

EE 428: Computer Vision

Homework 2: Book Search

Instructor: Jonathan Ventura

In this homework you will implement an image-based search system to find a book in a database given a photograph of the book cover.

The included `Childrens-Books.zip` archive contains almost 1000 images of book covers. Using the techniques described in class, you will build a visual vocabulary on these images, convert each image into a text document, and set up a document retrieval model.

Then you will test a set of query images (`queries.zip`) and evaluate how well the system is able to recognize the books in the images.

Note: it is okay to do this homework in a Python notebook, if you want to. It might be helpful since some of the steps like SIFT feature extraction and K-Means clustering are slow, so you might not want to run them over and over again as you develop your solution.

To install the dependencies for this homework:

```
pip install scikit-image scikit-learn
```

Code requirements

Build the visual vocabulary

1. Load each image from `Childrens-Books` and convert to grayscale. Here is example code to get a list of paths for the book images:

```
paths = sorted(glob.glob('Childrens-Books/*.jpg'))
```

2. Use `skimage.feature.SIFT` to extract a list of SIFT descriptors for each image.
3. Use `sklearn.cluster.KMeans` to cluster all of the SIFT descriptors (from all of the images) into 1000 clusters.

Convert images to documents

4. Use the KMeans clustering to convert each list of SIFT descriptors into a list of cluster labels (using `.predict()`).
5. Convert each list of cluster labels into a string (using `str()`). Now each image has been converted into a "text document."
6. Use `sklearn.feature_extraction.text.TfidfVectorizer` to convert each "document" to a tf-idf weighted histogram.
7. Build a k-Nearest Neighbors model (`sklearn.neighbors.NearestNeighbors`) on the histograms. Use the `cosine` metric.

Test image search

8. Convert each image in `queries` to a document using the structures you built above.
9. Find the five nearest neighbors in the dataset for each query image.
10. Plot each query image next to its five nearest neighbor images.

Report

Provide a short explanation of your solution. Be sure to document any sources you used in preparing your code, including websites and AI tools.

Discussion questions

1. Evaluate how accurately the system is able to find the books in the query images.
2. What do you notice about the search results that are not the same book as the query image? Are these search results logical or reasonable in some way?
3. The two images in `extra-queries` are of books not included in the database. What happens when you search for these books? Are the search results meaningful in any way?
4. Can you think of a way to automatically determine if the search was successful, or if the book being searched is not in the database?

Submission instructions

Submit your Python script or notebook (.py or .ipynb file) and report (PDF or docx). Please do not put them in a zip file, just submit the files directly.