

VLAIO TETRA

Machine Vision for Quality Control

(MV4QC)

Case 4 - Determination of the proper functioning of a LED display

Case 4



Can we utilize machine vision to evaluate the proper functioning of the various components of a forklift display?

Methodology: Data

No machine learning model



Dataset different



Images with different states of the display




Problems?

- Lighting
- Reflection
- position

Methodology: Setup

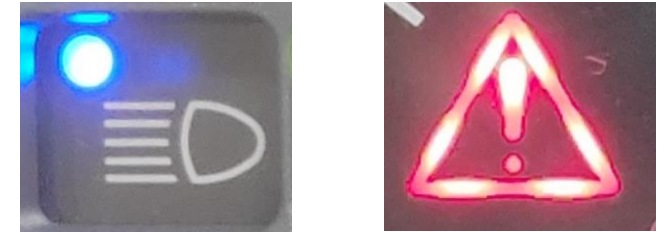
Traditional Computer Vision  No machine learning model

 Using only computer vision techniques

- Image thresholding
- Edge detection
- Finding contours
-

Methodology: Detection of Light

1: Take the component



2: Filter out non relevant data



Only want the active light

- HSV filtering
- Image thresholding

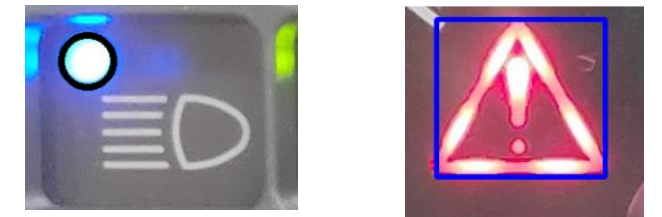


3: Find the source



Find the source of the Led by using contour detection

- Led circular
- Other shapes




Methodology: Evaluation of Gauge

1: Take the component  Pointer indicates value

2: Filter out relevant data  Only want the pointer

3: Plot needed lines/ploints  Using the filtered lines and extra validation lines/ploints

4: Extract value  Convert it to a value

- Convert in between bounds
- Margin of error



Methodology: seven-segment display.

7 segment

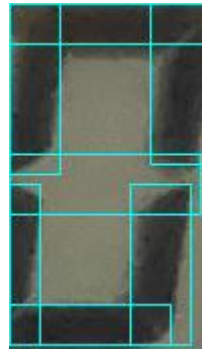


Number indicated by active segments



Detect the value

- Extract number
- Inspect segments
- Map states to numbers



```
DIGITS_LOOKUP = {  
  (1, 1, 1, 1, 1, 1, 0): 0,  
  (0, 1, 1, 0, 0, 0, 0): 1,  
  (1, 0, 0, 0, 1, 1, 1): 1,  
  (1, 1, 0, 1, 1, 0, 1): 2,  
  (1, 1, 1, 1, 0, 0, 1): 3,  
  (0, 1, 1, 0, 0, 1, 1): 4,  
  (1, 0, 1, 1, 0, 1, 1): 5,  
  (1, 0, 1, 1, 1, 1, 1): 6,  
  (1, 1, 1, 0, 0, 0, 0): 7,  
  (1, 1, 1, 1, 1, 1, 1): 8,  
  (1, 1, 1, 1, 0, 1, 1): 9  
}
```

```
(1, 1, 1, 1, 1, 1, 0): 0,
```



Alternative one could use
an OCR system

Methodology: board computer

See if 4 symbols are present

Template matching

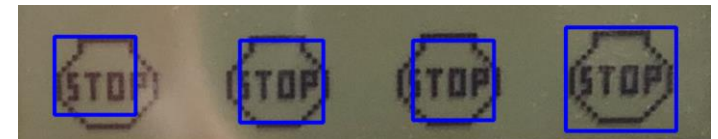
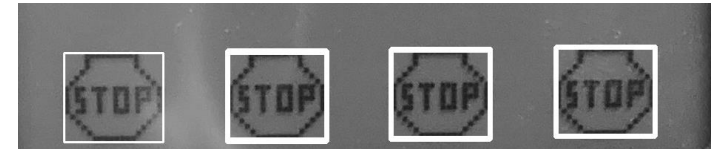
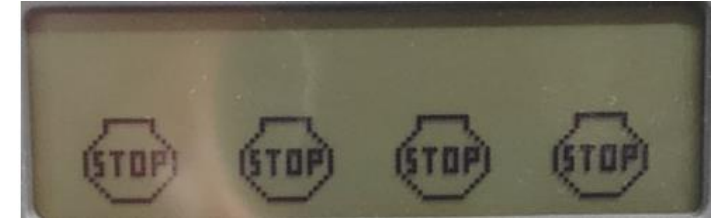
- Computationally expensive
- Searches for areas not symbols
- Sensitive to noise

Contour detection

- Faster and cheaper
- Searches for contours of exact size
- Less sensitive to noise



Preferred option



Methodology: Traction

What does the traction indicate?



Value indicated by a collection of black squares



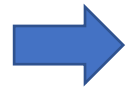
We need to see if all squares work meaning we test the maximum value



Look if we can detect the fully active bar

Methodology: Backlight

Backlight not visible



Influences other
components

We check if the color of a representative component is different

Results: Light detection

Multiple tests:

Detecting if all LEDs are on



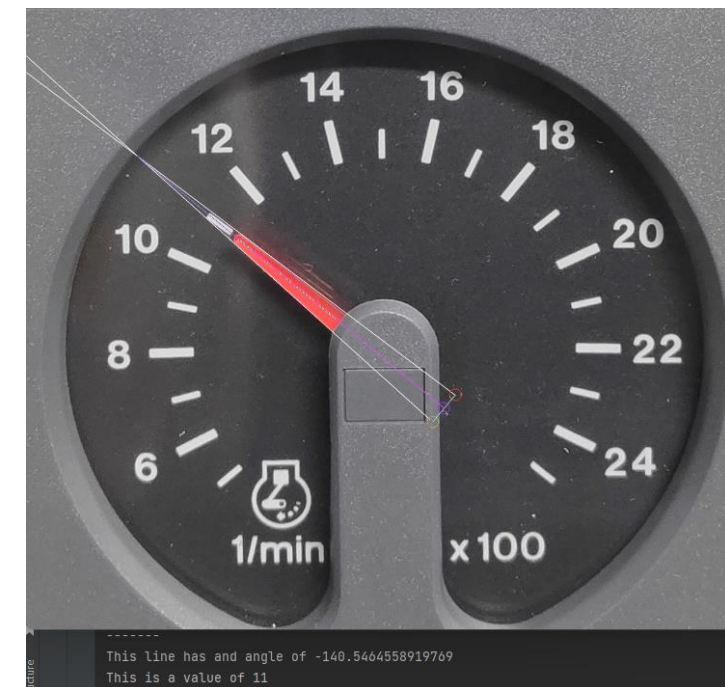
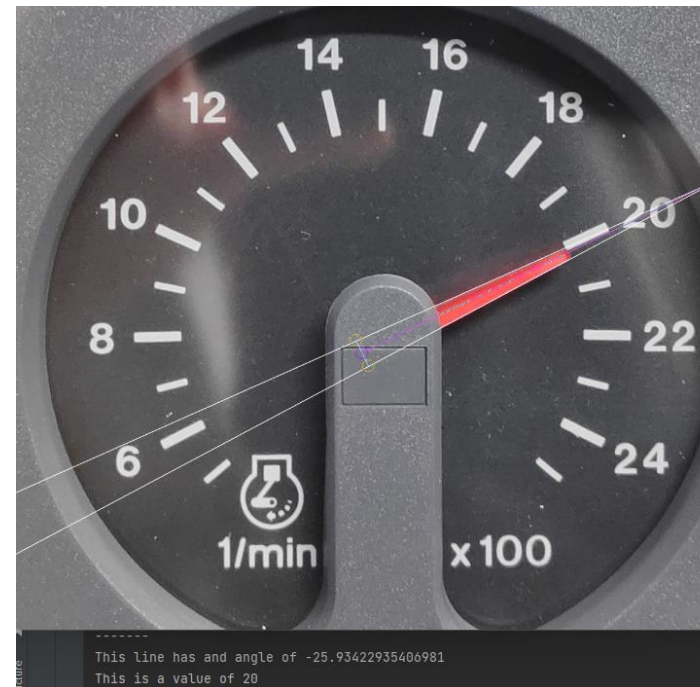
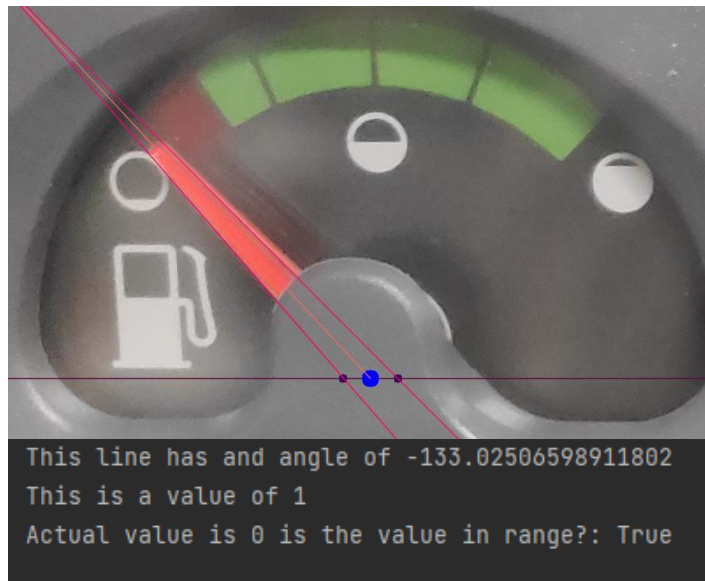
Detecting state of different LEDs



Using similar techniques, we can also detect an active warning symbol



Results: Analog instruments



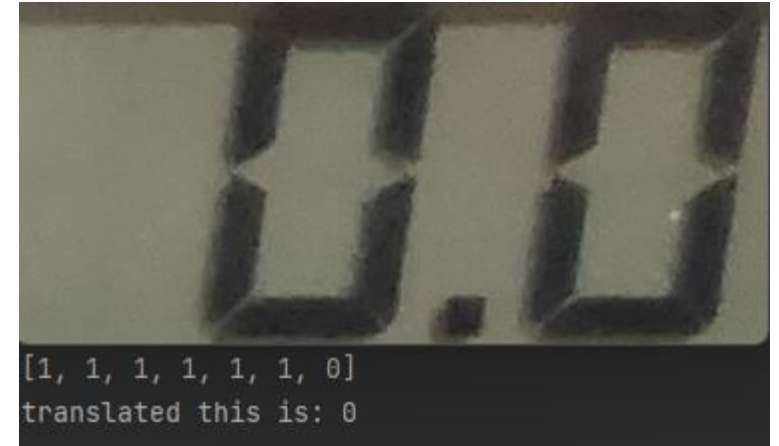
Results: seven-segment

Evaluation works most of the time

➡ Problems can occur

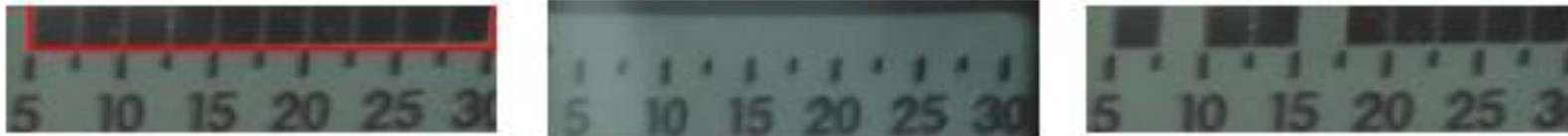


➡ Due to inconsistent lighting and reflection



Results: Traction meter

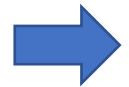
Using contour detection, we can detect the full bar



Inactive and faulty values do not pass

Results: Backlight detection

Board computer represents the status of backlight



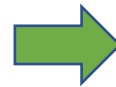
Filter out unwanted colors



Calculate the percentage of active pixels



Calculate the percentage of active pixels



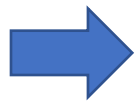
If more than 80% of the pixels are active the display is on



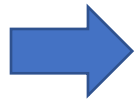
Results: Problems

Previous tests sometimes fail

- Inconsistent lighting
- Inconsistent distance
- Internal/external reflection



Lighting and reflection can be solved with better hardware



Positioning can be solved using a fixed setup



Future research on the implementation in industrial setting?

Results: Hardware testing

Best lighting? How to avoid reflection?



Earlier tests using IDS camera and polarised lights



Polarised Lens



Lights on long edge



Light perpendicular on display