

Smartphonespiel automatisiert spielen

Lukas Lihotzki

20.06.2016

Gliederung

Das Spiel "Flappy Bird"

Aufbau von Hardware und Software

Kameraauswertung

Modell des Spiels

Algorithmus zur Spielersteuerung

Erzeugen der Touchereignisse

Bekannte Schwierigkeiten

Vorführung

Gliederung

Das Spiel "Flappy Bird"

Aufbau von Hardware und Software

Kameraauswertung

Modell des Spiels

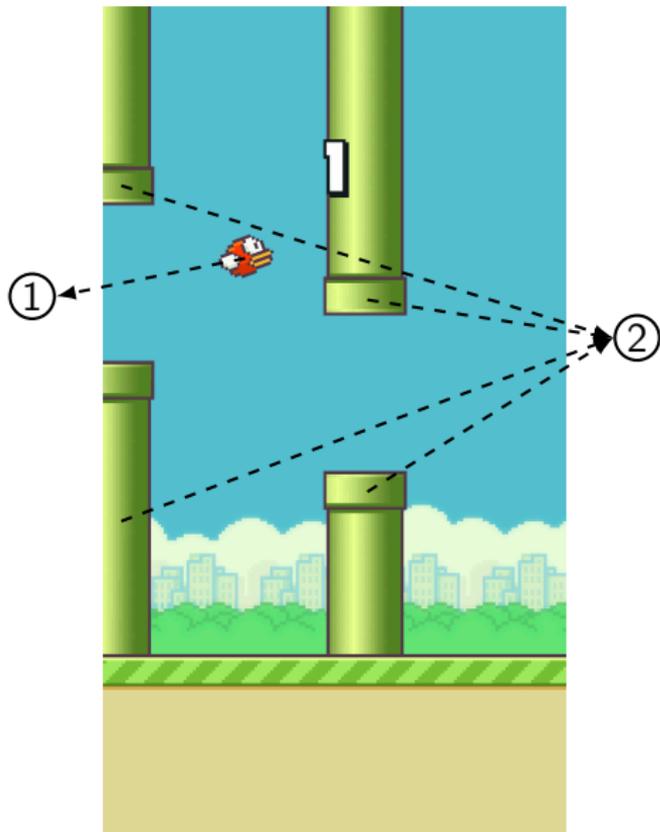
Algorithmus zur Spielersteuerung

Erzeugen der Touchereignisse

Bekannte Schwierigkeiten

Vorführung

Flappy Bird



1. Spieler (Vogel)

- ▶ Bewegt sich hoch oder herunter
- ▶ Unterliegt der Schwerkraft (fällt)
- ▶ Anheben durch Touch

2. Hinderniss (Röhre)

- ▶ Bewegen sich von rechts nach links
- ▶ Ziel: Zwischen ihnen durchfliegen

Flappy Bird Video

Im Hinblick auf Automation

- ▶ Touchposition ist irrelevant.
- ▶ Eine kleine Zeitspanne vergeht ab dem Spielstart bis zum ersten Hinderniss.
- ▶ Bei jedem Spielstart wird zufällig ausgewählt:
 - ▶ Einer aus zwei Hintergründen (Tag, Nacht)
 - ▶ Eine aus drei Vogelfarben (Rot, Orange, Blau)



Gliederung

Das Spiel “Flappy Bird”

Aufbau von Hardware und Software

Kameraauswertung

Modell des Spiels

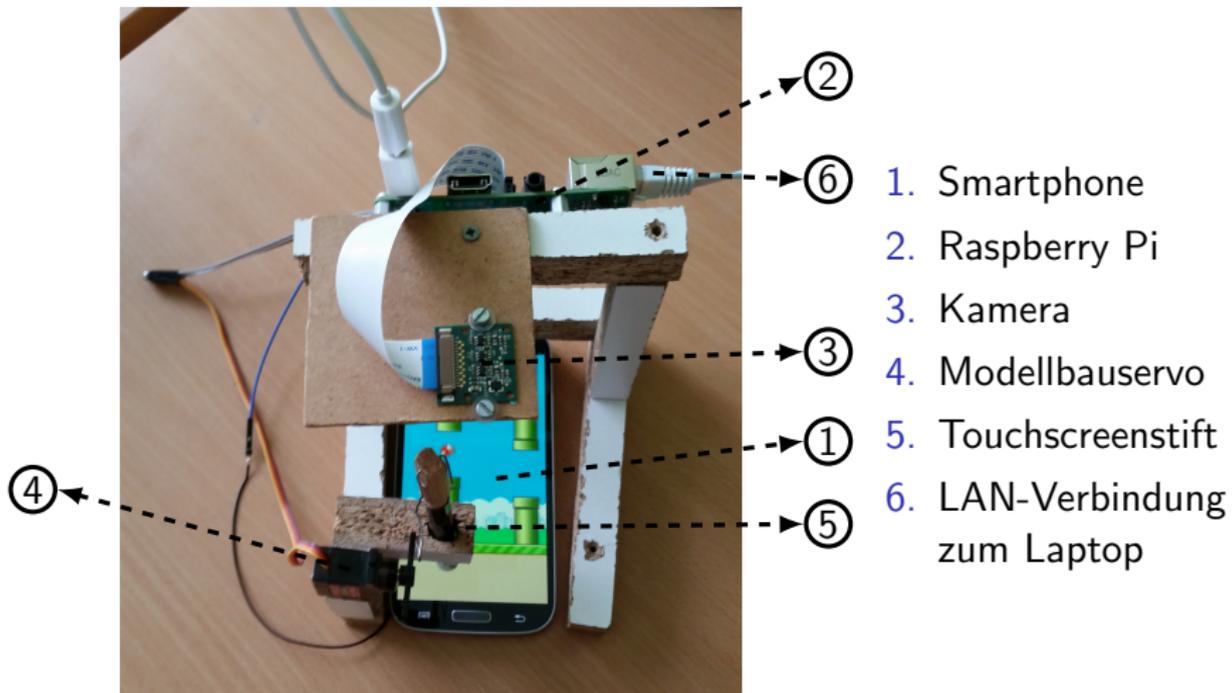
Algorithmus zur Spielersteuerung

Erzeugen der Touchereignisse

Bekannte Schwierigkeiten

Vorführung

Hardware



Software

- ▶ C++-Programm auf dem Laptop
 - ▶ Benutzeroberfläche (Qt)
- ▶ C++-Programm auf dem Raspberry Pi
 - ▶ Kamerabilderfassung
 - ▶ Steuerung des Servomotors
- ▶ Kommunikation über ein TCP-Binärprotokoll

Gliederung

Das Spiel “Flappy Bird”

Aufbau von Hardware und Software

Kameraauswertung

Modell des Spiels

Algorithmus zur Spielersteuerung

Erzeugen der Touchereignisse

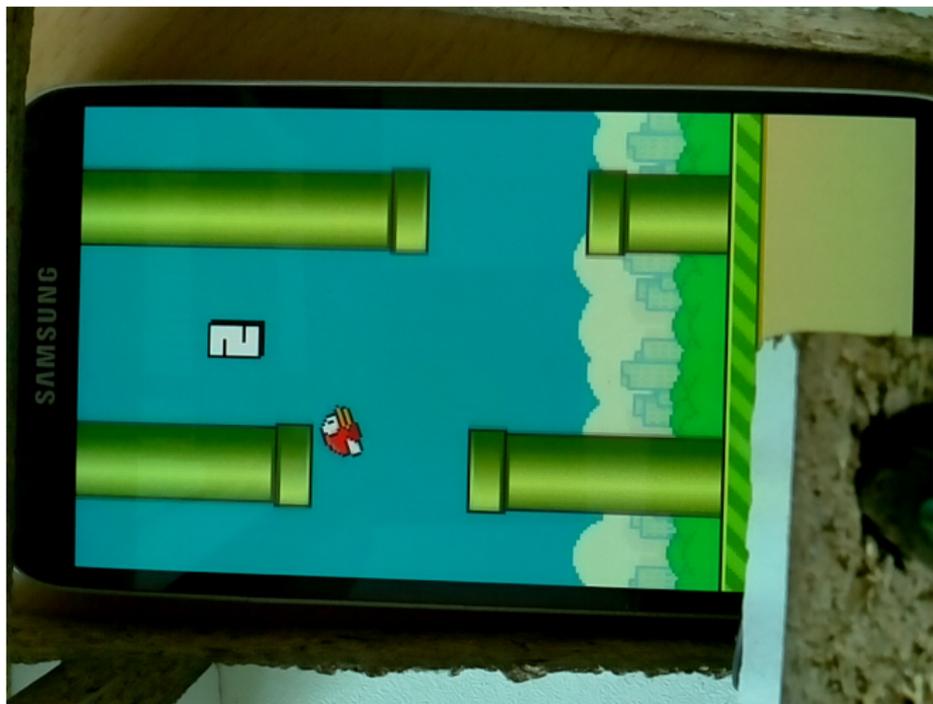
Bekannte Schwierigkeiten

Vorführung

Kameraschnittstelle

- ▶ Video4Linux2-API
 - ▶ Herstellerübergreifende API
 - ▶ Ermöglicht das Setzen von Kameraparametern
 - ▶ Liefert Speicherbereich mit Bilddaten
- ▶ Raspberry Pi-Kamera
 - ▶ Video4Linux2-kompatibel
 - ▶ Unkomprimierte Bildübertragung über das schnellere CSI statt USB
 - ▶ Ein Grund für das System mit Raspberry Pi

Kamerabild



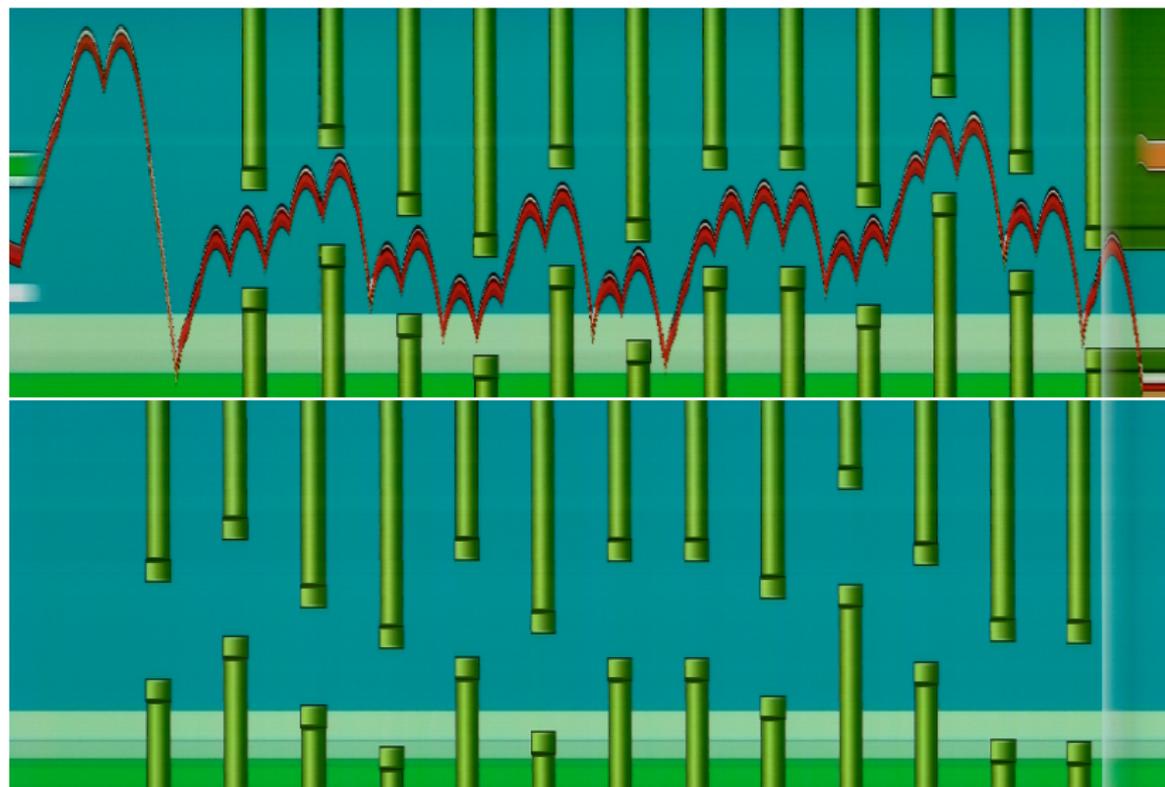
Interessante Linien



Linienextraktion

- ▶ Bresenham-Algorithmus
- ▶ Modifiziert, um die Pixel entlang einer Linie abzutragen anstatt eine Linie zu zeichnen

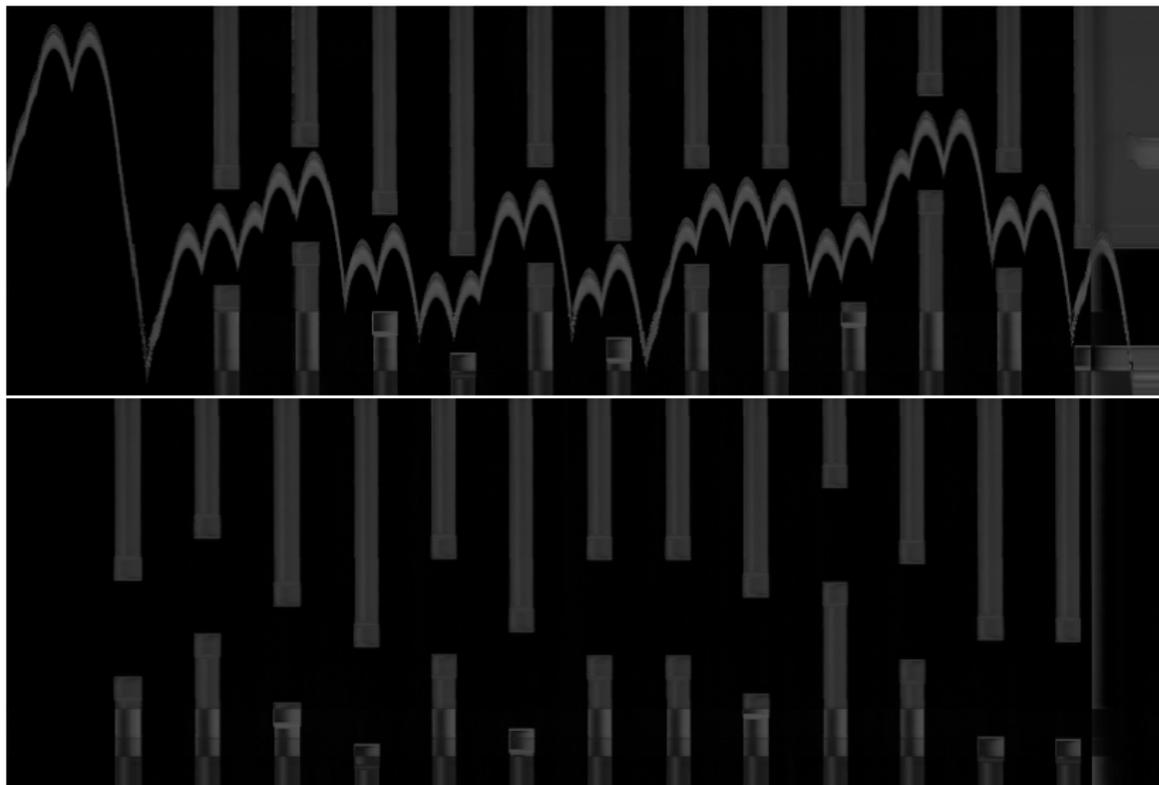
Interessante Linien über die Zeit



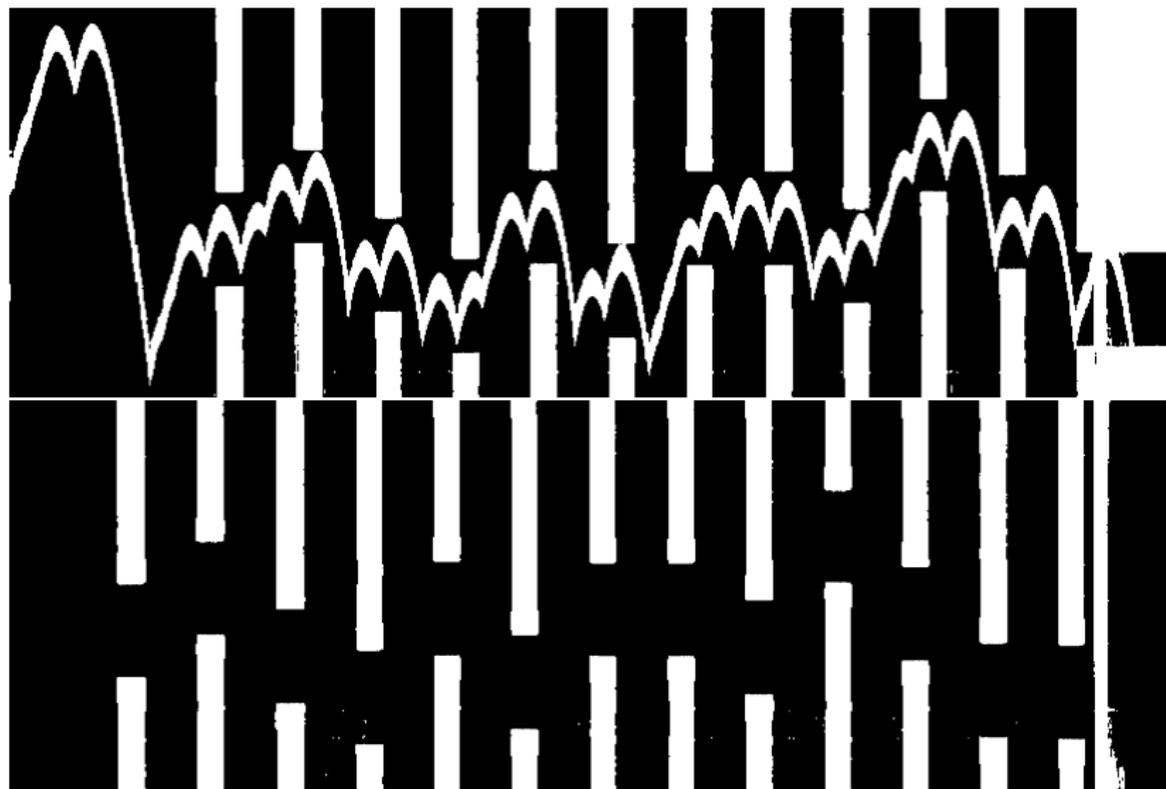
Medianwert



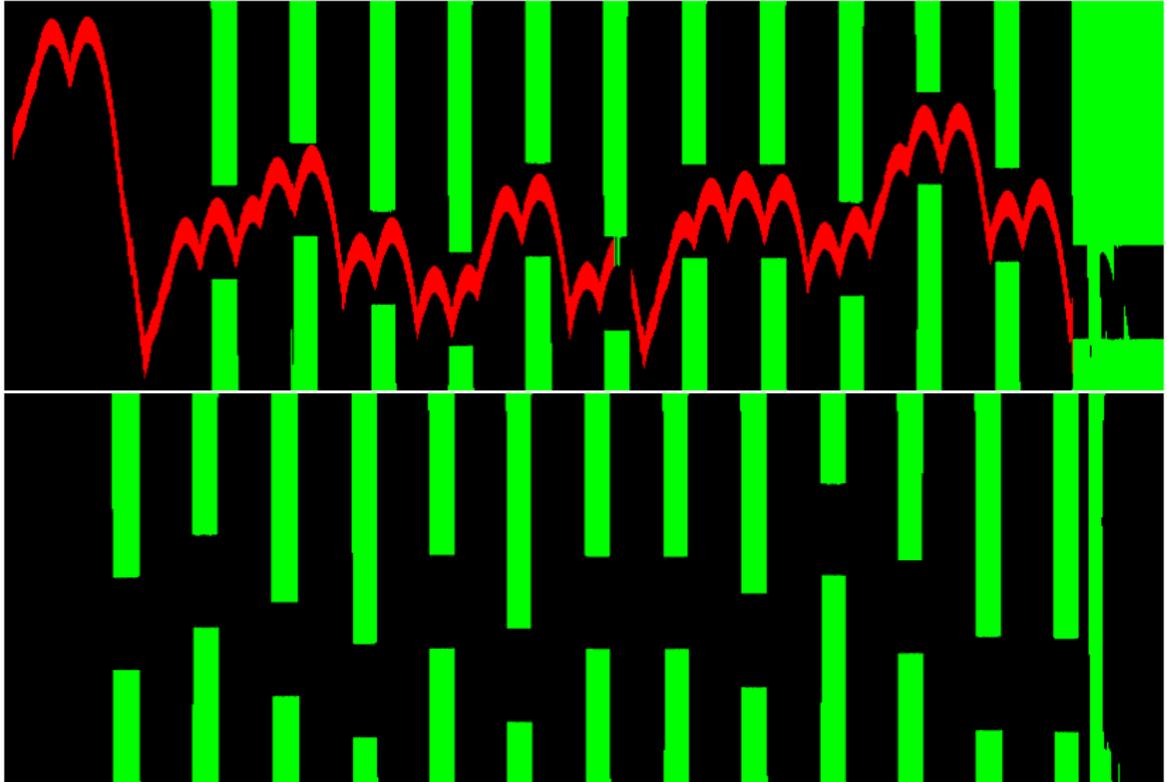
Distanz zum Medianwert



Schwellwert



Abschnittsfilter



Gliederung

Das Spiel “Flappy Bird”

Aufbau von Hardware und Software

Kameraauswertung

Modell des Spiels

Algorithmus zur Spielersteuerung

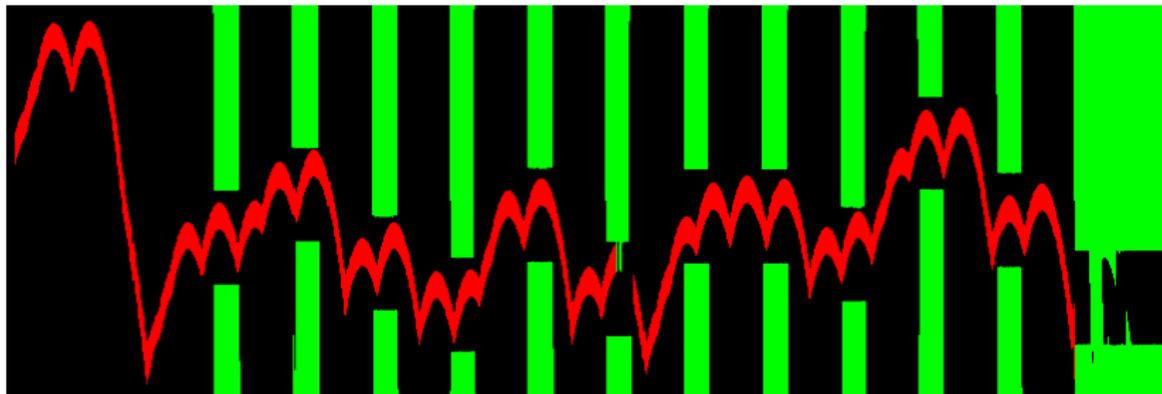
Erzeugen der Touchereignisse

Bekannte Schwierigkeiten

Vorführung

Modell des Spiels

- ▶ Vogel
 - ▶ Immer gleiche Flugparabel (nur andere Verschiebung)
- ▶ Rohr
 - ▶ Aufeinanderfolgende, gleiche Linien
- ▶ Latenz
 - ▶ Zeitspanne von dem Auslösen des Touches durch die Steuerung bis die Auswertung den Flugparabelbeginn registriert
 - ▶ etwa 0,2 Sekunden



Gliederung

Das Spiel "Flappy Bird"

Aufbau von Hardware und Software

Kameraauswertung

Modell des Spiels

Algorithmus zur Spielersteuerung

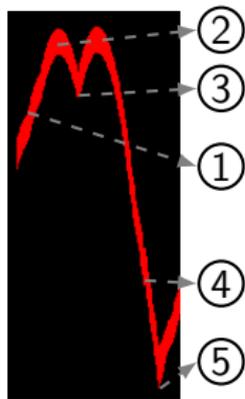
Erzeugen der Touchereignisse

Bekannte Schwierigkeiten

Vorführung

Einlernphase (Ermittlung der Flugparabel)

1. Schnell hintereinander touchen, um den Spieler weit nach oben zu bringen
 2. Warten auf die Sinkphase (Tiefpunkt nur in der Sinkphase)
 3. Touchen, um einen Tiefpunkt zu erzeugen
 4. Warten, bis der Vogel knapp über dem Boden ist
 5. Touchen, um nicht zu verlieren
- ▶ Zwei Messwerte für die Latenz: Oberer und unterer Tiefpunkt
 - ▶ Einlernphase mit festen Zeitabständen

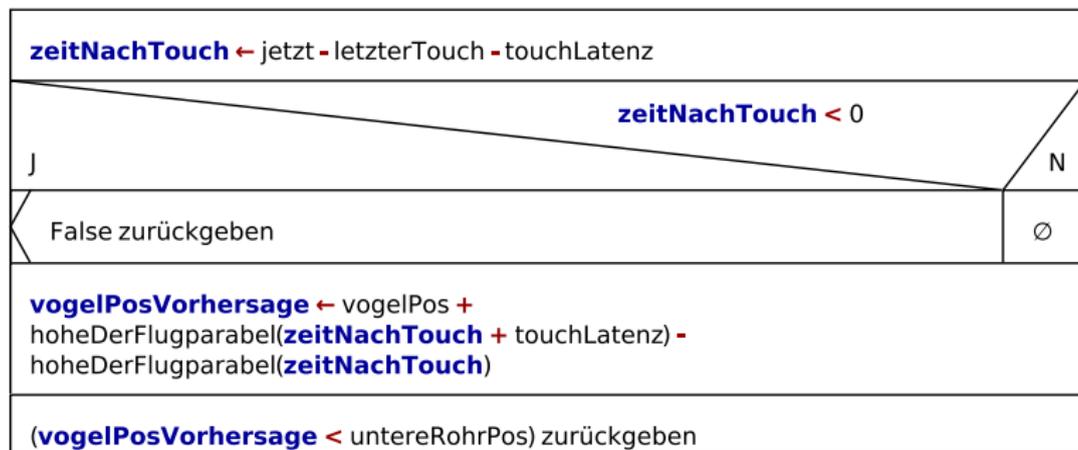


Spielphase

Operation solleTouchen(jetzt, vogelPos: GZ): Boolean der Klasse Steuerung

Attribute: letzterTouch, touchLatenz, untereRohrPos: GZ

Lokale Variablen: zeitNachTouch, vogelPosVorhersage: GZ



Gliederung

Das Spiel “Flappy Bird”

Aufbau von Hardware und Software

Kameraauswertung

Modell des Spiels

Algorithmus zur Spielersteuerung

Erzeugen der Touchereignisse

Bekannte Schwierigkeiten

Vorführung

Touch über den Servomotor

- ▶ Touchscreenstift an Servomotor
- ▶ Servomotor wird über Pulsdauer gesteuert
- ▶ Hardware-PWM vom Raspberry Pi
- ▶ Herstellerübergreifende PWM-Schnittstelle von Linux (sysfs)

Touch über Android Debug Bridge

- ▶ Android Debug Bridge (ADB)
 - ▶ USB-Verbindung
 - ▶ Ermöglicht das Ausführen von Shellskripten auf dem Smartphone
- ▶ Shellskript, das Touchereignisse erzeugt
- ▶ Screenshotfunktionalität wird nicht genutzt

Gliederung

Das Spiel “Flappy Bird”

Aufbau von Hardware und Software

Kameraauswertung

Modell des Spiels

Algorithmus zur Spielersteuerung

Erzeugen der Touchereignisse

Bekannte Schwierigkeiten

Vorführung

Bekannte Schwierigkeiten

- ▶ Touchereignisse werden vom Smartphone nicht erkannt, obwohl sich der Stift bewegt
- ▶ Zeitspanne bis zum ersten Hinderniss zufällig, Einlernphase in machen Fällen noch nicht fertig
- ▶ Fehlerhafte Positionserkennung in der Einlernphase

Gliederung

Das Spiel “Flappy Bird”

Aufbau von Hardware und Software

Kameraauswertung

Modell des Spiels

Algorithmus zur Spielersteuerung

Erzeugen der Touchereignisse

Bekannte Schwierigkeiten

Vorführung