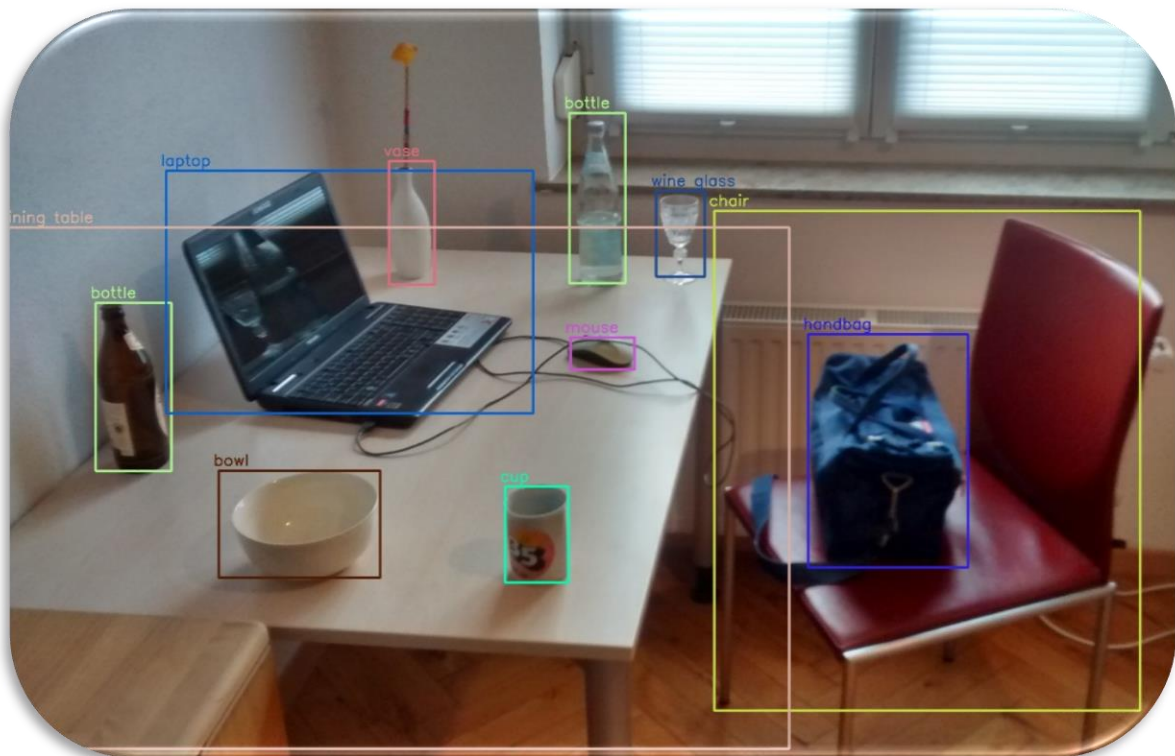


# Raspberry Pi - Objekterkennung



Autor: Mike Dätwyler  
E-Mail: [mike.daetwyler@bluewin.ch](mailto:mike.daetwyler@bluewin.ch)  
Website: [mike-daetwyler.ch](http://mike-daetwyler.ch)

# Inhaltsverzeichnis

Einleitung.....	3
Thema.....	3
Voraussetzungen .....	3
Zusätzliche Voraussetzungen für Ausgabe .....	3
Installation – Objekterkennung .....	4
Raspberry Pi OS - Installation .....	4
Kamera - Installation.....	7
Einrichtung.....	7
Edge TPU - Installation.....	8
Einrichtung.....	8
Programm .....	9
Einrichtung.....	9
Ausgabe speichern und wiederverwenden .....	10
In Datenbank speichern.....	10
In Website einbinden.....	11
Probleme – Lösungen .....	12
«cannot currently show the desktop» .....	12
ImageTk .....	12
Anhang .....	13
Materialverzeichnis .....	13
Softwareverzeichnis.....	13
Abbildungsverzeichnis .....	14
Literaturverzeichnis .....	14

# Einleitung

## Thema

Mit Hilfe einer Raspberry Pi Kamera sowie einer Edge TPU, sollen in Echtzeit Objekte erkannt werden. Die Kamera dient offensichtlich dazu, Bilder der Objekte aufzunehmen. Mit Hilfe eines Programms wird dann das gescannte Objekt mit Unterstützung einer Datenbank zugeordnet.

Damit das Ganze funktioniert, werden höhere Rechenleistungen vorausgesetzt als sie der Raspberry Pi allein bieten kann. Hier kommt die Edge TPU ins Spiel. Diese kann bei maximaler Ausschöpfung dem Raspberry Pi bis zu 20-mal mehr Rechenleistung bescheren.

In diesem Fall reicht die Standard-Konfiguration völlig aus. Diese gibt dem Raspberry Pi das fünffache an Rechenleistung.

Die Idee einer Objekterkennung erhielt ich von einem [Make-Artikel](#).

## Voraussetzungen

Kosten:	ca. 150.00 CHF
Kenntnisse:	Grundkenntnisse in Linux Grundkenntnisse in Python

## Zusätzliche Voraussetzungen für Ausgabe

Kenntnisse:	Grundkenntnisse in MySQL Grundkenntnisse in HTML/PHP
Umgebung:	MySQL-Datenbank Apache Webserver

# Installation – Objekterkennung

< Eine Software- sowie eine Materialliste finden Sie am Ende des Dokumentes im Anhang. >

## Raspberry Pi OS - Installation

1. Laden Sie den **Raspberry Pi Imager**<sup>1</sup> herunter.

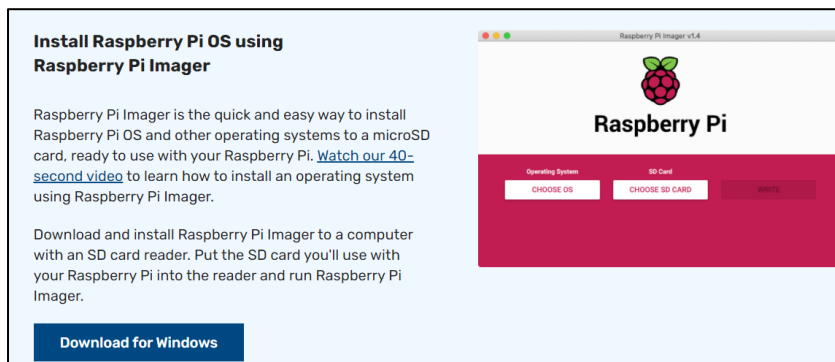


Abbildung 2: Imager - Download

! Achten Sie darauf, dass der Imager immer auf dem neusten Stand ist, um allfällige Komplikationen zu vermeiden.

2. Starten Sie den **Imager** und wählen Sie Ihre microSD-Karte, auf welche Sie das Betriebssystem installieren möchten.
3. Bevor Sie die microSD-Karte entfernen müssen Sie noch eine leere Datei namens «ssh» (ohne Dateiendung) auf die microSD-Karte kopieren.
4. Sie können nun die microSD-Karte an den Raspberry Pi anschliessen und diesen per Netzwerkanschluss mit Ihrem Netz verbinden.
5. Installieren Sie als nächstes **Putty**<sup>2</sup> auf Ihrem Computer.
6. Geben Sie hier die IP-Adresse des Raspberry Pis an. Es sollte bereits der richtige Port sowie «SSH» standartmässig eingetragen sein.

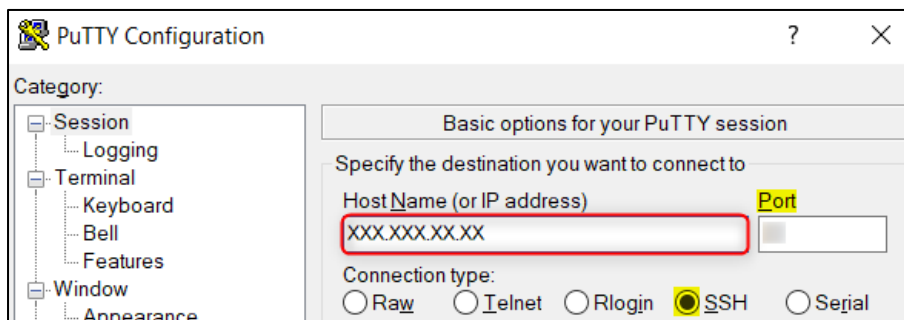


Abbildung 3: Putty - Benutzung

<sup>1</sup> <https://www.raspberrypi.org/software/>

<sup>2</sup> <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

Default-User	pi
Default-Password	raspberry

7. Geben Sie den Standardbenutzer sowie das zugehörige Standardpasswort an.  
(siehe oben)
8. Als nächstes sollten Sie direkt das Standardpasswort ändern und ein eigenes sicheres Passwort wählen.

command	passwd
---------	--------

9. Geben Sie als nächstes folgenden Befehl in die Konsole ein:

command	sudo raspi-config
---------	-------------------

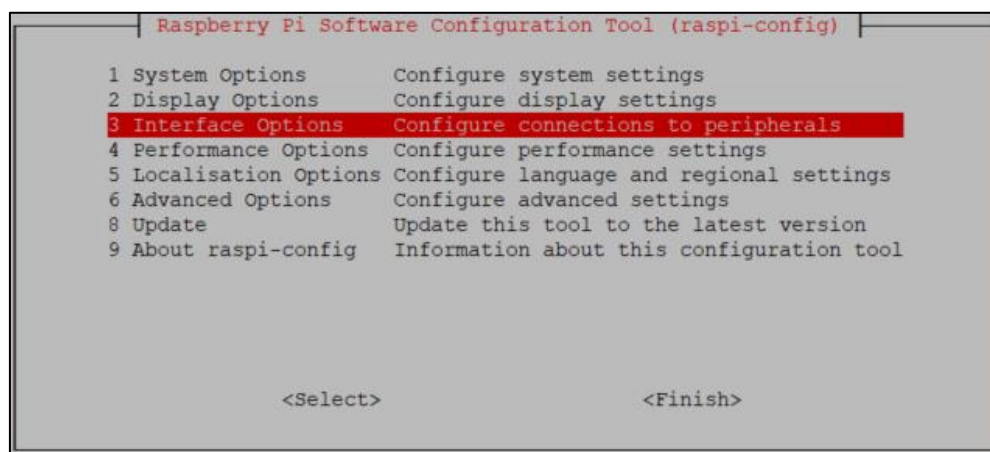


Abbildung 4: raspi-config

10. Hier gehen Sie zu «Interface Options» → «VNC» und wechseln Sie den Status zu «Enable».
11. Laden Sie als nächstes auf ihrem Computer den **VNC Viewer**<sup>3</sup> herunter.
12. Geben Sie hier die IP-Adresse an und melden Sie sich anschliessend mit Ihren aktuellen Benutzer-Daten an.

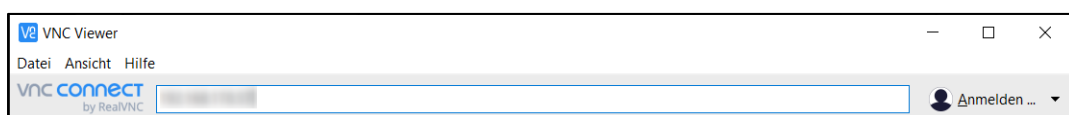


Abbildung 5: VNC Viewer - Benutzung

? **Problem-Lösung:**  
«cannot currently show the desktop»

<sup>3</sup> <https://www.realvnc.com/de/connect/download/viewer/>

13. Es empfiehlt sich zusätzlich das OS auf den neusten Stand zu bringen.

command	sudo apt update
command	sudo apt full-upgrade

## Kamera - Installation

Nach der erfolgreichen Inbetriebnahme des Raspberry Pis sowie dessen Betriebssystems, widmen wir uns nun der Kamera, welche die Objekte erkennen soll.

### Einrichtung

1. Schliessen Sie die Kamera am Raspberry Pi wie folgt an.

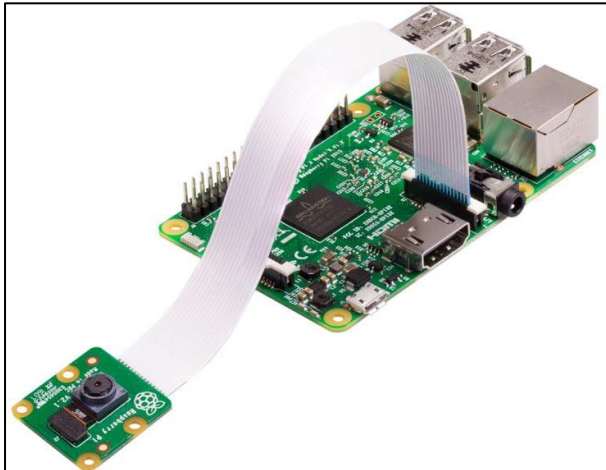


Abbildung 6: Kamera anschliessen

2. Aktivieren Sie als erstes den Kamera-Support des Raspberry Pis.  
Geben Sie den unteren Befehl ein und gehen Sie zu *Interface Options* → *Camera*

command

sudo raspi-config

3. Als nächstes sollten Sie nochmals Neustarten.

command

sudo raspi-config

4. Testen Sie nun Ihre Kamera:

command

raspistill -v -o test.jpg

## Edge TPU - Installation

Als nächstes installieren wir die Edge TPU. Hierbei ist es wichtig, dass Sie zuerst alles Software-Technische einrichten bevor Sie die Edge TPU physisch anschliessen.

### Einrichtung

1. Fügen Sie als erstes das **Debian package repository** zu Ihrem System hinzu.

command	echo "deb https://packages.cloud.google.com/apt coral-edgetpu-stable main"   sudo tee /etc/apt/sources.list.d/coral-edgetpu.list
command	curl https://packages.cloud.google.com/apt/doc/apt-key.gpg   sudo apt-key add -
command	sudo apt-get update

2. Nun installieren Sie die **Edge TPU runtime**:

command	sudo apt-get install libedgetpu1-std
---------	--------------------------------------

3. Schliessen Sie als nächstes die Edge TPU am Raspberry Pi an.
4. Damit Sie zusätzliche Funktionen erhalten, installieren Sie im nächsten Schritt die **PyCoral library**.

command	sudo apt-get install python3-pycoral
---------	--------------------------------------

Wählen Sie «y», um fortzufahren.



# Programm

Der Zeitpunkt ist gekommen, um den Programmcode auf den Raspberry Pi zu laden und auszuführen.

## Einrichtung

1. Laden Sie als erstes auf dem Raspberry Pi das benötigte Fehler! Verweisquelle konnte nicht g  
efunden werden. herunter.

(<https://github.com/robodhnb/SmartPiCam>)

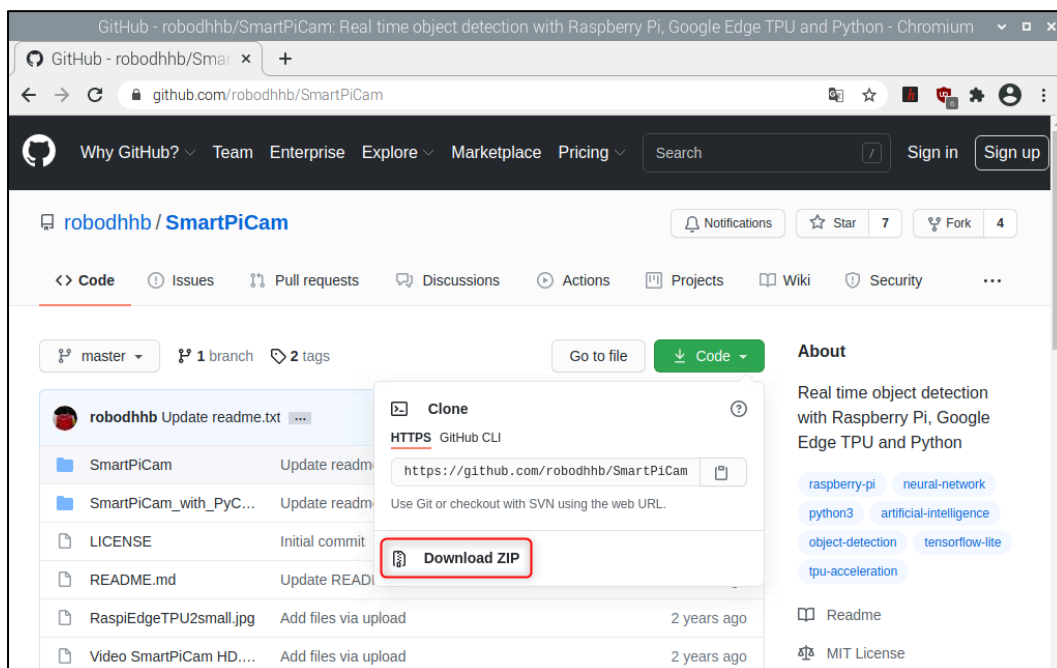


Abbildung 7: GitHub-Repository - Download

2. Extrahieren Sie das Skript auf den Desktop  
(/home/pi/Desktop)
3. Gehen Sie als nächstes in die Shell.
4. Wechseln Sie zum *SmartPiCam*-Ordner.

command

```
cd /home/pi/Desktop/SmartPiCam-master  
/SmartPiCam_with_PyCoral_API
```

5. Starten Sie nun das Programm.

command

```
sudo python3 smartPiCamMain.py
```

? **Problem-Lösung:**  
ImageTk

! «W0422 10:54:13.042898 2007 package\_registry.cc:65]  
Minimum runtime required by package (5) is lower than expected (10)»

Diese Meldung kann ignoriert werden, da es keine Auswirkung auf das  
Programm hat. Google arbeitet bereits an diesem Problem.

## Ausgabe speichern und wiederverwenden

Falls Sie das Ausgabene des Programms weiterverwenden möchten, zeigen wir Ihnen in diesem Kapitel wie Sie die Ausgabe in eine MySQL-Datenbank speichern können.

Wie Sie eine solche Datenbank auf Ihrem Raspberry Pi einbinden können, sehen Sie im Dokument zum LAMP-Server.<sup>4</sup>

### In Datenbank speichern

1. Das Programm läuft nun auf Ihrem Raspberry Pi. Nun müssen Sie den Programmcode bearbeiten, um Ihre Ausgabe speichern zu können. Überschreiben Sie dafür als erstes ihr *smartPiCamMain.py*-File mit dem folgendem: [smartPiCamMain.py](#)
2. Wechseln Sie nun zum *SmartPiCam*-Ordner:

command

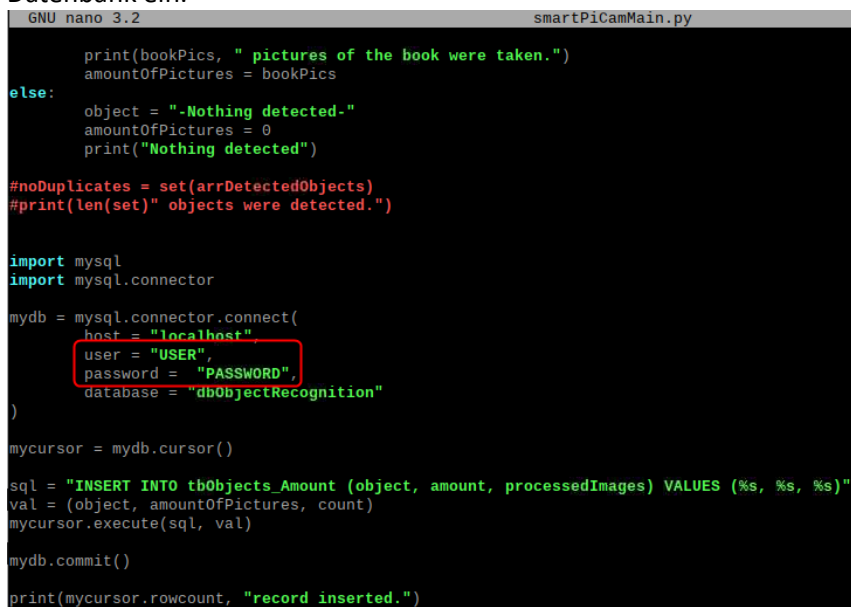
```
cd /home/pi/Desktop/SmartPiCam-master  
/SmartPiCam_with_PyCoral_API
```

3. Öffnen Sie als nächstes das File:

command

```
sudo nano smartPiCamMain.py
```

4. Gehen Sie hier bis zum untersten Block. Geben Sie hier Ihre Login-Daten Ihrer Datenbank ein.



```
GNU nano 3.2 smartPiCamMain.py  
  
print(bookPics, " pictures of the book were taken.")  
amountOfPictures = bookPics  
  
else:  
    object = "-Nothing detected."  
    amountOfPictures = 0  
    print("Nothing detected")  
  
#noDuplicates = set(arrDetectedObjects)  
#print(len(set) " objects were detected.")  
  
import mysql  
import mysql.connector  
  
mydb = mysql.connector.connect(  
    host = "localhost",  
    user = "USER",  
    password = "PASSWORD",  
    database = "dbObjectRecognition"  
)  
  
mycursor = mydb.cursor()  
  
sql = "INSERT INTO tbObjects_Amount (object, amount, processedImages) VALUES (%s, %s, %s)"  
val = (object, amountOfPictures, count)  
mycursor.execute(sql, val)  
  
mydb.commit()  
  
print(mycursor.rowcount, "record inserted.")
```

Abbildung 8: Anpassung - smartPiCamMain.py

Speichern Sie mit **ctrl+s** und verlassen Sie den Editor mit **ctrl+x**

<sup>4</sup> <https://mike-daetwyler.ch/sites/documentariesETC.php>

5. In Ihrer Datenbank müssen Sie nun folgendes erstellen:

- Datenbank = dbObjectRecognition
  - Tabelle = tbObjects\_Amount
    - Spalten = object | amount | processedImages

6. Als nächstes benötigen wir das MySQL-Modul auf dem Raspberry Pi:

command

`sudo apt-get install python3-mysqldb`

command

`sudo apt-get -y install python3-mysql.connector`

## In Website einbinden

Wenn Sie zusätzlich nun die Ausgabe von der Datenbank aus weiterverwenden möchten, können Sie dies beispielsweise per einer PHP-Website machen, welche eine Tabelle mit dem Resultat der Objekterkennung darstellt. Hier für benötigen Sie einen Apache Webserver und ein installiertes PHP-Modul. Wie Sie das auf Ihren Raspberry Pi installieren können, sehen Sie im Dokument [LAMP-Server](#).<sup>5</sup>

Befolgen Sie die darinstehenden Schritte zur Aufschaltung einer Website. Wenn Sie keine eigene PHP-Website coden wollen, können Sie hier eine downloaden: [resultObjectRecognition](#)<sup>6</sup>

Sie müssen dann lediglich wieder die Login-Daten anpassen.

Abbildung 9: Anpassung - index.php

<sup>5</sup> Siehe Fussnote 4

<sup>6</sup> [https://github.com/MtoseD/result\\_object\\_recognition](https://github.com/MtoseD/result_object_recognition)

# Probleme – Lösungen

## «cannot currently show the desktop»

Hierzu gibt es mehrere mögliche Lösungsansätze. Die folgenden Punkte sollte man kontrollieren:

- Ist das richtige Betriebssystem installiert? (mit Desktop)
- raspi-config
  - Sind die «Display-Settings» korrekt gesetzt? (meist *Default*)
  - Ist die «Boot-Autologin» Einstellung auf *Desktop Autologin Desktop GUI* eingestellt?<sup>7</sup>

command	sudo apt-get install lxsession
command	sudo apt-get install --reinstall libgtk2.0-0
command	sudo apt-get install --reinstall lxsession

## ImageTk

Falls das *ImageTk*-Paket nicht gefunden werden kann, müssen Sie es noch zusätzlich installieren:<sup>8</sup>

command	sudo apt install python3-pil.imagetk
command	sudo reboot

<sup>7</sup> <https://www.raspberrypi.org/forums/viewtopic.php?t=216737>

<https://www.tomshardware.com/how-to/fix-cannot-currently-show-desktop-error-raspberry-pi#:~:text=One%20reason%20that%20Raspberry%20Pi,or%20it's%20just%20not%20installed.>

<sup>8</sup> [https://github.com/robodhbb/SmartPiCam/blob/master/SmartPiCam\\_with\\_PyCoral\\_API/readme.txt](https://github.com/robodhbb/SmartPiCam/blob/master/SmartPiCam_with_PyCoral_API/readme.txt)

# Anhang

## Materialverzeichnis

Name	Quelle
Starter Kit	<a href="https://www.digitec.ch/de/s1/product/raspberry-pi-4-2g-model-b-starter-kit-armv8-entwicklungsboard-kit-11876692">https://www.digitec.ch/de/s1/product/raspberry-pi-4-2g-model-b-starter-kit-armv8-entwicklungsboard-kit-11876692</a>
Raspberry Pi 4 (2G; Model B; ARMv8)	<a href="https://www.digitec.ch/de/s1/product/raspberry-pi-4-2g-model-b-armv8-entwicklungsboard-kit-11267870">https://www.digitec.ch/de/s1/product/raspberry-pi-4-2g-model-b-armv8-entwicklungsboard-kit-11267870</a>
Official Raspberry Pi 4 Power Adapter	<a href="https://www.digitec.ch/de/s1/product/raspberry-pi-official-raspberry-pi-4-power-adapter-netzteil-elektronikzubehoer-gehaeuse-11268329">https://www.digitec.ch/de/s1/product/raspberry-pi-official-raspberry-pi-4-power-adapter-netzteil-elektronikzubehoer-gehaeuse-11268329</a>
OKdo Raspberry Pi 4 Model B Gehäuse	<a href="https://www.digitec.ch/de/s1/product/okdo-raspberry-pi-4-model-b-gehaeuse-gehaeuse-elektronikzubehoer-gehaeuse-11268336">https://www.digitec.ch/de/s1/product/okdo-raspberry-pi-4-model-b-gehaeuse-gehaeuse-elektronikzubehoer-gehaeuse-11268336</a>
microSD Karte (32GB)	<a href="https://www.digitec.ch/de/s1/product/intenso-3413480-microsd-32gb-speicherkarte-8396356?tagIds=77-535">https://www.digitec.ch/de/s1/product/intenso-3413480-microsd-32gb-speicherkarte-8396356?tagIds=77-535</a>
Raspberry Pi Kamera V2	<a href="https://www.digitec.ch/de/s1/product/raspberry-pi-board-v2-kamera-elektronikmodul-5999124">https://www.digitec.ch/de/s1/product/raspberry-pi-board-v2-kamera-elektronikmodul-5999124</a>
Google Coral USB Accelerator	<a href="https://buyzero.de/products/google-coral-usb-accelerator?_pos=1&amp;_sid=094931947&amp;_ss=r">https://buyzero.de/products/google-coral-usb-accelerator?_pos=1&amp;_sid=094931947&amp;_ss=r</a>

## Softwareverzeichnis

Name	Quelle
VNC-Viewer	<a href="https://www.realvnc.com/de/connect/download/viewer/">https://www.realvnc.com/de/connect/download/viewer/</a>
Putty	<a href="https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html">https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html</a>
Raspberry Pi - Imager	<a href="https://www.raspberrypi.org/software/">https://www.raspberrypi.org/software/</a>

## Abbildungsverzeichnis

Abbildung 1: Titelbild .....	1
Abbildung 2: Imager - Download.....	4
Abbildung 3: Putty - Benutzung.....	4
Abbildung 4: raspi-config.....	5
Abbildung 5: VNC Viewer - Benutzung .....	5
Abbildung 6: Kamera anschliessen .....	7
Abbildung 7: GitHub-Repository - Download .....	9
Abbildung 8: Anpassung - smartPiCamMain.py .....	10
Abbildung 9: Anpassung - index.php .....	11

## Literaturverzeichnis

- Bootstrap. (2021). *Bootstrap*. Retrieved from <https://getbootstrap.com/>
- Coral. (2020). *Get started with the USB Accelerator*. (Google, Editor) Retrieved Mai 18, 2021, from coral.ai: <https://coral.ai/docs/accelerator/get-started/>
- Dätwyler, M. (2020). *Raspberry Pi - Lamp Server.pdf*. Retrieved Mai 28, 2021, from <https://drive.google.com/file/d/1myvSieNwEwhD-2MfbJRI9wYFVAjZoweA/view?usp=sharing>
- Forum. (2020). *How do I count unique values inside a list*. Retrieved Mai 28, 2021, from stackoverflow.com: <https://stackoverflow.com/questions/12282232/how-do-i-count-unique-values-inside-a-list>
- Forum. (2020). *installing mysql connector for python 3 in raspberry pi*. Retrieved Mai 28, 2021, from Stackoverflow: <https://stackoverflow.com/questions/38007240/installing-mysql-connector-for-python-3-in-raspberry-pi>
- Karim. (2021). *How to Check if a Python String Contains Another String?* Retrieved Mai 28, 2021, from afternerd.com: <https://www.afternerd.com/blog/python-string-contains/#:~:text=You%20can%20use%20the%20in,%2C%20otherwise%20%2D1%20is%20returned.>
- Make Magazin. (2019). *Objekterkennung mit Pi-Kamera und Edge TPU*. Retrieved Mai 14, 2021, from heise.de: [make-magazin.de/xsnm](https://make-magazin.de/xsnm)
- Putty. (2020). *Putty Download*. Retrieved Februar 2, 2021, from <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>
- Raspberry Pi. (n.d.). Retrieved Februar 2, 2021, from Raspberry Pi OS: <https://www.raspberrypi.org/software/>
- Raspberry Pi. (n.d.). *Camera configuration*. Retrieved Mai 18, 2021, from raspberrypi.org: <https://www.raspberrypi.org/documentation/configuration/camera.md>
- robodhhb. (2021). *SmartPiCam Repository*. Retrieved Mai 18, 2021, from github.com: <https://github.com/robodhhb/SmartPiCam>
- The Net Ninja. (2019). *PHP Tutorial for Beginners*. Retrieved Juni 4, 2021, from <https://youtube.com/playlist?list=PL4cUxeGkcC9gksOX3Kd9KP0-O68ncT05o>
- VNC. (n.d.). *VNC Viewer Download*. Retrieved Februar 2, 2021, from <https://www.realvnc.com/de/connect/download/viewer/>
- W3Schools. (2021). *PHP, Python, MySQL*. Retrieved from w3schools: <https://www.w3schools.com/>
- What Object Categories / Labels Are In COCO Dataset?* (2018). Retrieved Mai 21, 2021, from Amikelive | Technology Blog: <https://tech.amikelive.com/node-718/what-object-categories-labels-are-in-coco-dataset/>