Stockholm Beamer Theme

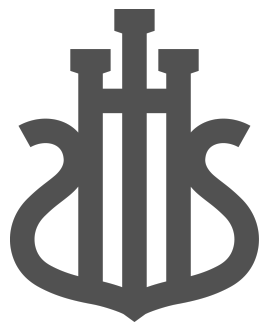
sthlm v2.0.2 is based on hsrm & mTheme

MarkOlson.SE

Updated: 2017/05/16

Made in *Sweden*

File: 20150731-081156-rs2.2B-sthlmBeamerTemplate



Overview

1. General Information
2. Colors
3. Introduction
4. What is RaspberryPi?
5. Initialize RaspberryPi
6. Initialize RaspberryPi
7. Initialize RaspberryPi

General Information

sthlm Theme Information

sthlm theme was originally designed to bring pdflatex support and color to the unique beamer hsrm theme designed by Benjamin Weiss. Thank You Ben!

https://goo.gl/NRseuc

Since then, sthlm has borrowed heavily from mTheme developed by Matthias Vogelgesang.

sthlm Theme Information

sthlm continues to be a theme that can easily be modified through the style files. If you are looking for a packaged theme, then I highly recommend mTheme.

I use a custom version of sthlm for daily decks and make a vanilla version of the theme available for others to use and modify. - Enjoy!

sthml Build Information

sthlm theme has been designed and tested to work within the SageMathCloud (Linux) environment.

Warning of Build Issues

I cannot guarantee that the code used to create the sthlm theme is *error free, optimized, well written* nor *if it will work in your production environment*.

Have Fun!

If you have read this far, then you are probably interested in using / modifying this theme for your own project.

Everything you need is in the

- style files:
  - `beamerthemesthlm.sty`,
  - `beamerfontthemesthlm.sty`,
  - `beamercolorthemesthlm.sty`.

Theme Package Requirements

This theme requires that the following packages are installed:

- *beamer*
- *backgrounds*
- *booktabs*
- *calc*
- *datetime*
- *ragged2e*
- *tikz*

There is always the option of simplifying the theme to reduce the number of required packages.

Replace the Logo With Your Own

The Sigtunaskolan Humanistiska Läroverket logo, `logo.png`, should be replaced with your own. I teach within the Mathematics Institution at SSHL.

Figure: SSHL Logo

Theme Options

Option	Description
`newPxFont`	newpxtext and newpxtext fonts will be used (pdfLaTeX)
`progressbar`	Frame Title progress bar
`sectionpages`	Section pages
`fullfooter`	Footers with logo
`numfoooter`	Footers with page number only
`greybg`	Frame background default is set to grey
`cblock`	Blocks with colored background
`protectFrameTitle`	Protect the frame title (if needed)
`valigncolumns`	Vertically align columns

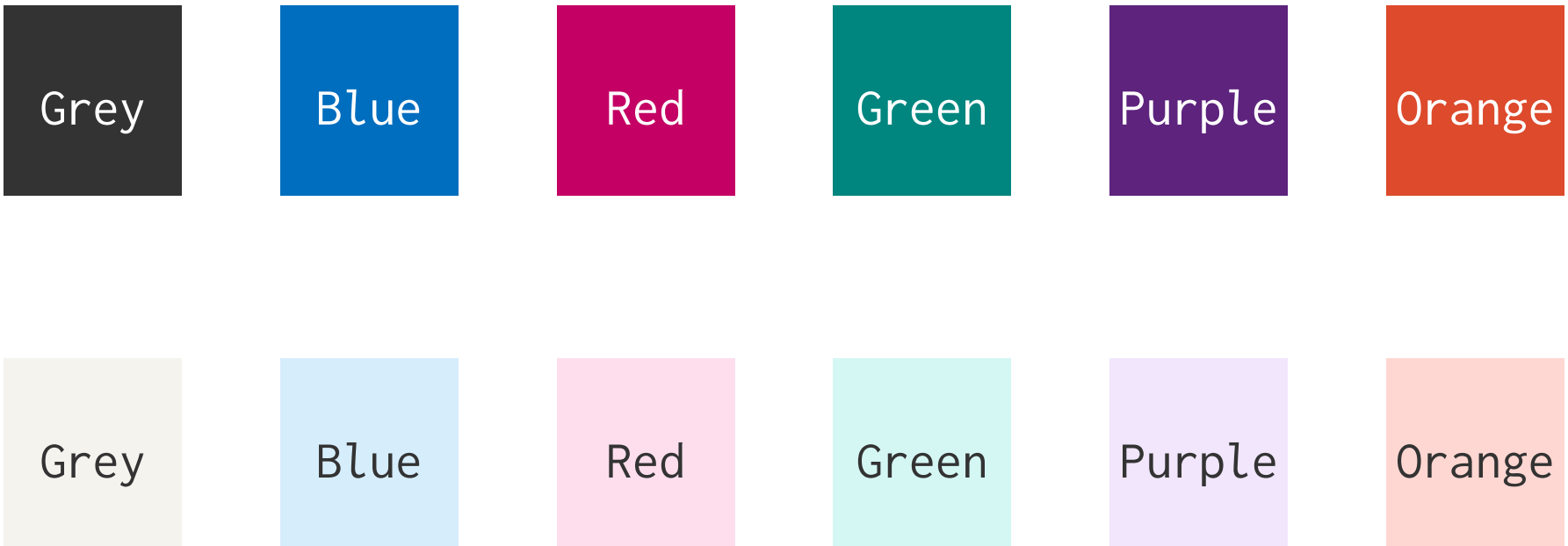
Colors

Color Style File

The sthlm theme style file `beamerthemesthlm.sty` references the `beamercolorthemesthlm.sty` file for the theme colors automatically.

If you wish to bring your own color theme, then you will have to either change the reference in the `beamerthemesthlm.sty` file or rename your style file to `beamercolorthemesthlm.sty`.

Primary Presentation Colors



Colored Text

Table: Colored Text

Red	LightRed	Red
Blue	LightBlue	Blue
Green	LightGreen	Green
Purple	LightPurple	Purple
Orange	LightOrange	Orange
Grey	Grey	DarkGrey

Introduction

Hintergrund

- Methodology
  Initialize Rasperripi and camera unit
  Open CV description
  Face recognition
  motion detector
- Results
  Face recognition
  Motion detector
- Weakness of project&future improvement

What is RaspberryPi?

Hintergrund

- RaspberryPi is a micro-computer.
  Contains the same function as normal size computer.
  The operation system is Linux.

Figure: RaspberryPi

Initialize RaspberryPi

Hintergrund

- Install and upgrade operation system
- Connect camera unit on RaspberryPi
- Initialize camera function

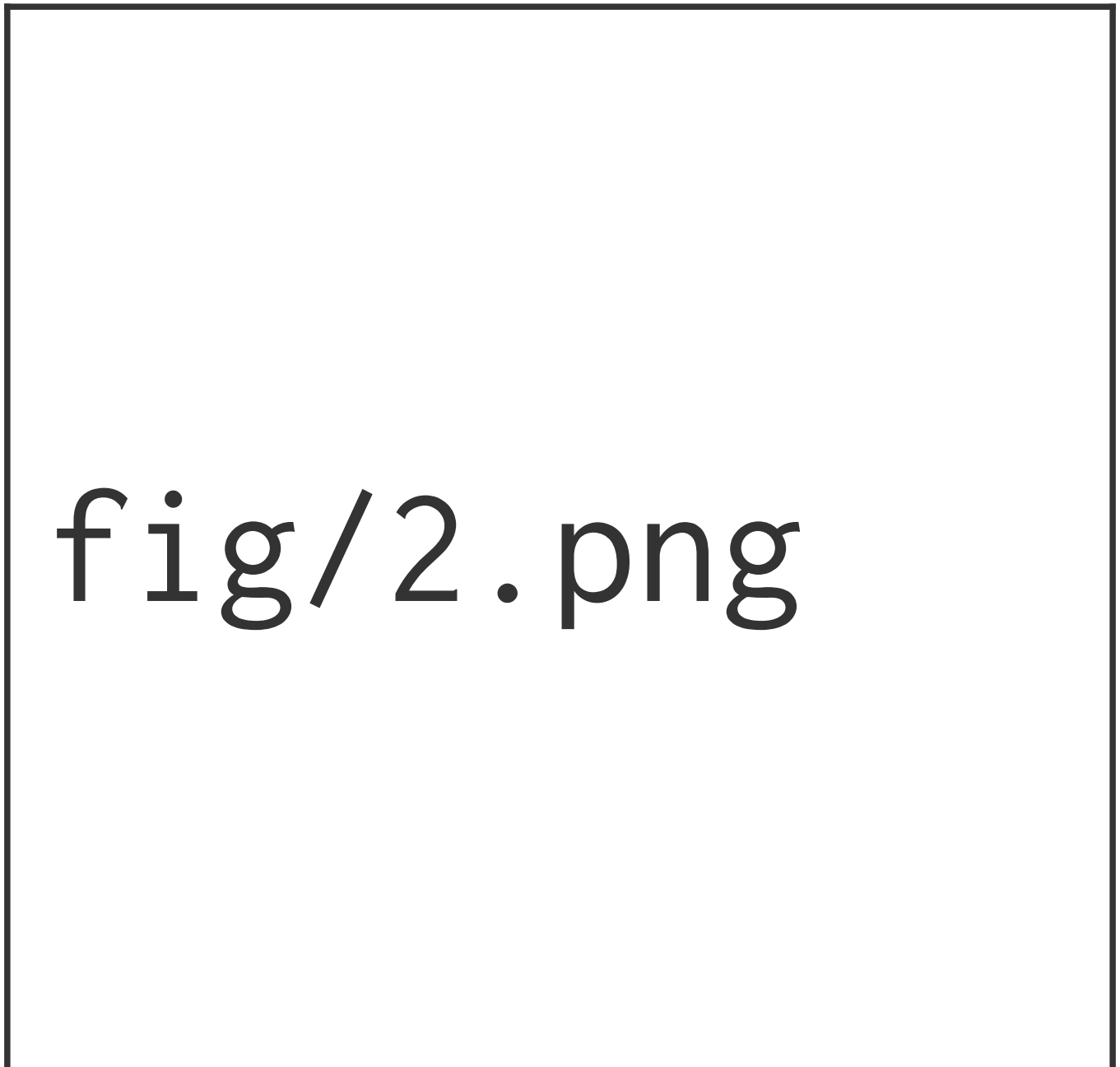
Initialize RaspberryPi

Figure: Beginning page

Initialize RaspberryPi

fig/3.png

Initialize RaspberryPi


fig/2.png

HUGEOpen CV

haar-like feature

haar-like feature

Haar-like feature:

- digital image features for real-time face detector
- quantize face features for calculation

haar-like feature

Haar-like feature is defined as:

- $$F_{Haar} = \frac{E(R_{black}) - E(R_{white})}{\sqrt{|E(R_{\mu})^2 - E(R_{\mu}^2)|}}$$

adaboost

adaboost

stage 1

For optimal weak classifier: setting threshold value

stage 2

For strong classifier: weighting sum on misjudgment rate

stage 3

For cascade classifier: serial connecting strong classifiers

outcome

outcome

characteristic: ordinary true accept rate and low false accept rate

example

If 99% photos with faces and 50% photos without faces can pass 20 cascaded classifiers, the total true accept rate is $0.99^{20} = 82\%$ and the total false accept rate is $0.5^{20} = 0.0001\%$.

We can directly call face cascade in our code.

HUGEFace recognition description

Face Tracking

Face Tracking

1. Camera setup and video recording
2. Face recognition with Cascade classifier
3. Face tracking in rectangle
4. Improvement in increasing FPS

Camera setup and video recording

```
#load picamera library
import picamera



#setup camera
camera = picamera.PiCamera()
camera.resolution = (320, 240)
camera.framerate = 60
rawCapture = PiRGBArray(camera,size=(320,240))
```

Face detection with Cascade classifier

Face recognition with Cascade classifier

```
#import opencv library
import cv2

# Load a cascade file for detecting faces
face_cascade = cv2.CascadeClassifier('file path')

# Detect faces with cascade classifier
gray = cv2.cvtColor(image,cv2.COLOR_BGR2GRAY )
faces = face_cascade.detectMultiScale(gray)
```

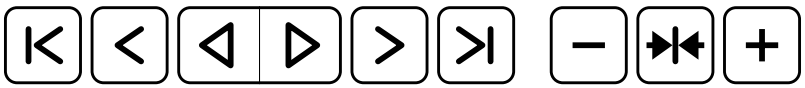
Face tracking in rectangle

```
while(1):
#face detection

# Draw a rectangle around every face
for (x, y, w, h) in faces:
cv2.rectangle(image,(x,y),(x+w,y+h),(0,255,255),2)
```

Execute in a while() loop to track face in every frame: LOW FPS!

Demo 1



Run face detection every four frames

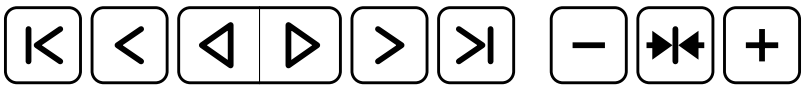
```
#Initialize frame counter
fcounter = 0
#Read frames from camera to array
for frame in camera.capture_continuous():
image = frame.array

# Run face detection every four frames
if fcounter == 3:
fcounter = 0
#run face tracking algorithm

else:
#run face tracking algorithm

fcounter += 1
```

Demo 2



HUGEMotion Detector

Gemeinschaftsprojekt

Motion detector:

Detect the movement of the object

total.png

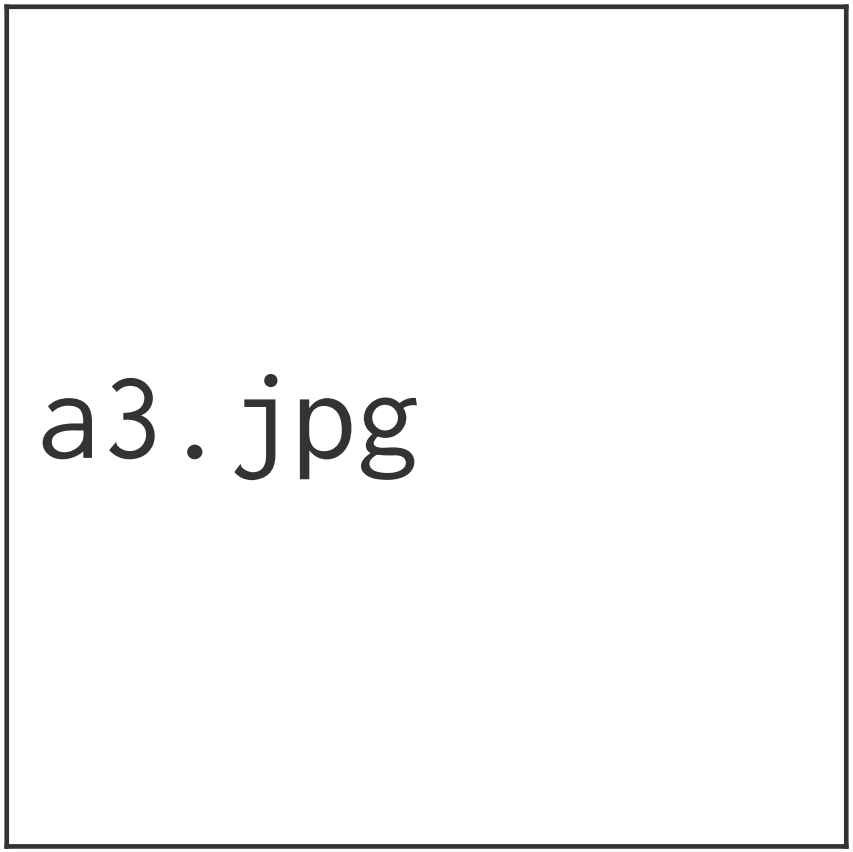
detect.jpg

Motion Detector

Hintergrund

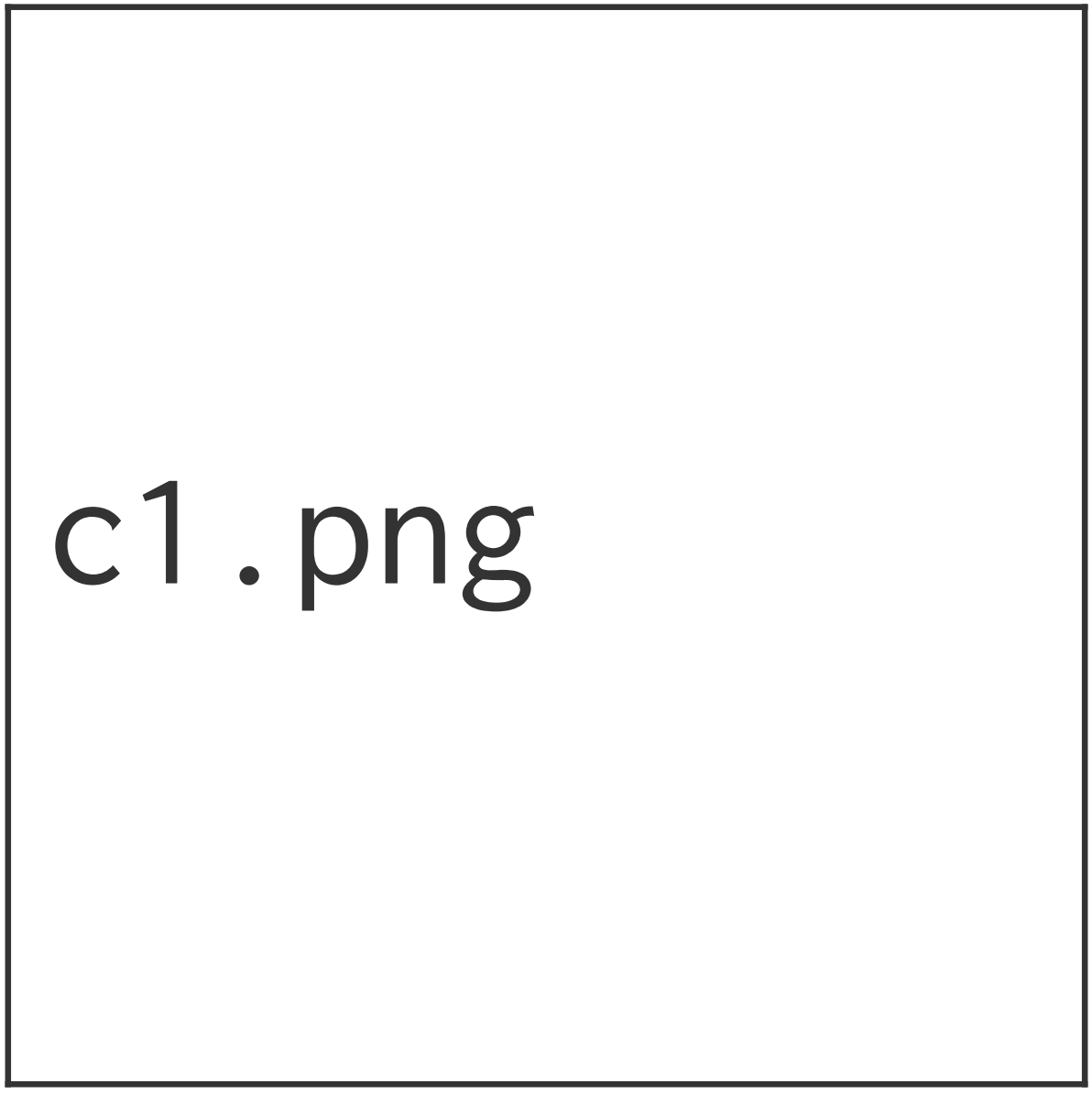
Motion Detector

Hintergrund



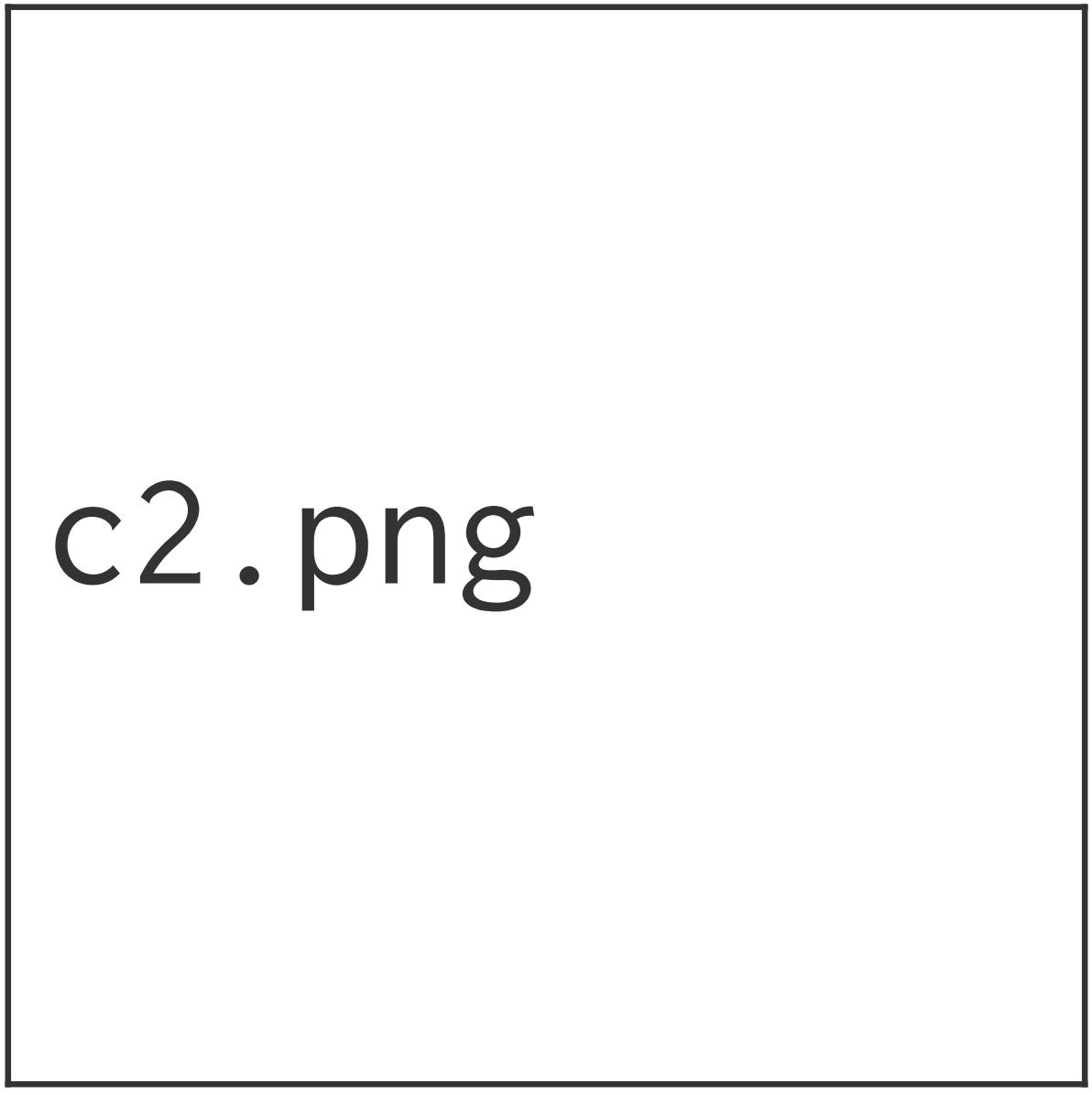
Motion Detector

Hintergrund



Motion Detector

Hintergrund



HUGEDifficulties and Improvements

Difficulties and Improvements

Difficulties

The difficulties we met:

- Remote control software can lead to time delay.
- Motion detection function is affected by illumination intensity.
- For face detection, the accuracy and the fluency cannot be assured simultaneously.
- The heating problem is severe.

Feasible Improvements

Some feasible improvements:

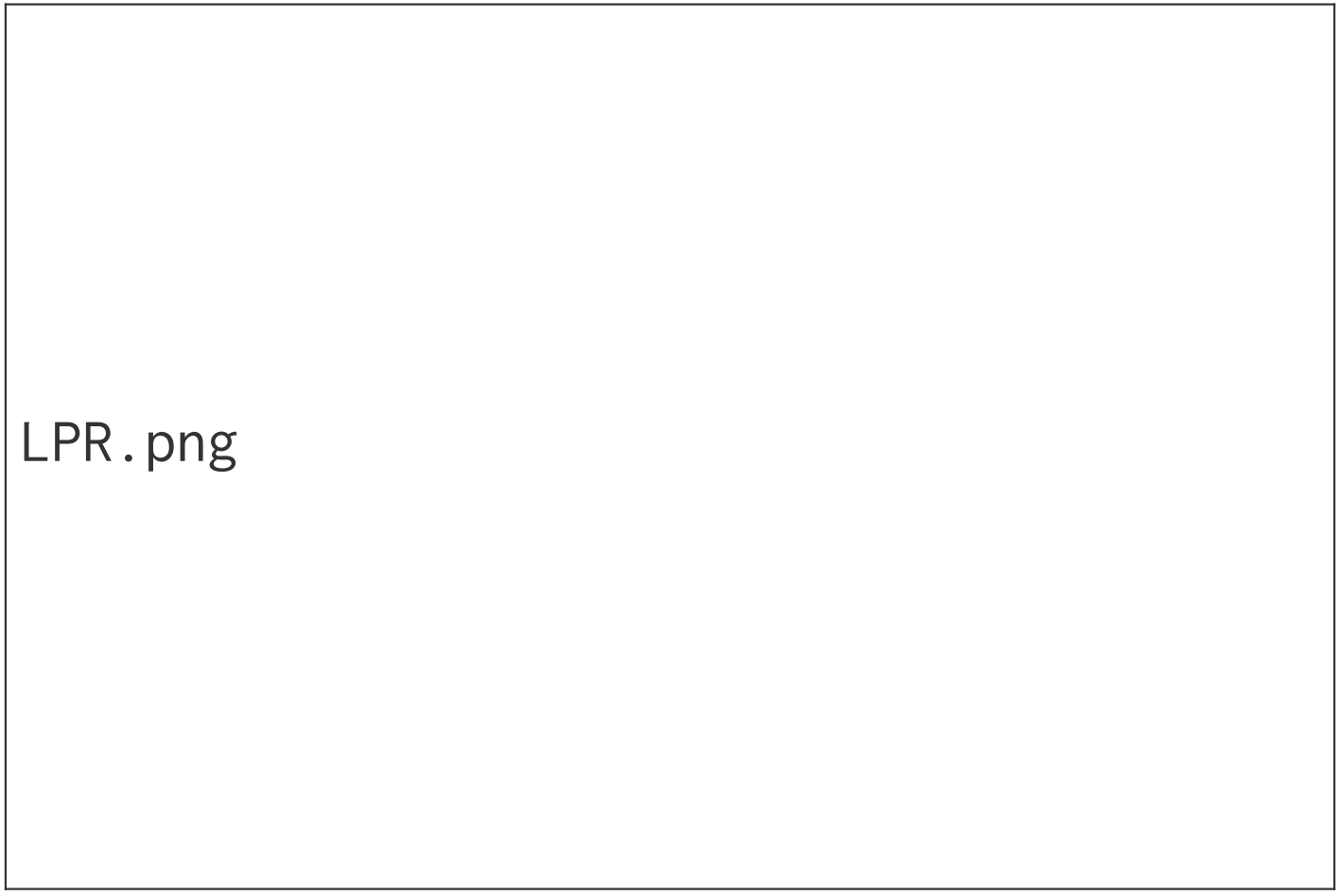
- Apply Real VNC to remote control raspberry.
- Apply pyroelectric infrared sensor.
- Further study is necessary.
- Apply more advanced device.

HUGEDifficulties and Improvements

Further Functions: LPR System

Further Improvments: LPR System

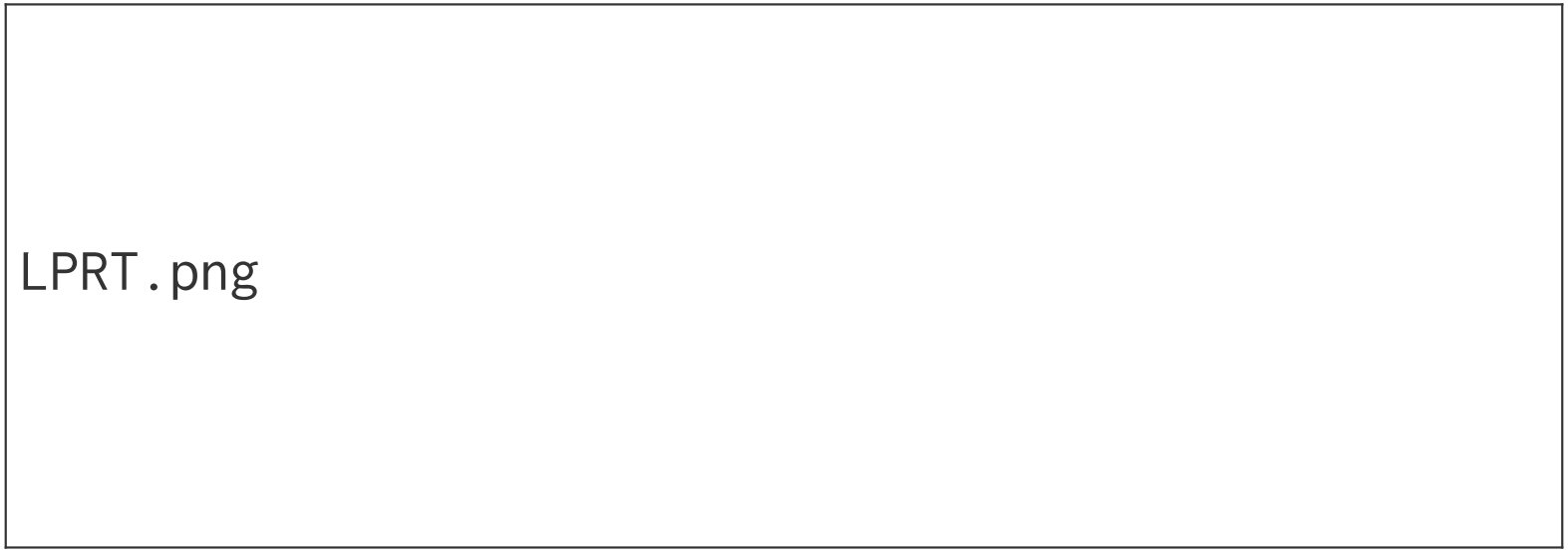
Application: Access Control System



Further Improvments: LPR System

License plate recognition system

- Definition: Recognize license plate of vehicle
- Theory: Emphasize the boundaries
- Programm: OpenCV

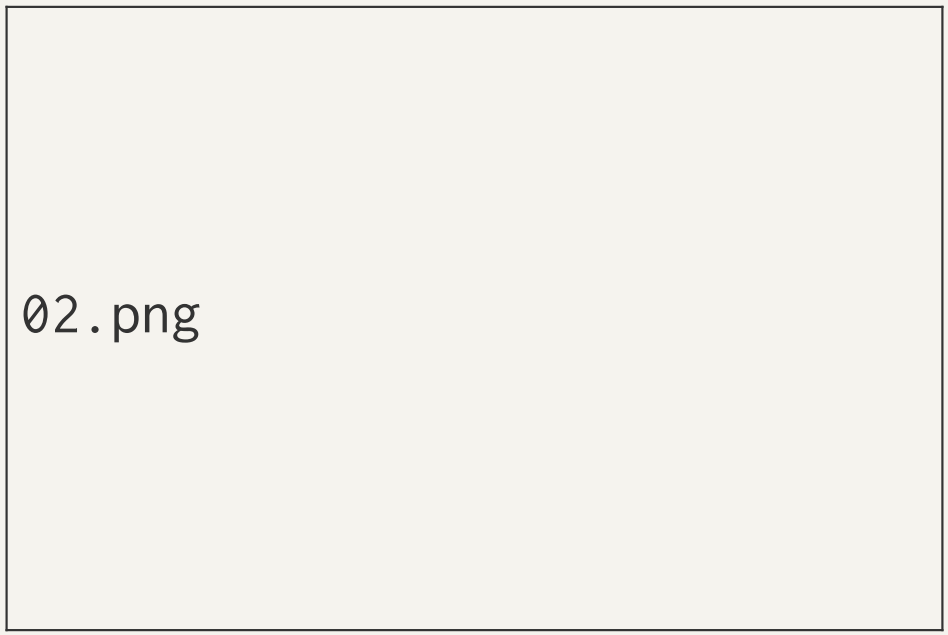


Further Functions: PT Camera Control

Further Improvements: PT Camera Control

Pan-Tilt-Zoom (PTZ) Camera Control

- Definition: Specific Camera that can Change direction
- Theory: Change Directions in X and Y axis
- Device: Pan-Pi Holder



Conclusion

Conclusion

Achievements

1. Build the Complete structure of Surveillance System
2. Accomplish various of functions: Record Video; Movements Detection; Face Recognition...

Drawbacks

1. Inaccuracy of Face Recpgnition
2. Surrounding Light influence the Motion Detection
3. Heating Issues in Processor

Further functions

1. LPR system
2. PT camera control

Multiple Columns

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam.

- Point 1
  - Sub point a
  - Sub point b
- Point 2

References

About

If you have any questions or comments

- Website: markolson.se
- Twitter: @markolsonse
- Instagram: @markolson.se

THE END