körper
Gravitation wirkt nun auf den Körper ($1g = 9.81 \frac{m}{s^2}$)

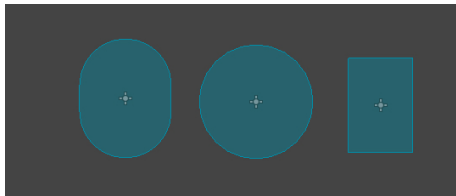
$$z(t) = -\frac{1}{2} * g * t^2 + z_0$$

Achtung: die Bewegung ist zeitabhängig und wirkt nur auf die z Achse
Was passiert, wenn Linear Geschwindigkeit auf Element wirkt?

- Collider

acing)

Hüllkörper - einfache geometrische Formen (Rechteck, Kreis, Kapsel), `CollisionObject`

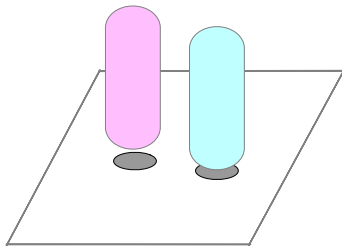


`inTest` löst ein Signal aus (`body_enter` o.Ä.), Verbindung f. Reaktion.

Kollisionsebenen (Layers)

Vorhersagemodell für Optimierung der Kollisionserkennung (Reduktion!)

Ziel: Erkennung wann sich 2 Element berühren oder schneiden



- Gravitation
- Gegenstand steht auf Boden
- Rennwagen berührt Leitplanke
- Interaktion mit Gegenstand (Umwerfen) oder Aktion auslösen
- Darstellungsoptimierung (Vermeidung von Clipping)

Komplexe 3D Modelle mit Polygonausläufern



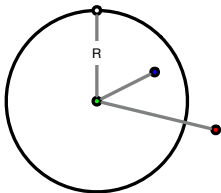
Vereinfachung durch einfache geometrische Figuren (sog. Hüllkörper)



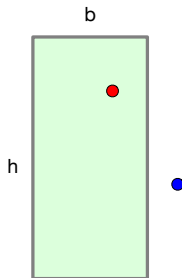


Kapsel, Kubus, Kugel

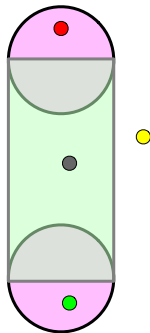
Kugel



Kubus



Kapsel



inTest bzgl dieser Figuren ist sehr einfach.

Liegt ein Punkt in der geometrischen Figure?

Gibt es einen Schnittpunkt zwischen einem Strahl und der geometrischen Figur?

- Hüllkörper müssen an eigentliches Objekt gebunden werden (Bewegung!)
- Hüllkörper haben lokales Koordinatensystem mit entsprechender Ausrichtung - dies muss bei Kollision berücksichtigt werden

- Bewegung + Steuerung des Spielers
- Animation, (Sound)
- Gegner Bewegung, Spawn neuer Elemente
- Kollisionserkennung