

Py_to_PDF

May 8, 2025

```
[ ]: #!/usr/bin/env python3
"""
Configuration module for water-line detection system.

This file defines a global CONFIG dictionary containing all
user-adjustable parameters for the main cycle, capture, processing,
and cropping routines. Future maintainers can modify paths,
processing parameters, or hardware settings here.
"""

# Main system selection: choose between PC or Raspberry Pi execution
CONFIG = {
    # "raspberrypi" uses GPIO and Linux paths; "pc" uses local Windows paths
    "system": "pc",

    # File paths for image input, results output, and offline uploads
    "paths": {
        "pc": {
            # Directory where new images are stored
            "image_path": r"C:\\Users\\bjorn\\Desktop\\Studie\\Graduation\\01.
↳ THESIS\\Scripts\\wd_directory\\wd_data",
            # Directory where processed results will be saved
            "output_path": r"C:\\Users\\bjorn\\Desktop\\Studie\\Graduation\\01.
↳ THESIS\\Scripts\\wd_directory\\wd_results",
            # Directory for saving uploads when offline
            "offline_uploads": r"C:
↳ \\Users\\bjorn\\Desktop\\Studie\\Graduation\\01.
↳ THESIS\\Scripts\\wd_directory\\offline_uploads"
        },
        "raspberrypi": {
            # Base image directory on Raspberry Pi
            "image_path": "/home/bjorn/Desktop/wd_directory/wd_data",
            # Base output directory on Raspberry Pi
            "output_path": "/home/bjorn/Desktop/wd_directory/wd_results",
            # Offline uploads directory on Raspberry Pi
            "offline_uploads": "/home/bjorn/Desktop/wd_directory/
↳ offline_uploads"
```

```

    }
},

# ORC (Open River Cam) API credentials and endpoint
"orc": {
    "base_url": "https://openrivercam.com/api", # API endpoint
    "username": "a.e.c.rens@student.tudelft.nl", # ORC account username
    "password": "JB$R3BS6txD8Y#di", # ORC account password
    "site_id": 12 # Identifier for the
↪camera site
},

# Capture settings: how often and how many images per burst
"capture_params": {
    "burst_intervals": [0.05], # seconds between frames in a burst
    "cycle_interval": 5, # seconds between bursts in one cycle
    # Desired camera resolution [width, height]
    "resolution": [640, 480]
},

# Image-processing parameters for waterline detection
"processing_params": {
    "angle": -3, # rotation angle (degrees) to deskew image
    "box_height": 10, # vertical height of comparison boxes (px)
    "min_distance": 10, # minimum peak distance for find_peaks (px)
    "sigma": 10, # Gaussian smoothing sigma for probability curve
    # Choose metric: "mean" for mean-difference, "ks" for Kolmogorov-Smirnov
    "diff_method": "mean"
},

# Cropping parameters: pixel coordinates in the rotated image
"crop_params": {
    # Left, top, right, bottom boundaries of crop box (px)
    "left": 297,
    "top": 170,
    "right": 350,
    "bottom": 273
},

# Rest time between full cycles (seconds)
"cycle_rest_seconds": 600,

# Debug flags
"debug_mode": False, # If True, enable additional debug logging
"dummy_mode": False # If True, skip actual capture and use dummy data
}

```