

1 Recherche

1.1 Mikrocontroller

- [Raspberry Pi](#)
 - verschiedene Modelle, verschiedene Versionen, verschiedene Serien
 - unterstützt verschiedene Linux-Distributionen
 - [Raspberry Pi 1 Model A+](#)
 - * TODO: Spezifikationen?
 - [Raspberry Pi 3 Model B](#)
 - * CPU: Broadcom 64-Bit Quadcore 1200 MHz
 - * Memory: 1 GB
 - * Bluetooth, WLAN, LAN
 - * 4x USB 2.0
 - * HDMI-Out
 - [Raspberry Pi Zero W](#)
 - * CPU: Single core 1 GHz
 - * Memory: 512 MB
 - * Bluetooth, WLAN
 - * Mini HDMI
- [Arduino](#)
 - diverse Modelle
 - zum direkten Programmieren auf der Hardware, ohne Betriebssystem
 - [Arduino Uno Rev3](#) (im Starter-Kit enthalten)
 - * CPU: 16 MHz
 - * Flash Memory: 16 kB
 - * SRAM: 2 kB
- [LattePanda](#)
 - Für Windows 10 (Home Edition)
 - Linux sollte auch funktionieren: [Anleitung](#)
 - etwas teurer als Raspberry Pi + Arduino
 - [LattePanda 4G/64GB](#)
 - * CPU: Intel CherryTrail Atom X5 QuadCore 1.8 GHz
 - * Memory: 2-4 GB
 - * Flash: 32-64 GB
 - * LAN, WLAN, Bluetooth
 - * HDMI
 - * USB 2.0, USB 3.0
 - * enthält zusätzliches Arduino-Board
- [CubieBoard](#)
 - Diverse ARM-CPUs
 - Leistungsklasse ähnlich Raspberry Pi
 - tendenziell etwas grösser als Raspberry Pi für die gleiche Leistung

- schwieriger zu beschaffen

Der Raspberry Pi kommt unserer Vorstellung von einem “normalen Computer” sehr nahe, zumal sich auf ihm ein beliebiges Linux-Betriebssystem ausführen lässt. Beim Arduino hingegen wird direkt auf der Hardware programmiert, die, verglichen mit einem “normalen” Computer oder einem Raspberry Pi, sehr schwach ist. Der LattePanda kombiniert die beiden Welten auf einer Platine.

Für die gestellte Aufgabe wird es nötig sein, Bilder einer Kamera in Echtzeit auszuwerten. Dazu ist der Arduino zu schwach. Die Bildauswertung könnte jedoch auch den Raspberry Pi sehr stark fordern, sodass die Motorensteuerung evtl. nicht mehr flüssig laufen könnte. Von daher empfiehlt sich ein kombinierter Ansatz mit einem Raspberry Pi und einem Arduino bzw. mit einem LattePanda.

1.2 Software/Betriebssystem/Programmiersprache

- Plattform/Betriebssystem
 - Je nach Hardware beide Plattformen möglich: Windows oder Linux.
 - Für die gestellte Aufgabe ergibt der Betrieb einer grafischen Benutzeroberfläche keinen Sinn. Diese konsumiert nur unnötige Ressourcen.
- Bildverarbeitung/Machine Learning
 - [OpenCV](#)
 - * Linux, Windows, MacOS
 - * C, C++, Python, Java
 - [SimpleCV](#)
 - * Python
 - [Accord.NET Framework](#)
 - * Nur auf .Net? Mono?
 - * C# – und andere?
- Programmiersprache
 - Die Wahl der Programmiersprache ist v.a. von der Wahl der Software-Libraries abhängig
 - auf Arduino wird mit C/C++ programmiert