CoasterDex

by David Pažout and Dagur Elinór Kristinnson

Motivation

My friend has a large collection of beer coasters (+200).
When I'm in a pub and see a coaster, I don't know if they already have it or not. I could ask them but sometime they don't know either. I don't want to bring them duplicate coaster or bother them all the time (and its nice to surprise them with a new coasters).

Requirements: from basic to nice to have

- determining if a coaster in 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)
- compare multiple approaches to this problem

.

- coaster recognition in a scene
- transforming the marched coaster to a frontal view
- retrieve relevant information (text, shape, maker...)

.

- generalize to other object domains
- retrieval from video at framerate

Problem task

instance recognition: computer vision task of recognizing a specific instance of an object, rather than simply the category to which it belongs [1]
instance retrieval: finding images similar to a provided query from a large database [2]

image classification

Instance recognition/retrieval

- **SIFT based**
  - feature extractor (SIFT, SURF, ...)
  - codebook training (feature clustering)
  - feature encoding/quantization
  - search (ANN, inverse table)
- **CNN based**
  - pertained CNN model -> feature vectors from layer
  - model trained to create a small feature vector

[6]

Relevant works

- postage stamp recognition system [7]
- DELF/DELG: CNN feature extractor [8]
- HotSpotter: individual animal identification [9]
- grocery store product recognition [10]

Differences

- single domain/class with high variability
- only one instance of each item present in the dataset
- compare possible advantages of CNN approach on smaller scale datasets with high demand on time and storage (model and feature representation) performance

Methods

Some level of machine learning (e.g. YOLO) would be required to locate the object. We would like to avoid having it do the comparison due to the inherent need for dynamic collection management.
This makes it stand out

Performance

We would like to minimize the time it takes for the person photographing the object to receive a result.
Object tracking time and image comparison time.

Precision for recognition/retrieval

Datasets

- website of a personal beer coaster collection: https://www.beer-coasters.eu/cz/pivni-tacky.html [3]

[4]

- collecting photo in real environment, for example:

[5]

- manufactured dataset from printed paper coasters

Thank you for your attention

Sources

[1] https://blog.research.google/2020/09/advancing-instance-level-recognition.html

[2] https://paperswithcode.com/task/image-retrieval

[3] https://www.beer-coasters.eu/cz/pivni-tacky.html

[4] https://www.beer-coasters.eu/coasters/branik-10.jpg

[5] https://g.denik.cz/54/45/20151127-pivo-tacek-osek_denik-galerie-800@2x.jpg

[6] https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7935507

[7] https://www.diva-portal.org/smash/get/diva2:422669/FULLTEXT01.pdf

[8] https://github.com/tensorflow/models/tree/master/research/delf

[9] https://ieeexplore.ieee.org/abstract/document/6475023

[10] https://dl.acm.org/doi/pdf/10.1145/3240508.3240522