

CoasterDex

by David Pažout and Dagur Elinór Kristinnson

Achievements

from first presentation

- success
 - determining if a coaster is present in a dataset and retrieve the match, otherwise add the coaster to the collection
 - matching a coaster from a complex scene (varied lighting and angles)
 - coaster recognition in a scene
 - custom dataset
- somewhat
 - transforming the matched coaster to a frontal view
 - retrieval from video at framerate
- failed
 - extract additional data

Achievements

from last week presentation

- success
 - combine modules for full functionality
 - different search for faster coaster addition
- somewhat
 - async call implementation
- failed
 - Color dependant features/recognition

Background

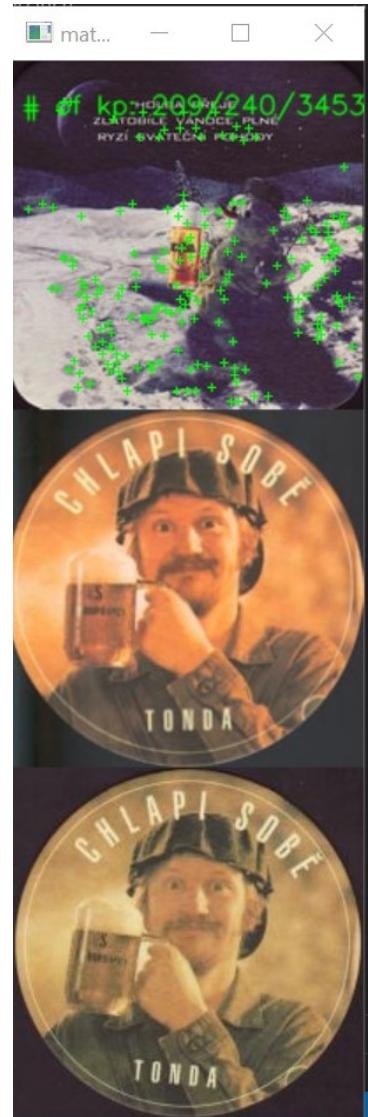
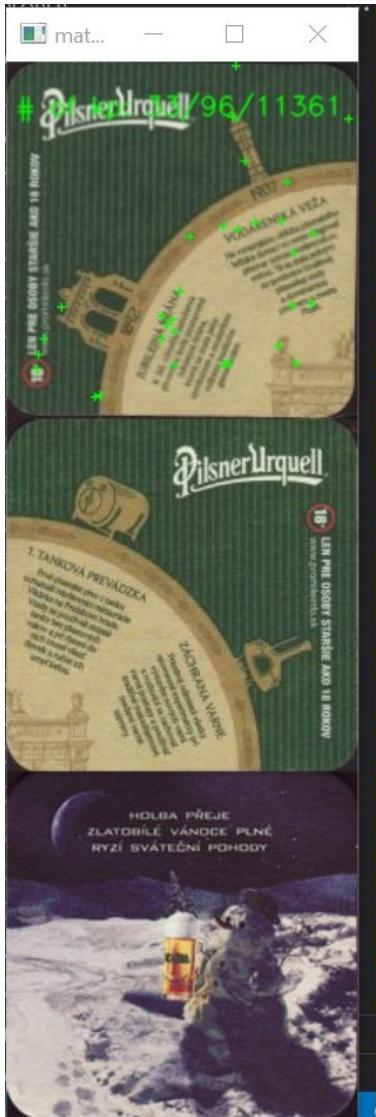
- instance recognition overviews [1], [2]
- website with coaster scans [3]
- custom dataset - photos of printed coaster scans [4]

Instance recognition - methods

- used methods:
 - Contrast Limited Adaptive Histogram Equalization (CLAHE)
 - SIFT features, VLAD encoding, BallTree NN search
- other steps
 - key-point matching filter
 - dynamic database
 - bbox rescaling

Instance recognition - results

	accuracy	FPS
RoorSIFT	0.7578	1.37
SIFT	0.7842	1.37
ORB	0.3578	9.05
SIFT+cls	0.9631	1.48



Coaster detection - Dataset augmentation

Roboflow provided automatic augmentation

Flip, 90°Rotate, Crop, Shear, Grayscale, Hue, Brightness, Exposure, Blur,
Bounding Box: Rotation, Bounding Box: Shear, Bounding Box: Brightness,
Bounding Box: Noise

190 images -> 498 images

Coaster detection - Dataset augmentation

Old Dataset

- 179 Photos of 120 coasters, 5 empty photos

Additions

- 4 coasters with a beer glass nearby
- 2 pictures with faces
- Dataset augmentation

Coaster detection - Results

44 images with 50 coasters not present in training set

Synthetic version:

Recall = 0.96, Precision = 0.83
Average confidence of correct detections = 0.74

Augmented version:

Recall = 0.9, Precision = 0.85
Average confidence of correct detections = 0.78

Detections example V1



Detections example V2



Coaster detection - Finalizing dataset

- Training dataset needs more environment objects, specifically beer for better recall
- Augmentation can be experimented with to be utilized more effectively to improve generalization
- Augmented model might prove superior since both train and test datasets are most likely too specific

Instance Recognition + Coaster detection - Results

accuracy = 0.937

Work involved

- feature matching for instance level recognition
 - parallelization
 - integration of existing code
- model training for coaster detection
 - custom dataset creation
 - model training environment errors took time to resolve
- integration across systems

Conclusion

- would we do differently, if are doing the project over again?
 - better research methodology
 - better recording of the process
 - create a testing framework
 - don't get sick
- future improvements
 - PCA for feature vector size reduction
 - full parallelism
 - proper database (SQL)
 - more advanced testing
 - bigger, more varied dataset
 - full mobile application

Thank you for your attention

Questions?

Sources

- [1] <https://ieeexplore.ieee.org/document/7935507>
- [2] <https://arxiv.org/pdf/2101.11282.pdf>
- [3] <https://www.beer-coasters.eu/cz/pivni-tacky.html>
- [4]
<https://drive.google.com/drive/u/0/folders/1yGVfdQc7eryq0ctGnrMsVCRHP1Rgl8Wu>
- [GitHub] <https://github.com/pazoudav/CoasterDex/tree/main>