Precision at Pixel Level: YOLO doing UI test automation

© Nikolaus Rieder

FH University of Applied Sciences TECHNIKUM WIEN

A few things about me

- 7 years work experience in Quality Assurance (Testing)
- Developing test automation for integration and system testing
- Working with distributed embedded systems
  - Nurse call system

FH University of Applied Sciences TECHNIKUM WIEN

SCHRACK SECONET

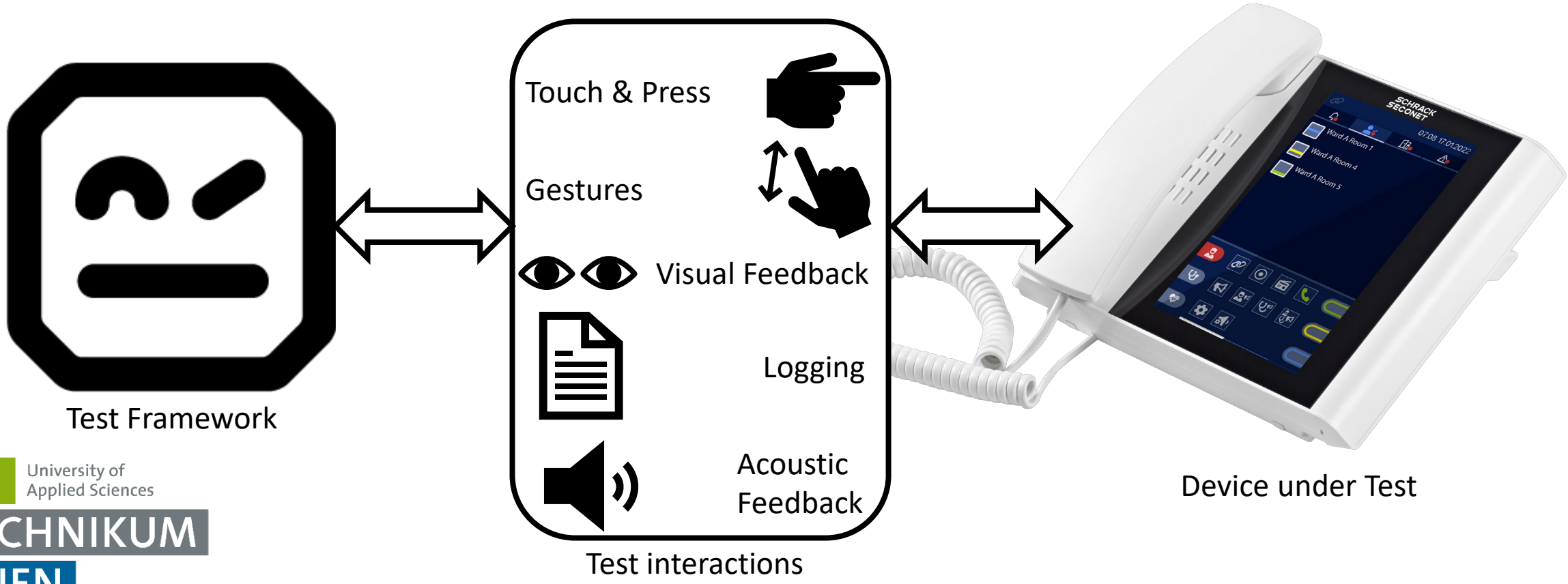
Why am I doing this?

- Reliably working user interfaces are critical in our products
- There is no margin for error when it comes to emergencies

FH University of Applied Sciences TECHNIKUM WIEN

What is embedded UI test automation?

- Machine interacting with user interface of another machine
- Simulating human interactions



What are these maintenance problems?

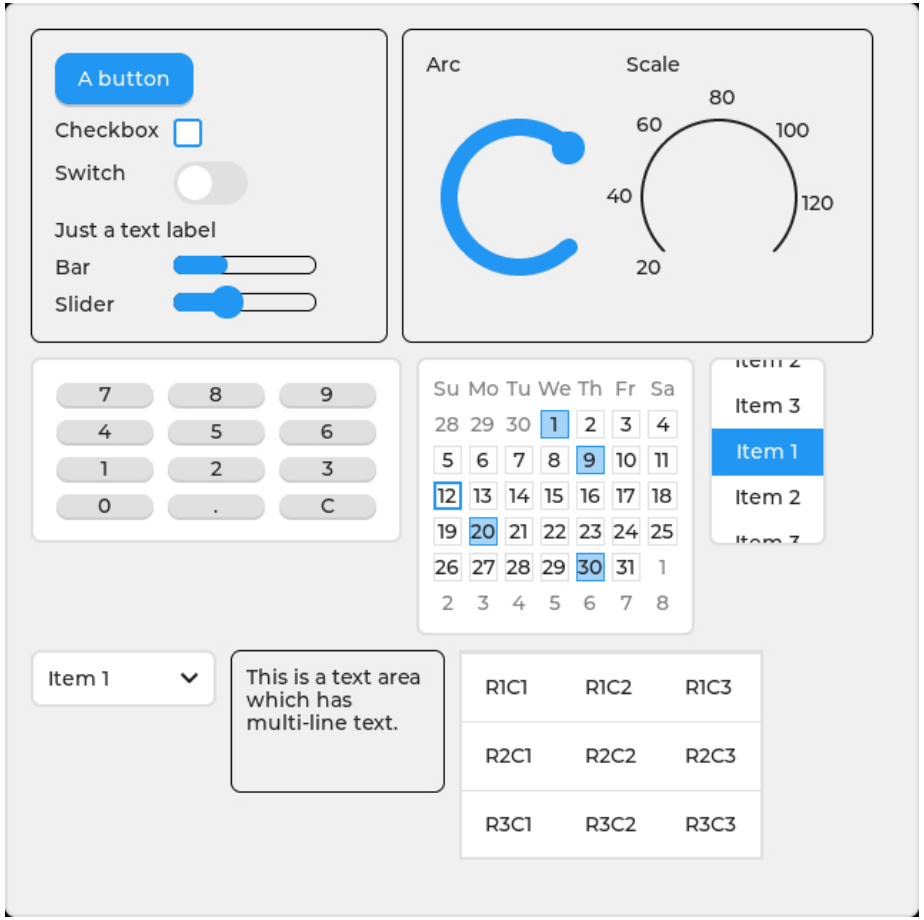
TEST	**Unlock display lock on touch terminal with message expectation**	00:00:02.856
	Full Name: Eval.DeviceControl.Unlock display lock on touch terminal with message expectation	
	Tags: display-unlock, testdev-06	
	Start / End / Elapsed: 20240121 17:38:47.854 / 20240121 17:38:50.710 / 00:00:02.856	
	Status: PASS	
SETUP	**Single device setup** ${StTouchXYZ}, 10.64.7.93, TouchTerminal	00:00:00.055
KEYWORD	VCIP. **SET FILTER MODE ON DEVICE** ${StTouchXYZ}, Allow, Ignore	00:00:00.008
KEYWORD	VCIP. **FILTER TAG ON DEVICE** ${StTouchXYZ}, hid	00:00:00.008
KEYWORD	VCIP. **START LOGGING ON DEVICE** ${StTouchXYZ}	00:00:00.121
KEYWORD	VCIP. **TOUCH DRAG ON DEVICE** ${StTouchXYZ}, 441, 350, 191, 707, 1000	00:00:02.625
KEYWORD	VCIP. **EXPECT MESSAGE IN DEVICE LOG** ${StTouchXYZ}, X:191 Y:707	00:00:00.013
KEYWORD	VCIP. **STOP LOGGING ON DEVICE** ${StTouchXYZ}	00:00:00.009
TEARDOWN	**Single device Teardown** ${StTouchXYZ}	00:00:00.010

Our UI interactions require X,Y screen coordinates for input simulation

FH University of Applied Sciences TECHNIKUM WIEN

Project goals

- Address maintenance and visual challenges in UI test automation
- Automate detection of **widgets** from UI screenshots

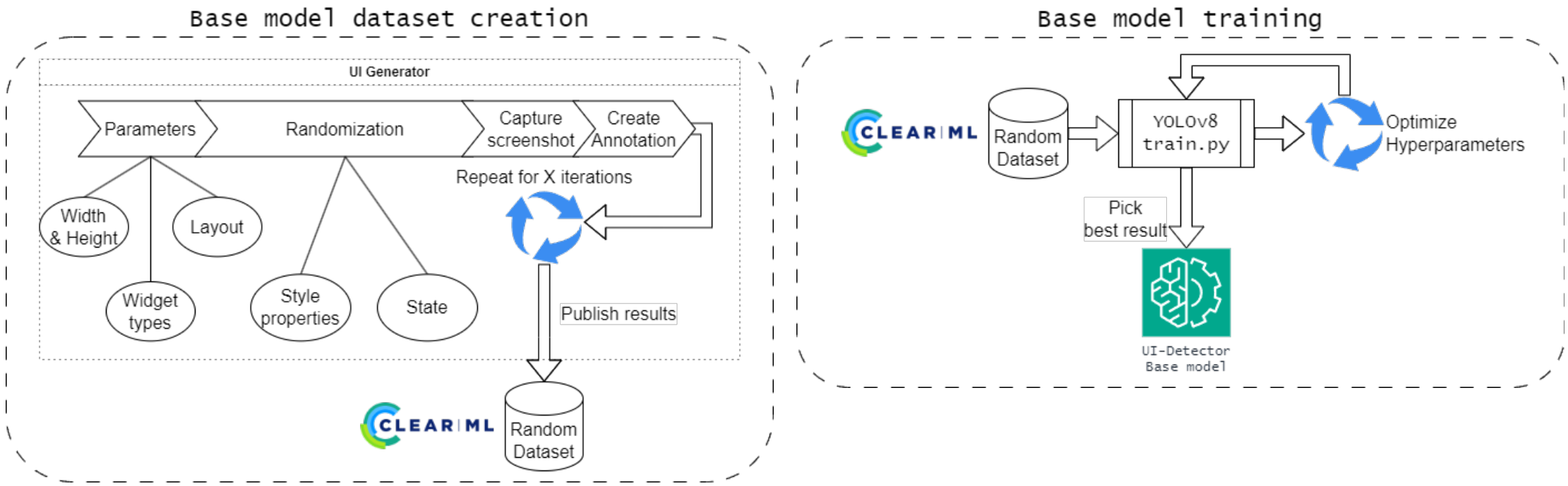


FH University of Applied Sciences TECHNIKUM WIEN

Project overview

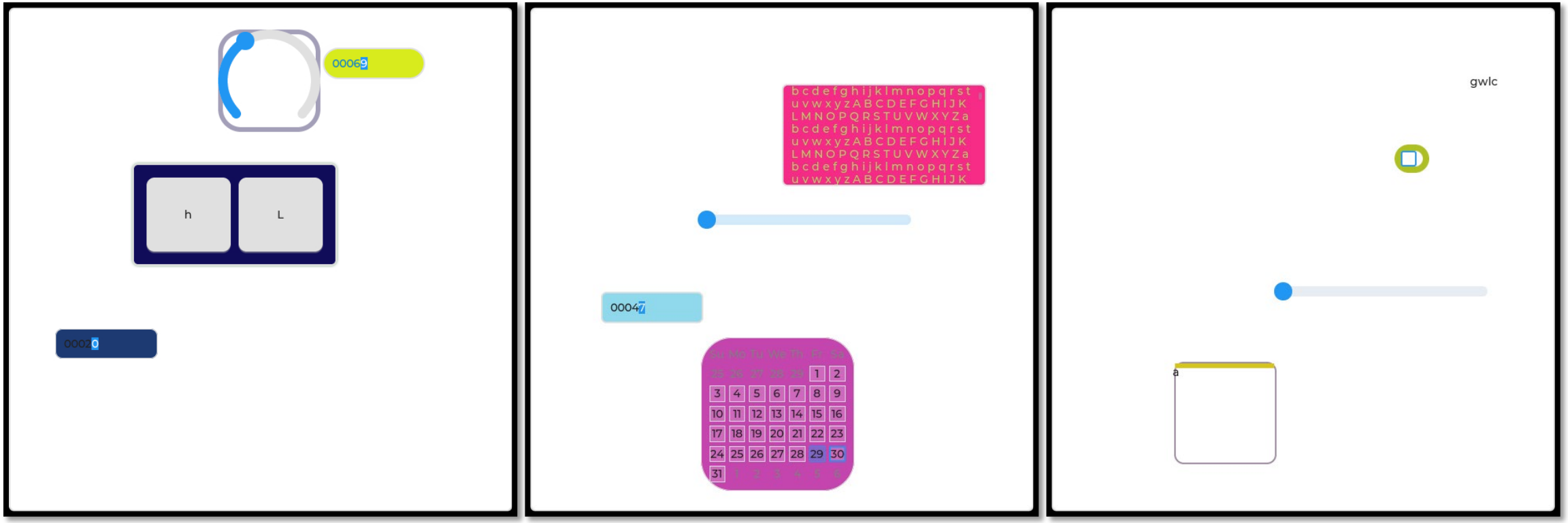
- User interface generator using LVGL
  - … with random widgets & styles
  - … with design specification
- Base model trained on random datasets
- Improved base model trained on design specification
- YOLOv8 for object detection
- ClearML for overall experiment & dataset tracking

Training a base model



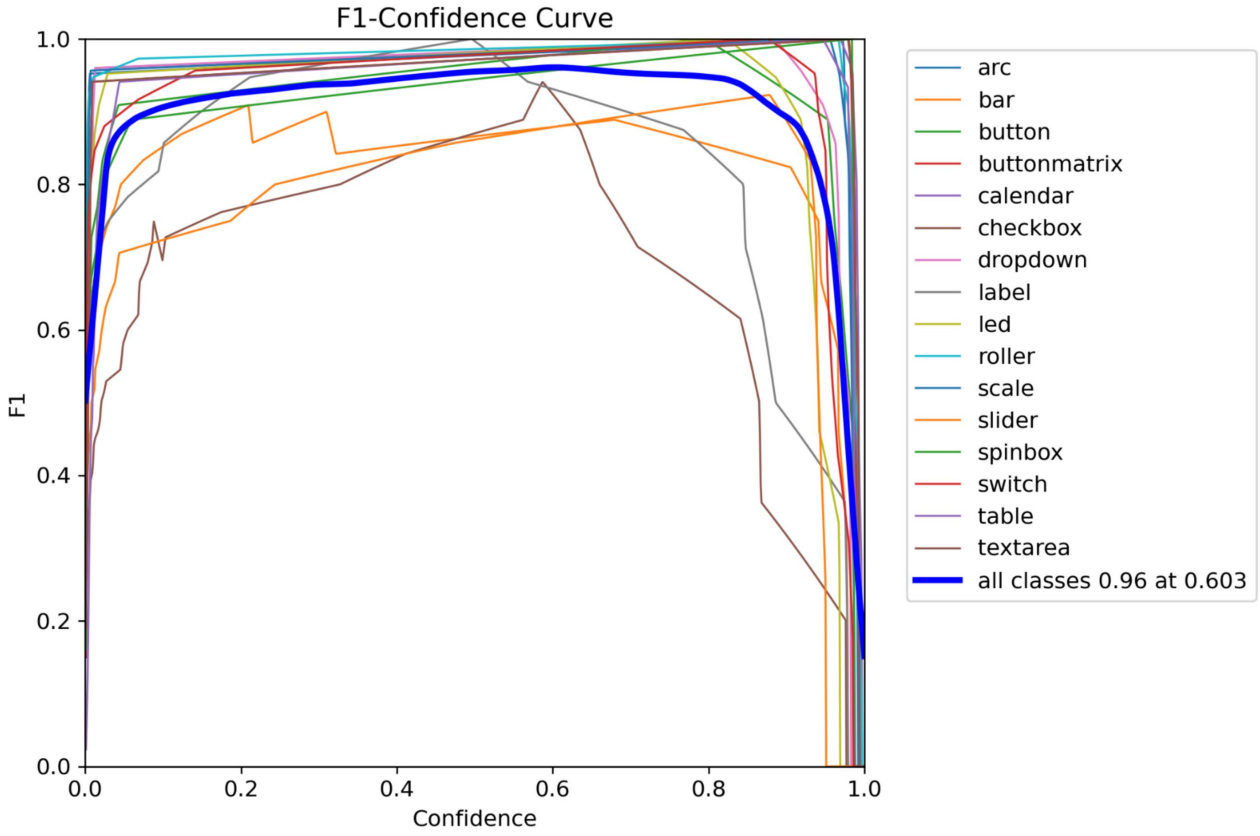
FH University of Applied Sciences TECHNIKUM WIEN

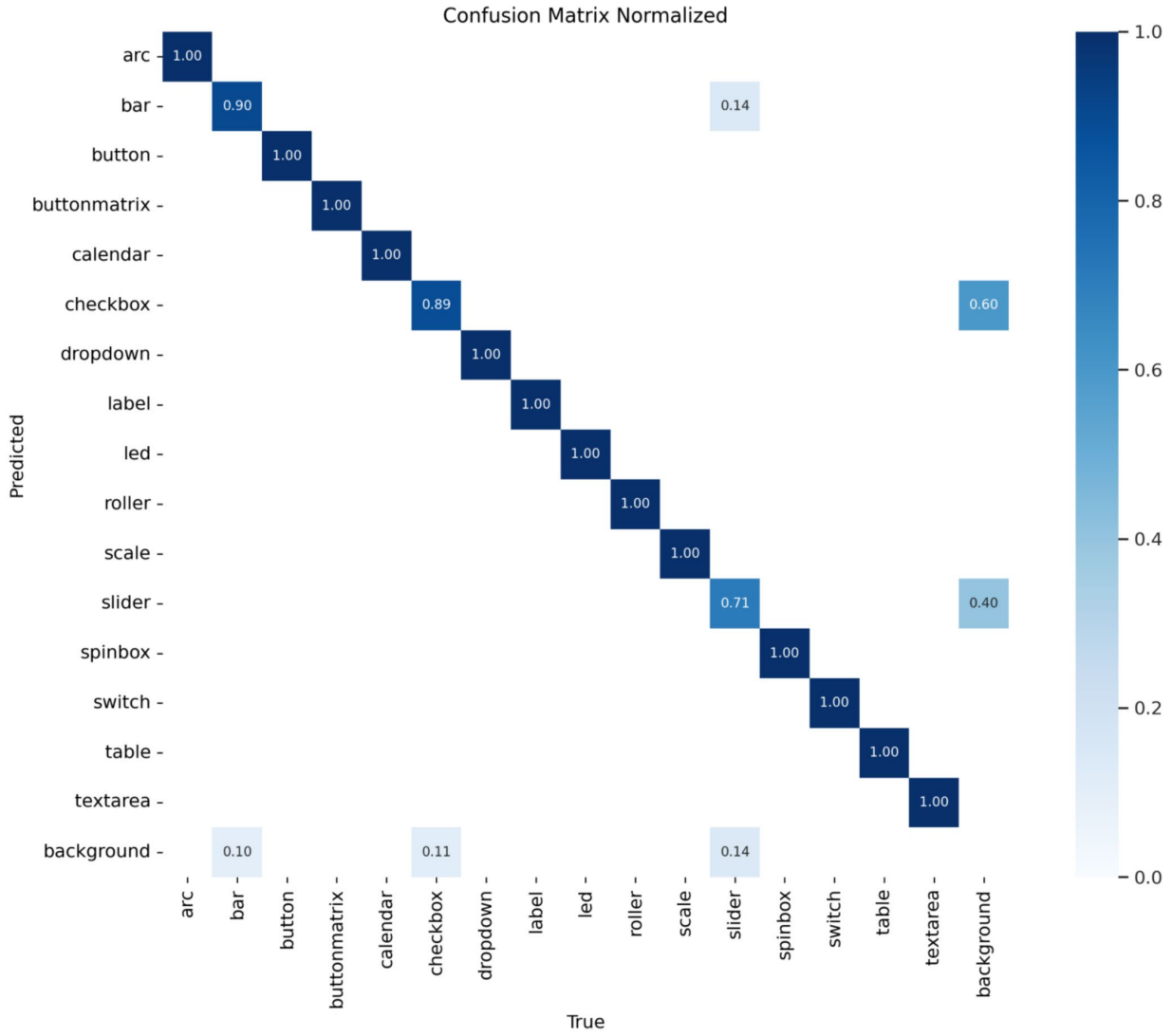
Examples from random widget dataset



FH University of Applied Sciences TECHNIKUM WIEN

Base model: Optimized Results



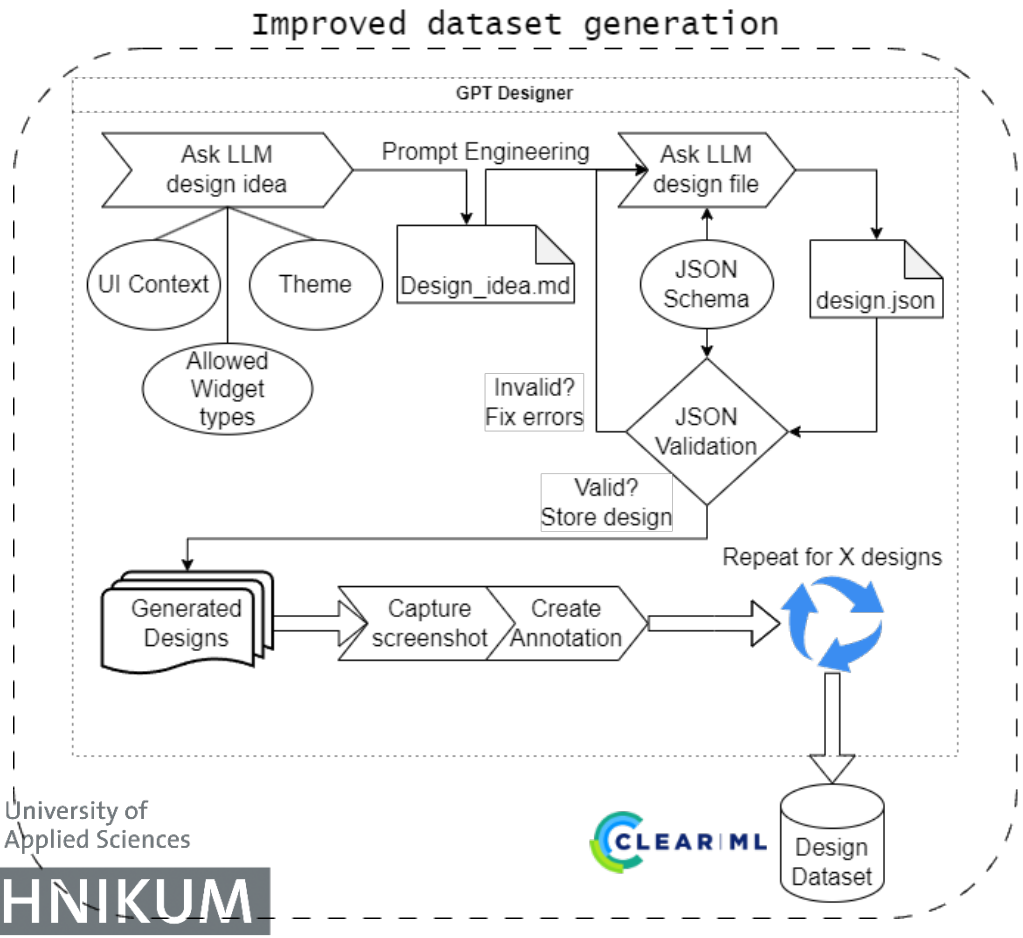


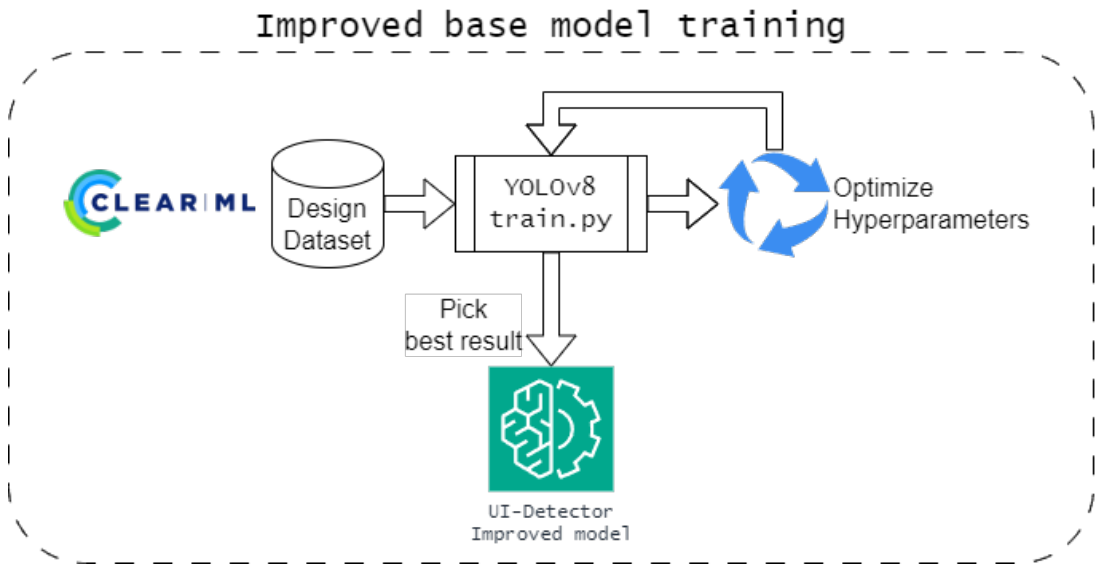
FH University of Applied Sciences TECHNIKUM WIEN

Problems with random widget creation

- Only good on paper, optimized best-case scenario
- Bad model performance with realistic UI screenshots is anticipated
  - *Bias due to white background and styles properties*
  - *Individual widgets “pop-out” from background*
  - *Widgets are absolutely positioned with distinctive distances*

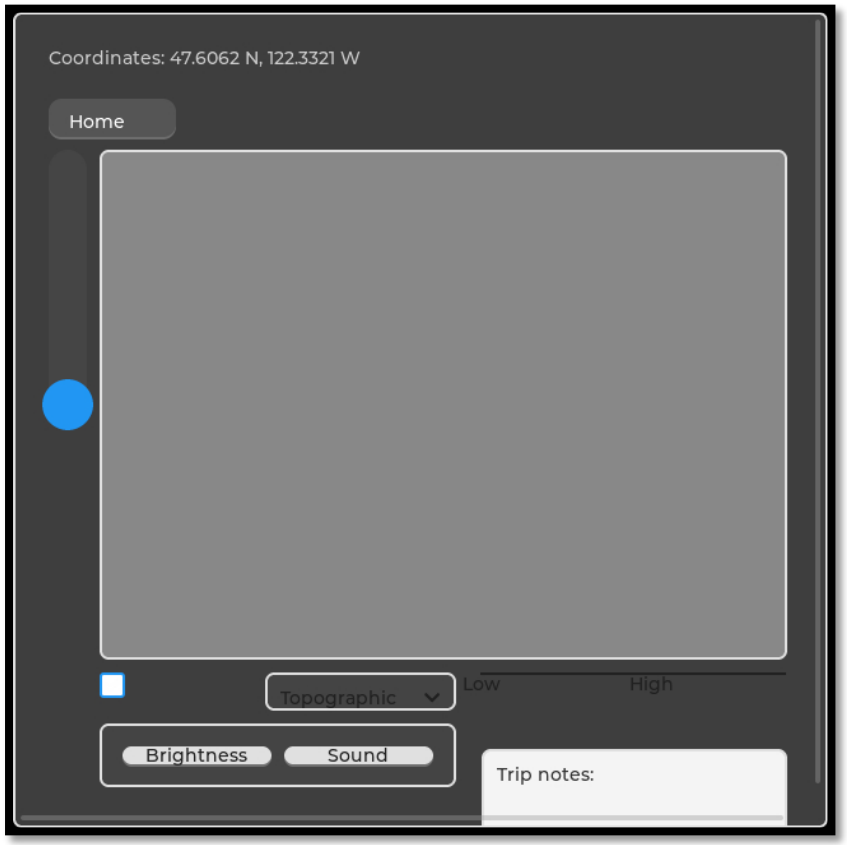
Improved base model

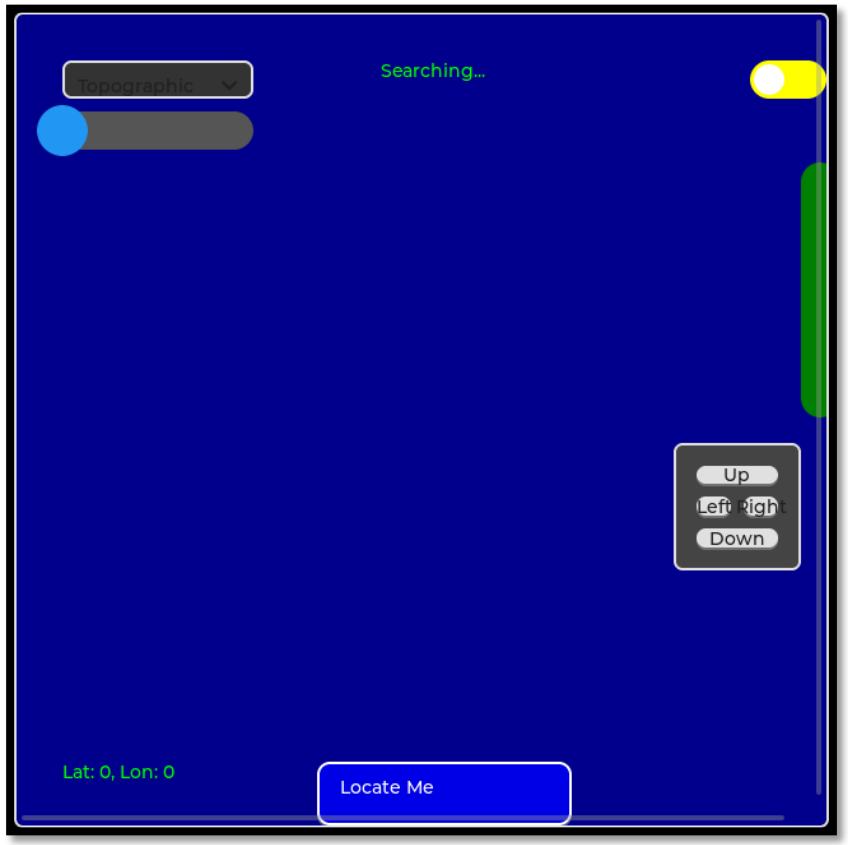




FH University of Applied Sciences TECHNIKUM WIEN

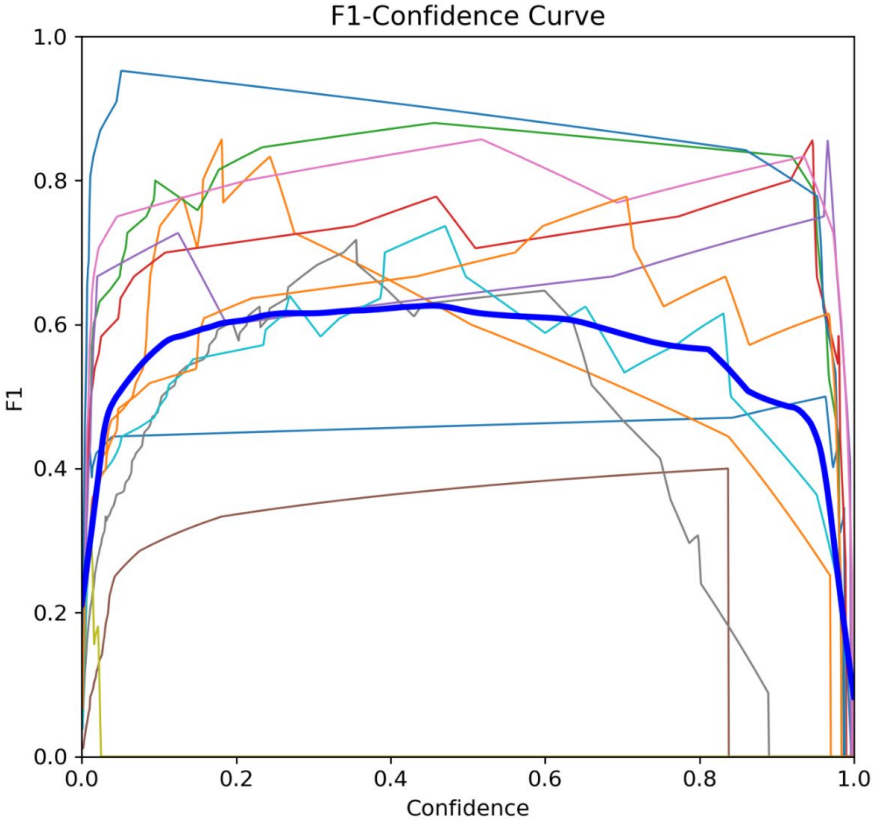
Examples from design widget dataset

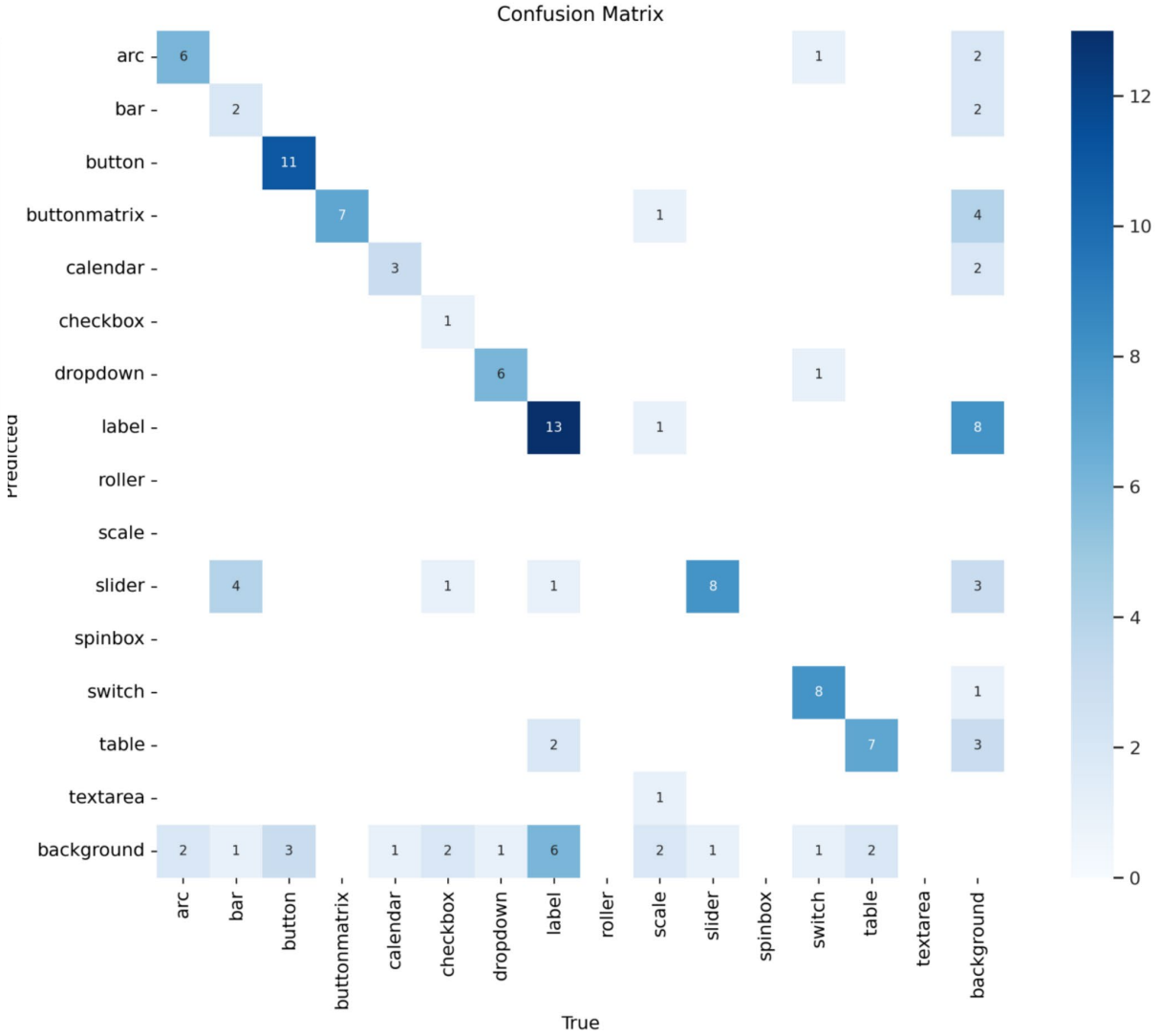




FH University of Applied Sciences TECHNIKUM WIEN

Improved base model: Tune Results





FH University of Applied Sciences TECHNIKUM WIEN

Improved base model: Train Results

Class	Images	Instances	Box(P	R	mAP50	mAP50-95): 100%| | 1/1 [00:00<00:00, 7.83it/s]
all	13	107	0.768	0.624	0.686	0.606
arc	13	8	0.422	0.5	0.393	0.364
bar	13	7	1	0.469	0.924	0.822
button	13	14	0.994	0.786	0.861	0.706
buttonmatrix	13	7	0.627	1	0.883	0.858
calendar	13	4	0.55	0.75	0.888	0.849
checkbox	13	4	0.697	0.25	0.249	0.0991
dropdown	13	7	0.83	0.857	0.837	0.762
label	13	22	0.835	0.5	0.647	0.412
scale	13	5	1	0	0.181	0.121
slider	13	9	0.675	0.778	0.734	0.67
switch	13	11	1	0.823	0.908	0.908
table	13	9	0.586	0.778	0.726	0.703

Best results after 150 epochs.

FH University of Applied Sciences TECHNIKUM WIEN

Final conclusions

Thoughts gathered over roughly ~450 experiments

- Synthesizing datasets for UI object detection is very error-prone
- A targeted dataset for very specific widget detection is possible
- Datasets must contain varying styles and states of widgets for realistic performance
- Styles have a significant impact on widget detection

Future work

- Designing a targeted dataset for our critical user interfaces

FH University of Applied Sciences TECHNIKUM WIEN

Q & A

Thank you for the attention!
Feel free to ask any questions.

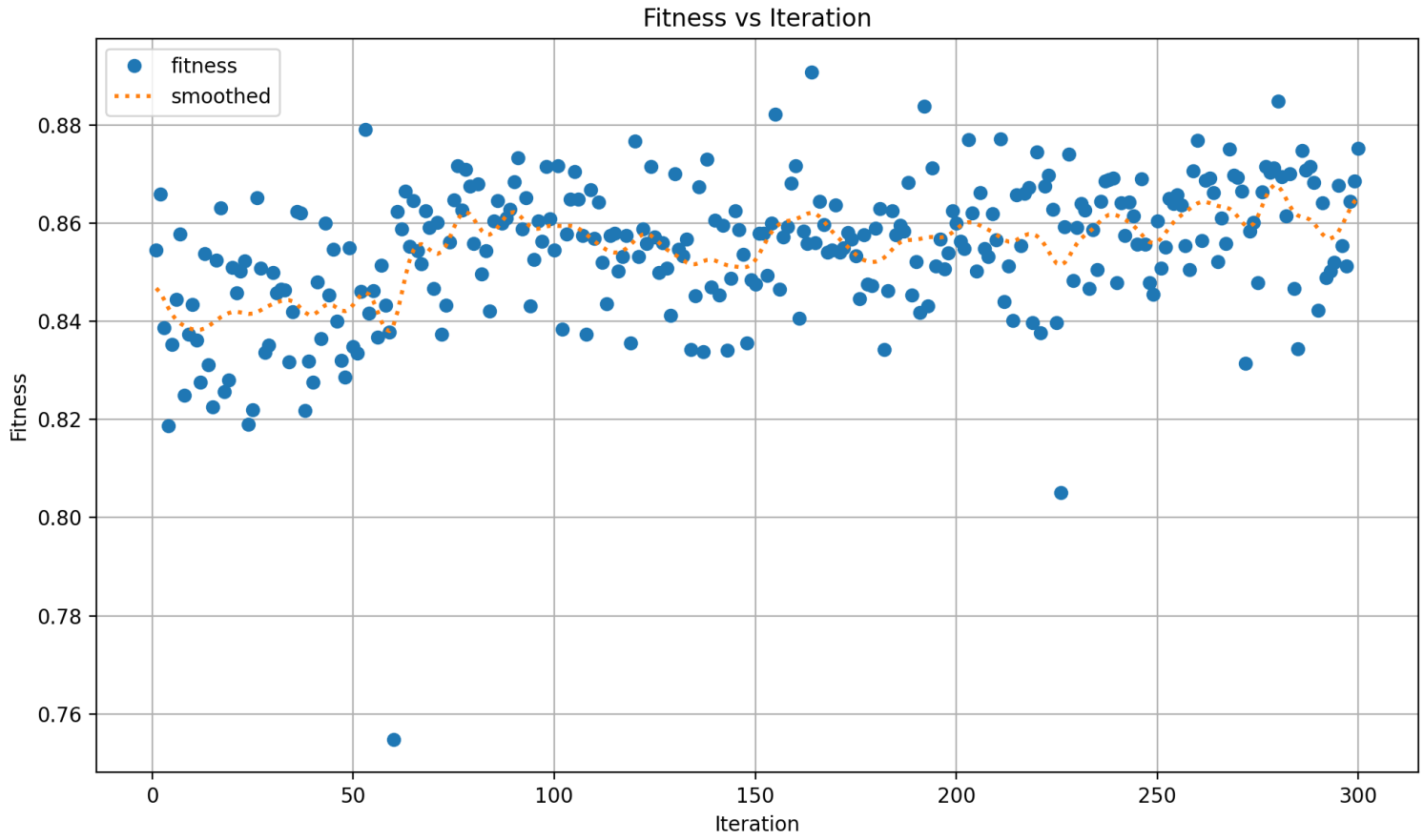
FH University of Applied Sciences TECHNIKUM WIEN

BACKUP SLIDES

Further visualizations and excluded content from the presentation

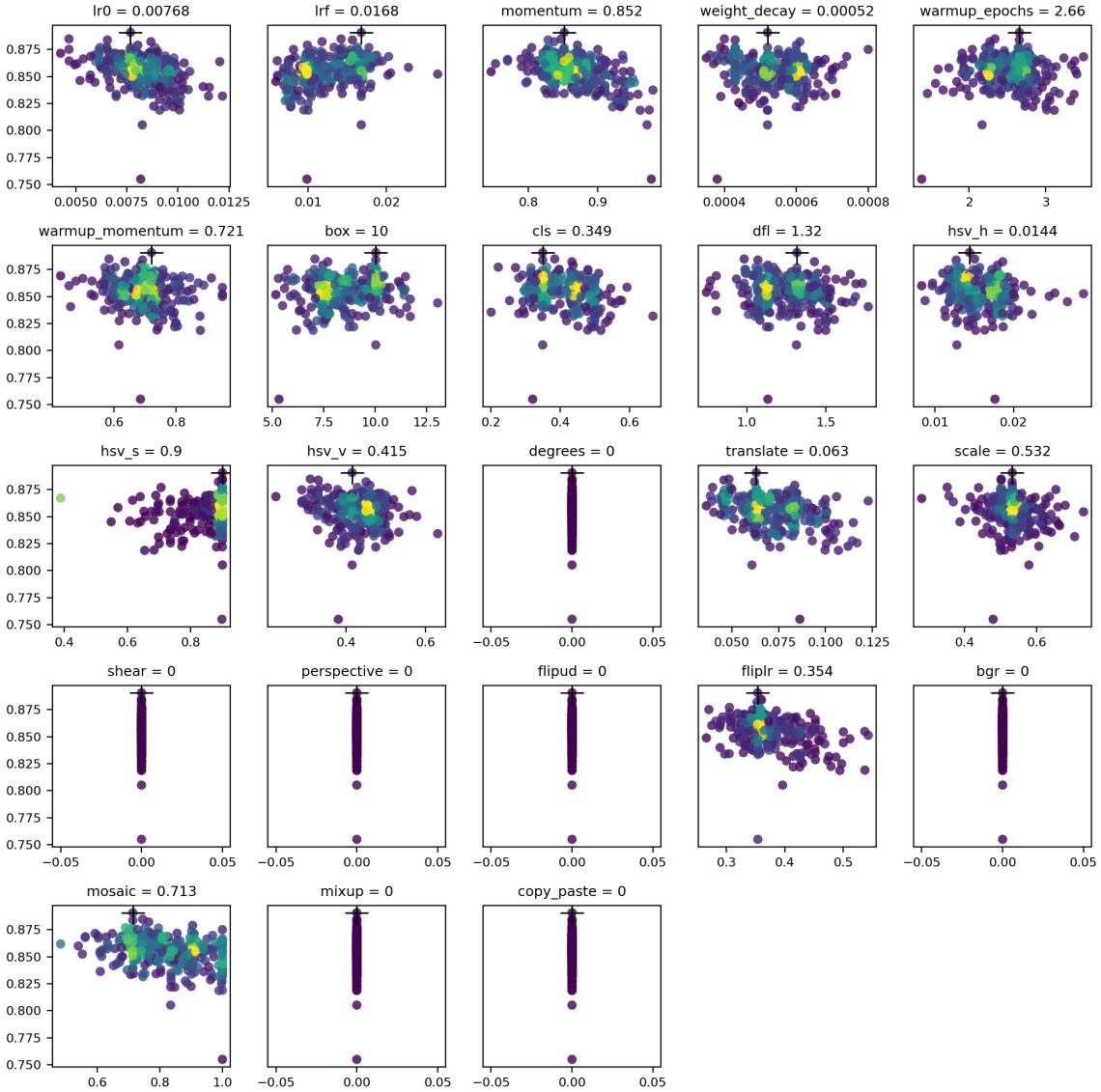
FH University of Applied Sciences TECHNIKUM WIEN

YOLOv8 Model tuning Random (1)



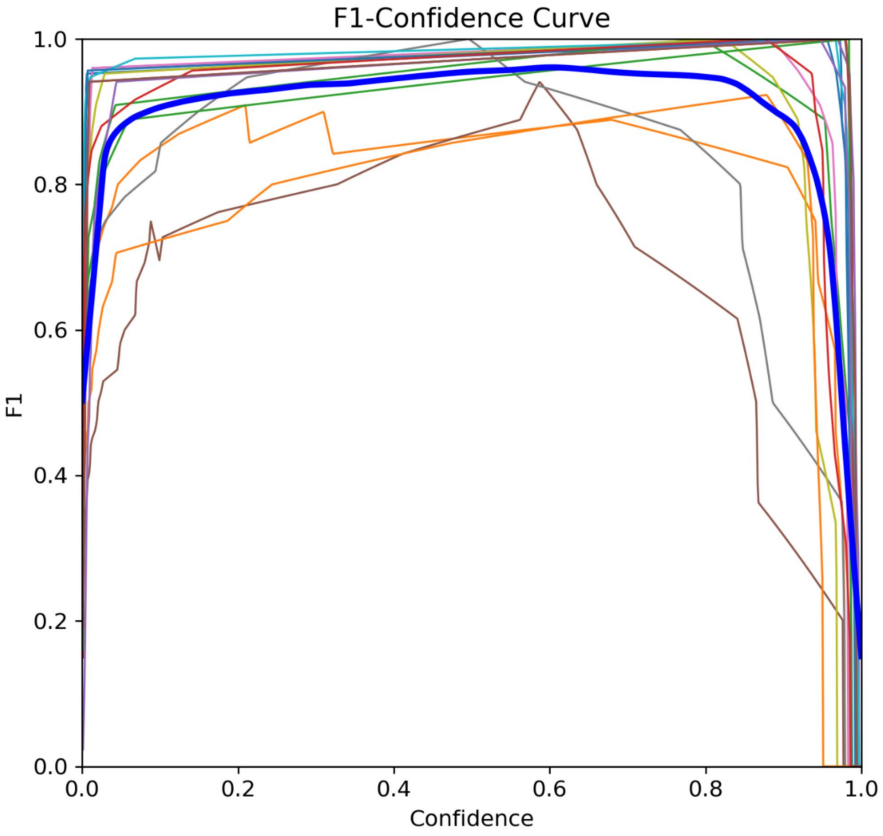
FH University of Applied Sciences TECHNIKUM WIEN

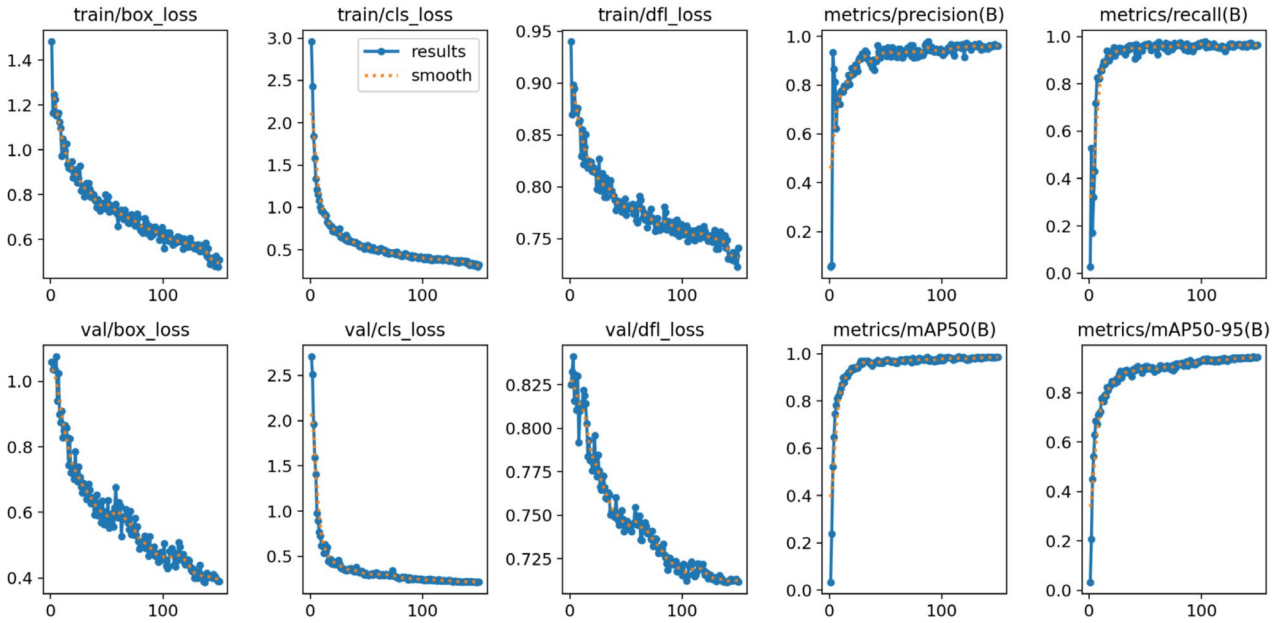
YOLOv8 Model tuning Random (2)



FH University of Applied Sciences TECHNIKUM WIEN

Base model: Optimized Results

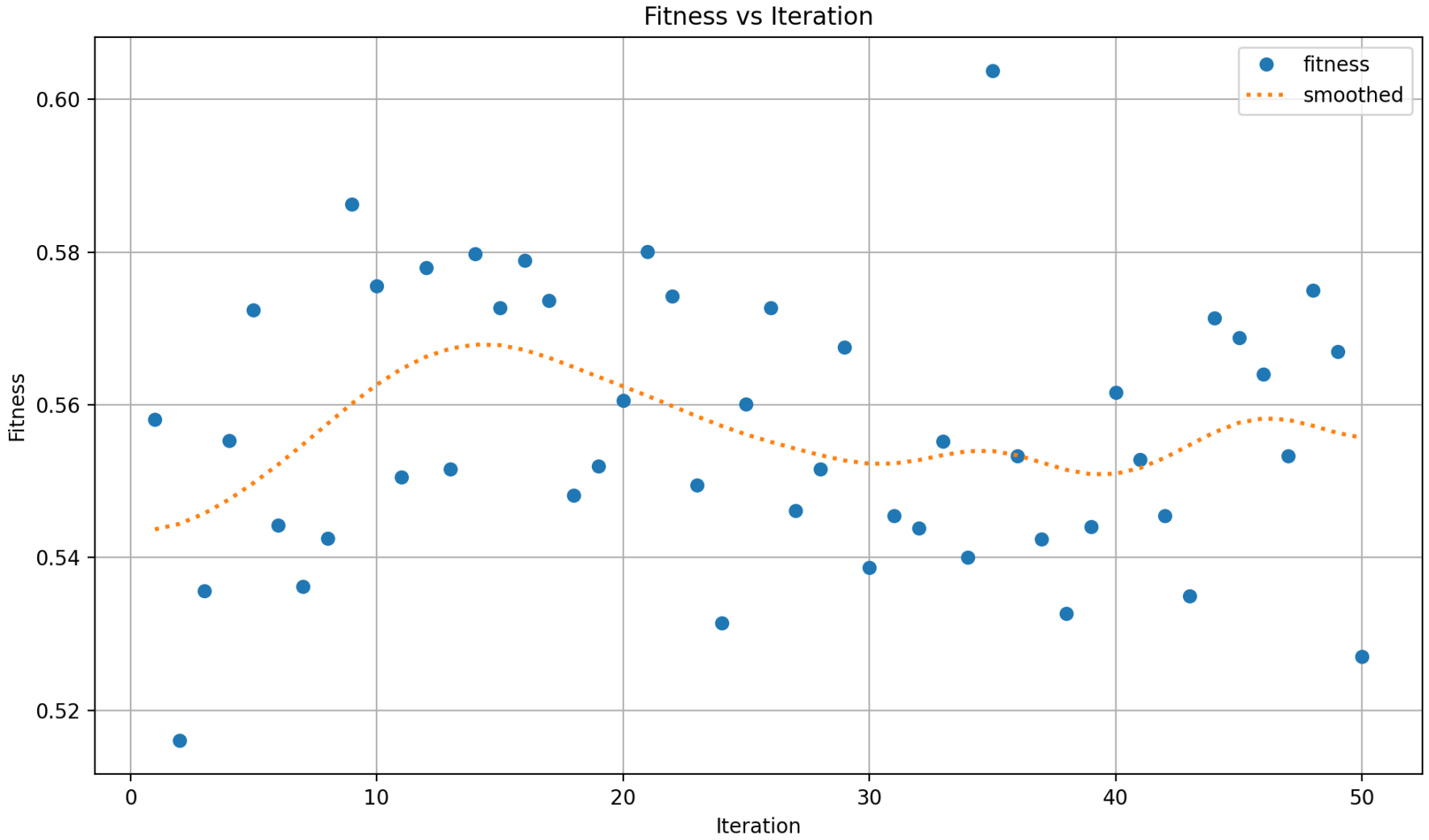




Best training result:
150 epochs
mAP50-95: 0.9449

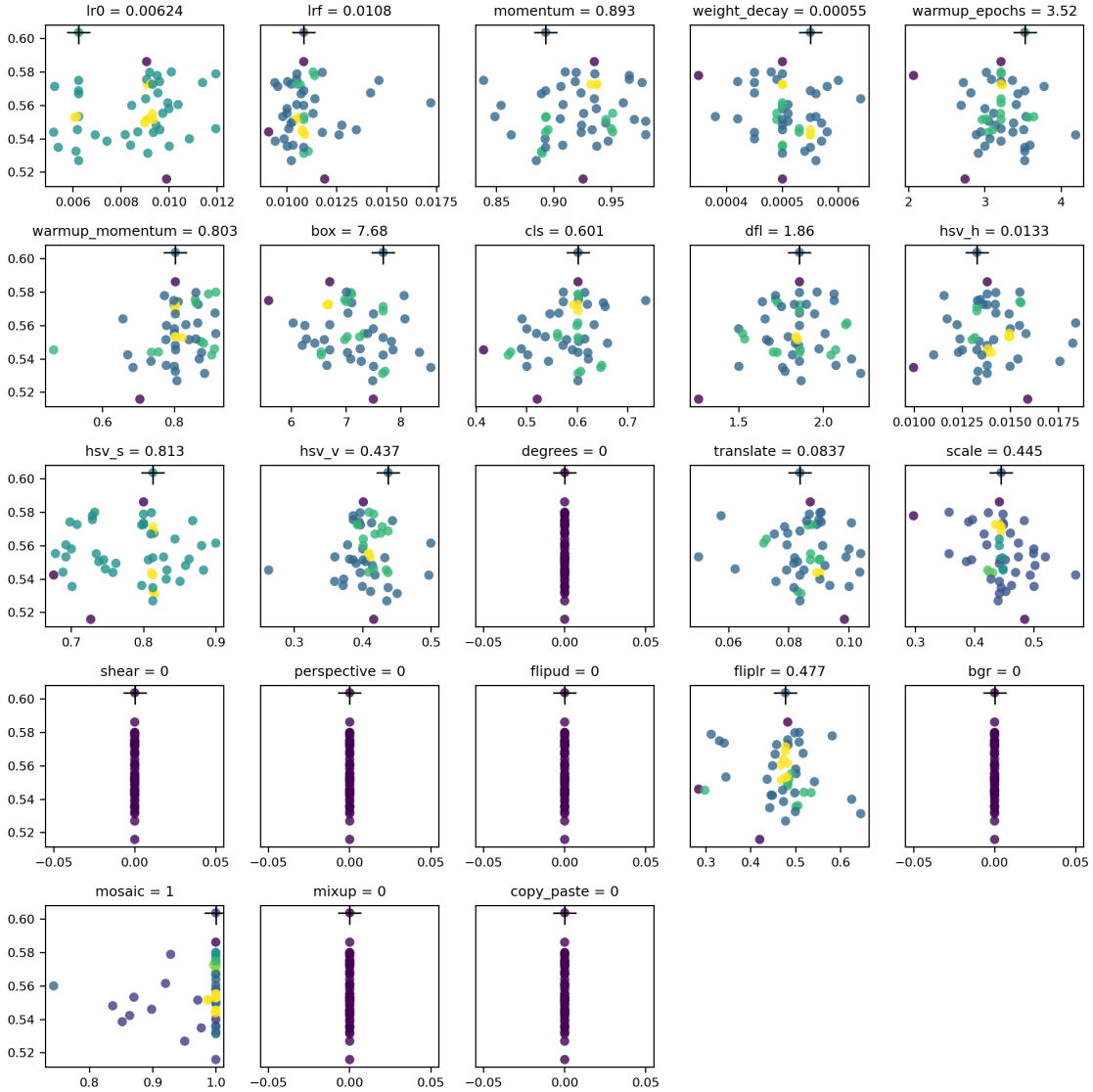
FH University of Applied Sciences TECHNIKUM WIEN

YOLOv8 Model tuning Design (1)



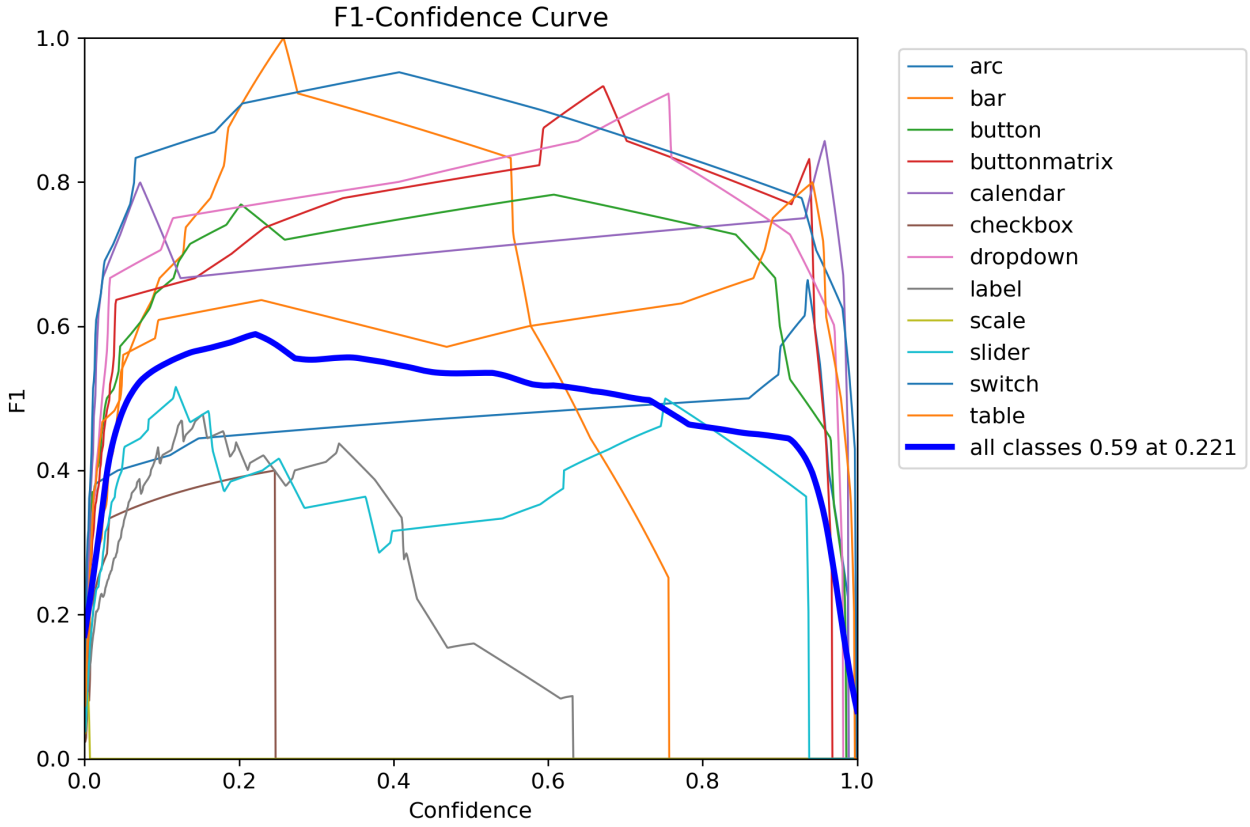
FH University of Applied Sciences TECHNIKUM WIEN

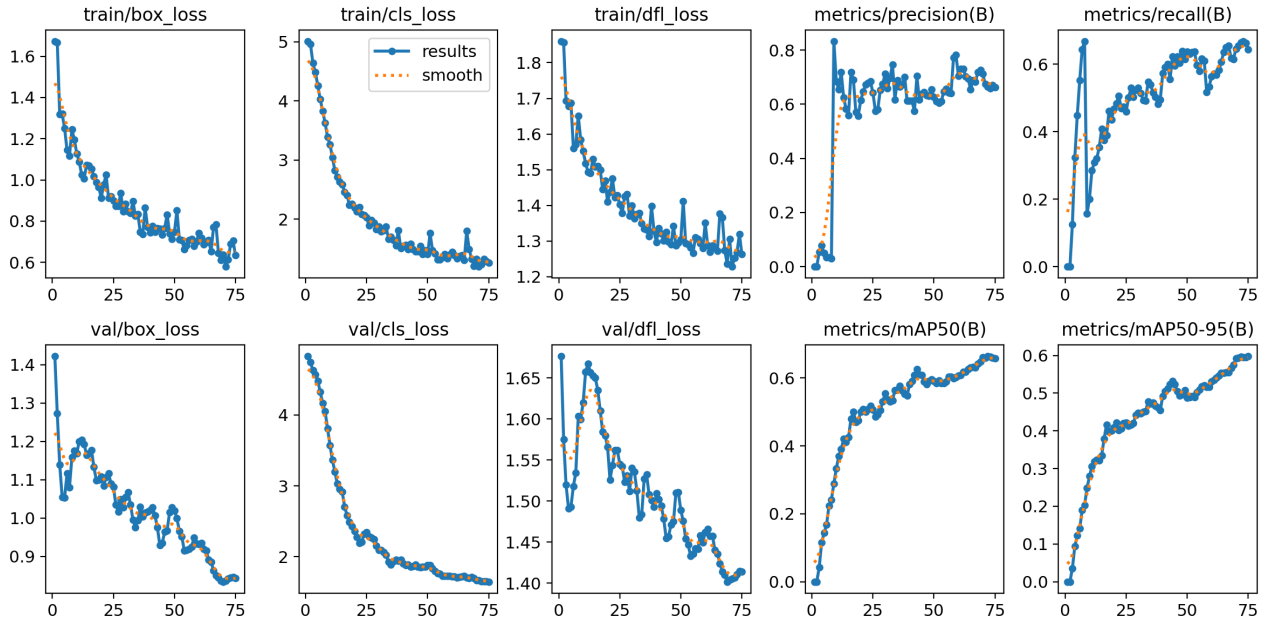
YOLOv8 Model tuning Design (2)



FH University of Applied Sciences TECHNIKUM WIEN

Improved base model: Tune Results

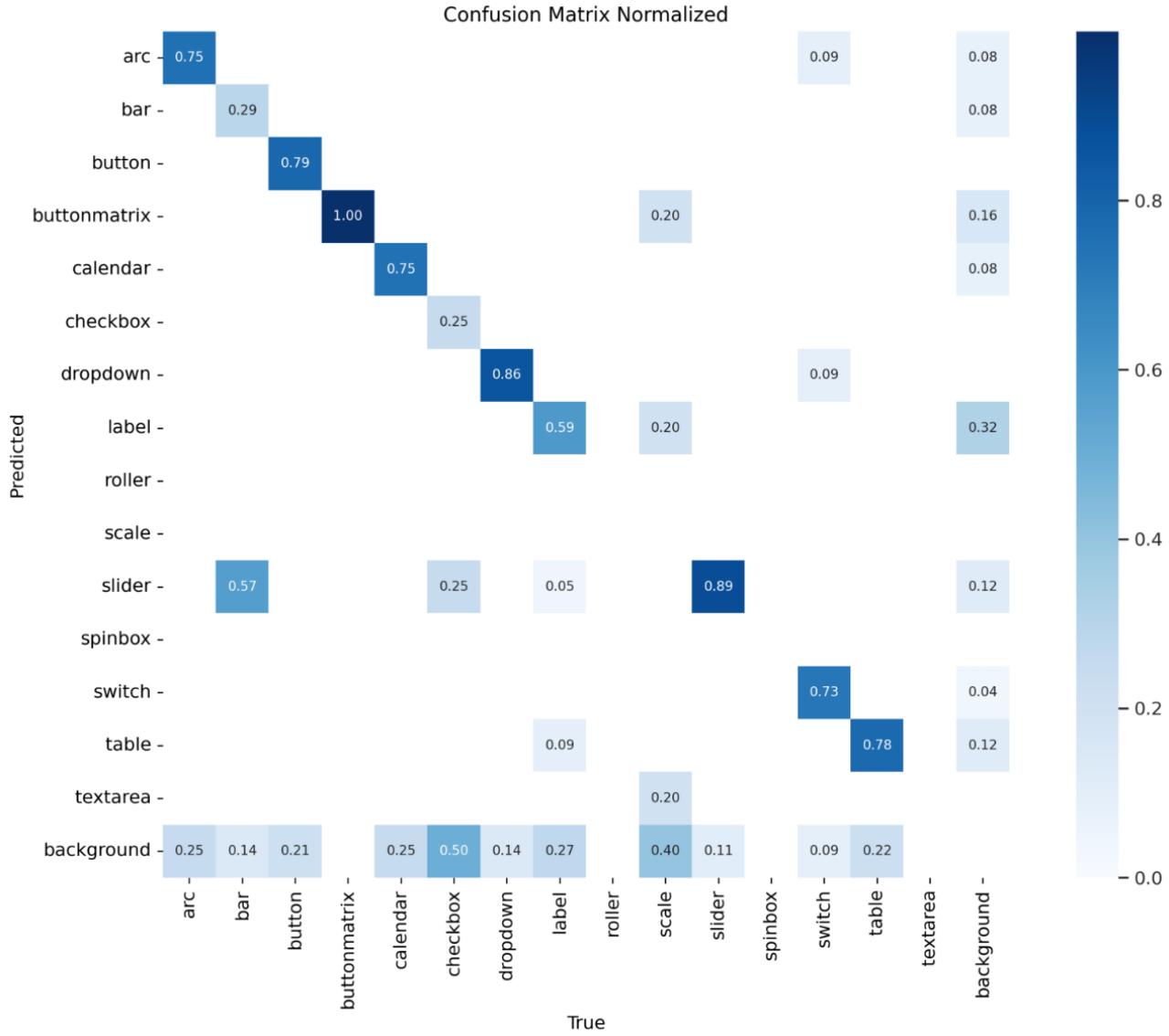




Best tune result of 50 iterations @ 75 epochs
mAP50-95: 0.59786
precision: 0.66274
recall: 0.64346

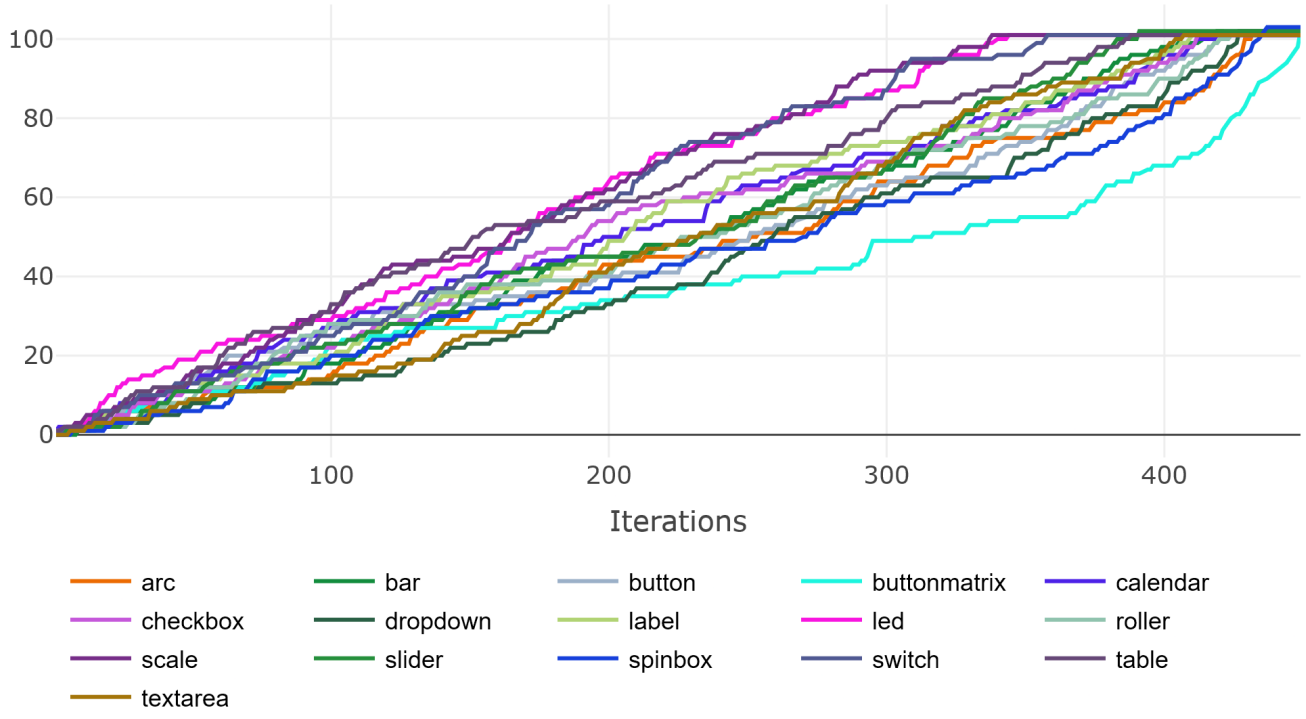
FH University of Applied Sciences TECHNIKUM WIEN

Improved base model: Train results (normalized)



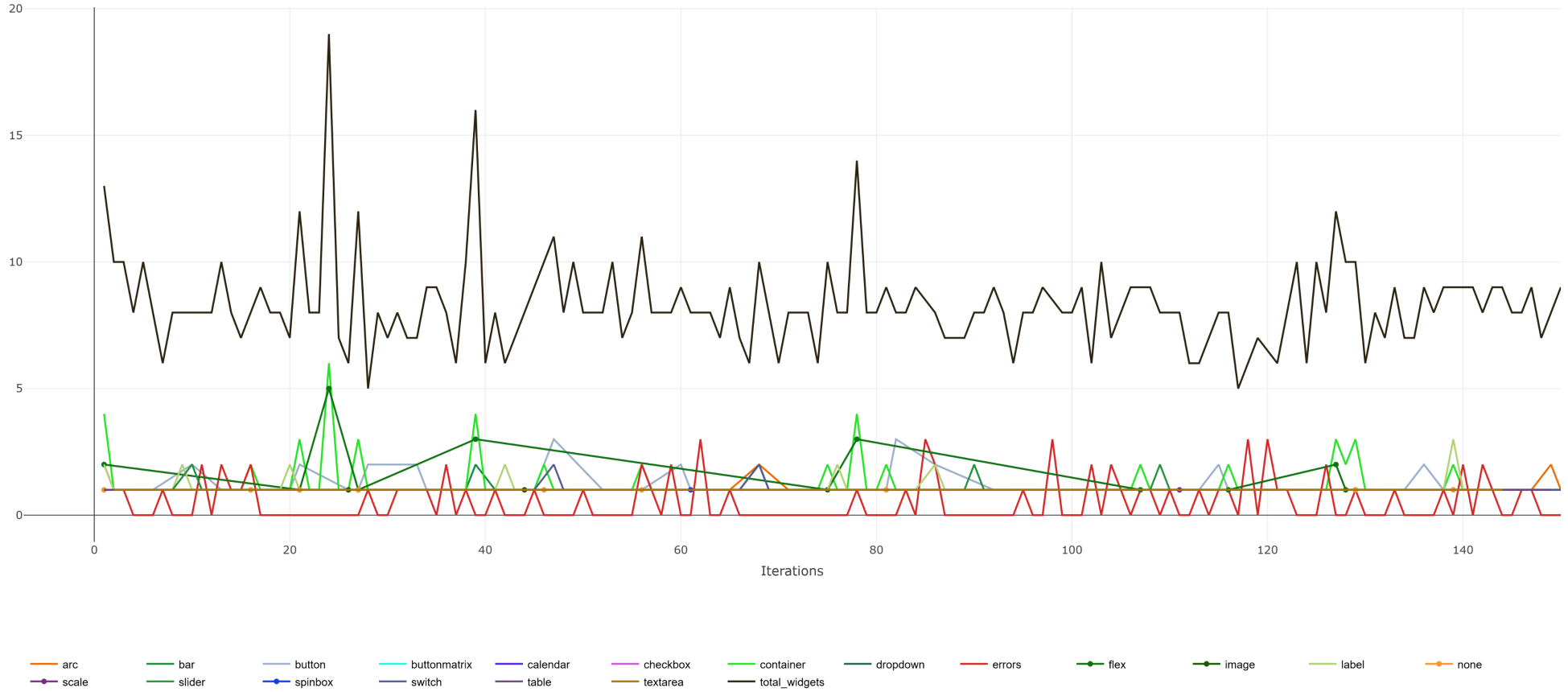
FH University of Applied Sciences TECHNIKUM WIEN

Random dataset creation (100 per widget)



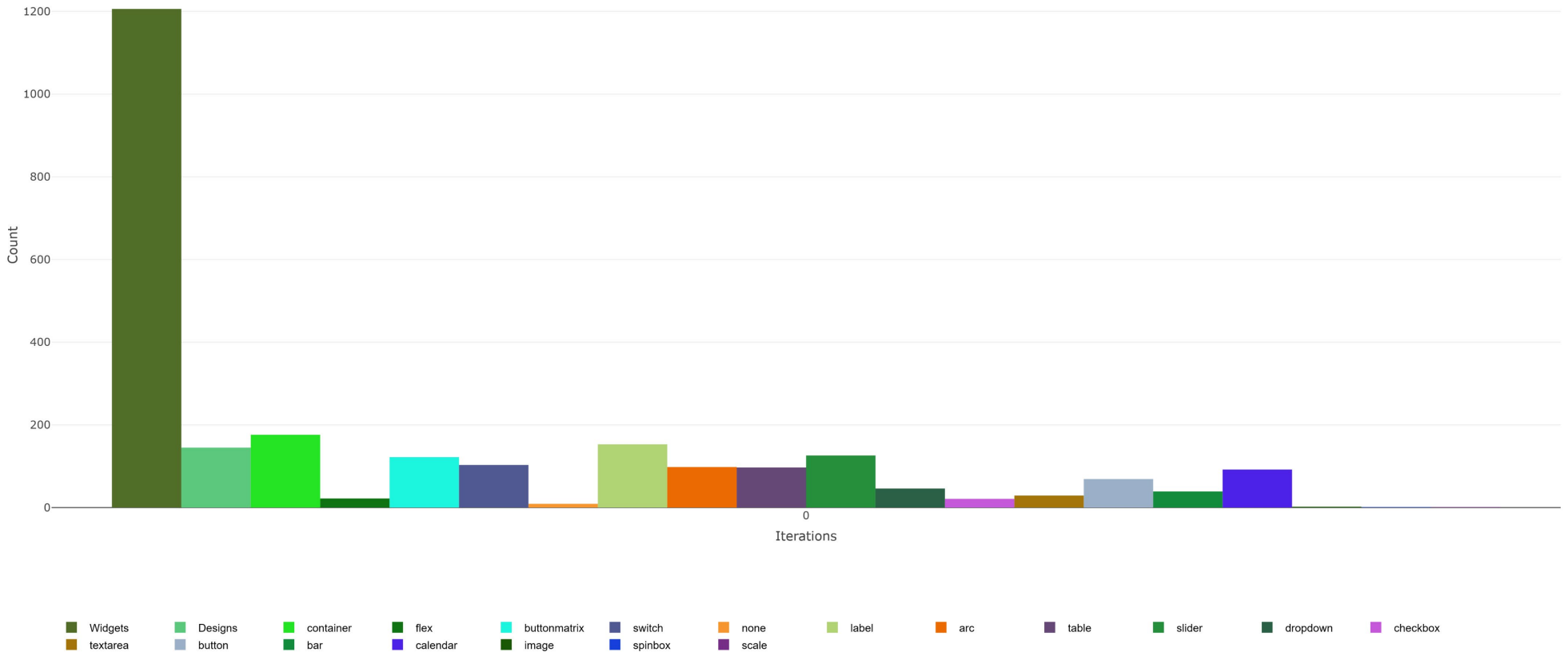
FH University of Applied Sciences TECHNIKUM WIEN

Design dataset creation (150 iterations)



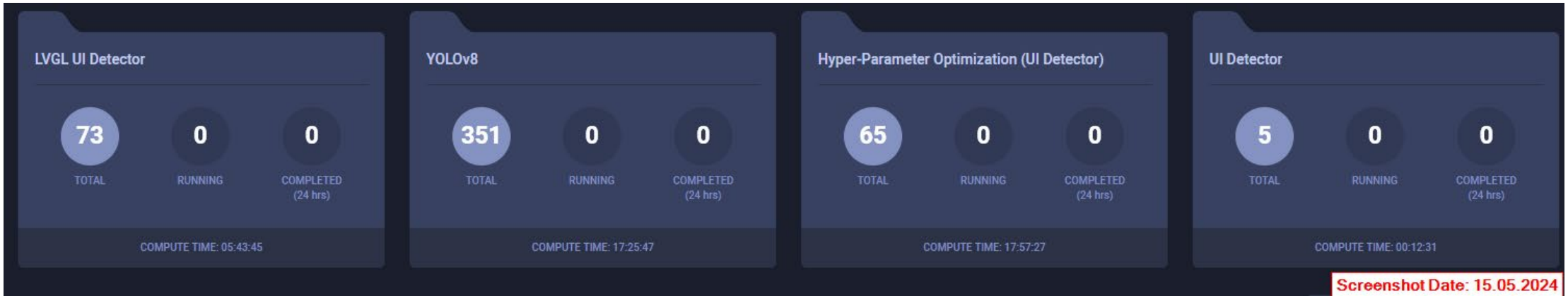
FH University of Applied Sciences TECHNIKUM WIEN

Design dataset creation (150 iterations)



FH University of Applied Sciences TECHNIKUM WIEN

ClearML experiment tracking



FH University of Applied Sciences TECHNIKUM WIEN

ClearML dataset versioning