CS 5330 Project Two Report

Name: Hang Zhao

Email:zhao.hang1@northeastern.edu

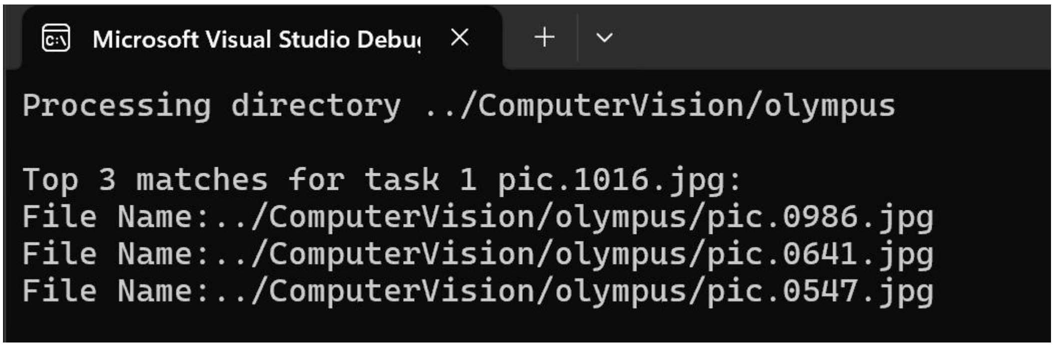
Team member: Hang Zhao

Project Description:

This project builds a content-based image retrieval system that finds similar images in a database based on visual features. It uses both traditional image processing techniques and deep learning-based embeddings. Classic methods include extracting a 7x7 center patch for direct comparison, computing color histograms for matching, and combining texture and color features using Sobel filters. The deep learning approach utilizes a pre-trained ResNet18 network to generate 512-dimensional feature vectors, with cosine distance used to measure similarity. The system is implemented in C++ with OpenCV and designed to be modular, allowing easy integration of new feature extraction methods and similarity metrics. To improve efficiency, it supports feature caching to avoid redundant computations. The program is structured to handle various retrieval tasks, and results are evaluated by comparing retrieved images against expected outputs. This project provides a flexible foundation for exploring different feature-based matching techniques in image retrieval.

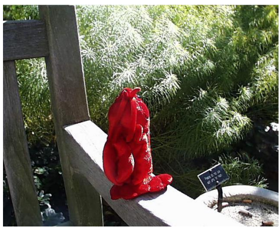
Images and description

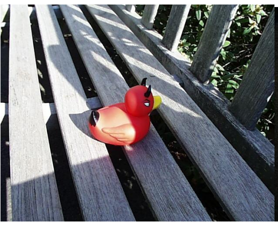
Task 1 Baseline Matching:

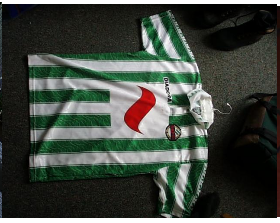


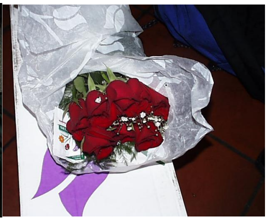
```
Processing directory ../ComputerVision/olympus

Top 3 matches for task 1 pic.1016.jpg:
File Name:../ComputerVision/olympus/pic.0986.jpg
File Name:../ComputerVision/olympus/pic.0641.jpg
File Name:../ComputerVision/olympus/pic.0547.jpg
```

The top three matches for the target image pic.1016.jpg. are pic.0986.jpg, pic.0641.jpg, pic.0547.jpg

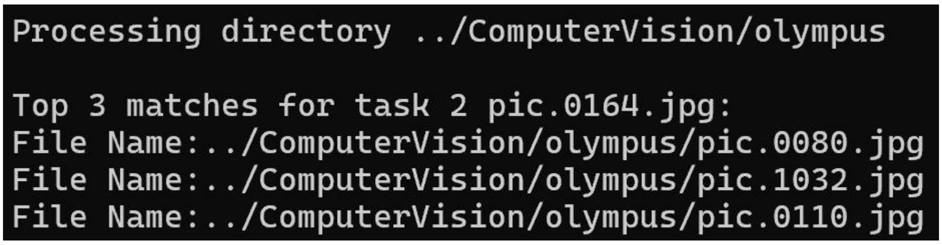

target pic.1016.jpg


pic.0986.jpg


pic.0641.jpg

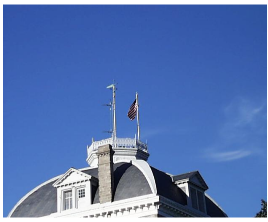

pic.0547.jpg

Task 2 Histogram Matching :

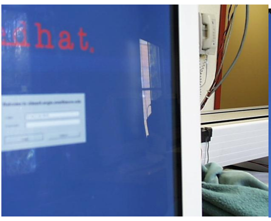


```
Processing directory ../ComputerVision/olympus

Top 3 matches for task 2 pic.0164.jpg:
File Name:../ComputerVision/olympus/pic.0080.jpg
File Name:../ComputerVision/olympus/pic.1032.jpg
File Name:../ComputerVision/olympus/pic.0110.jpg
```

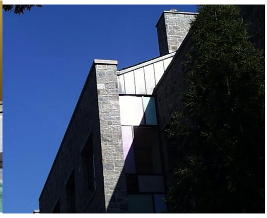
The top three matches for the target image pic.0164.jpg are pic.0080.jpg, pic.1032.jpg, pic.0110.jpg



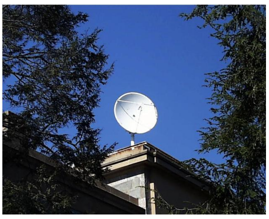
target pic.0164.jpg



pic.0080.jpg

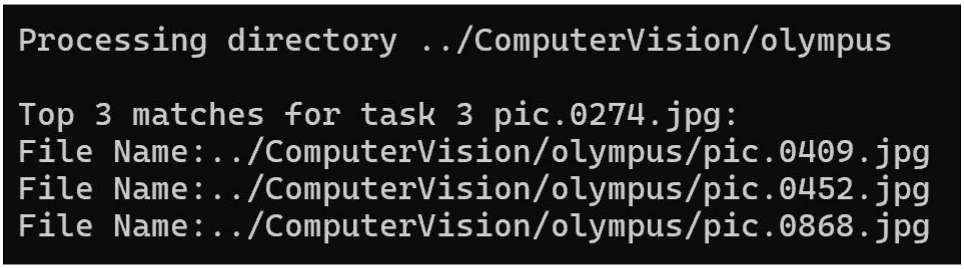


pic.1032.jpg



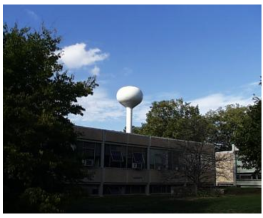
pic.0110.jpg

Task 3 Multi-histogram Matching :

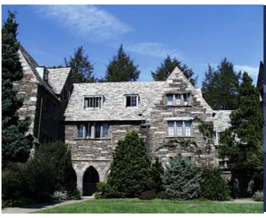


```
Processing directory ../ComputerVision/olympus

Top 3 matches for task 3 pic.0274.jpg:
File Name:../ComputerVision/olympus/pic.0409.jpg
File Name:../ComputerVision/olympus/pic.0452.jpg
File Name:../ComputerVision/olympus/pic.0868.jpg
```

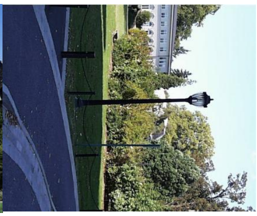
The top three matches for the target image pic.0274.jpg are pic.0409.jpg, pic.0452.jpg, pic.0868.jpg



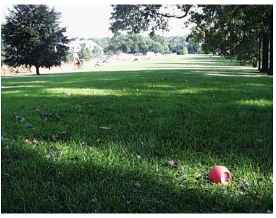
target pic.0274.jpg



pic.0409.jpg

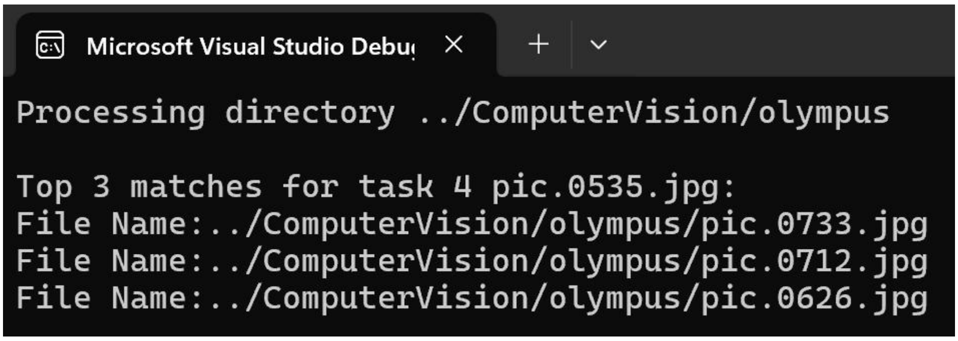


pic.0452.jpg



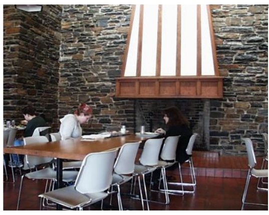
pic.0868.jpg

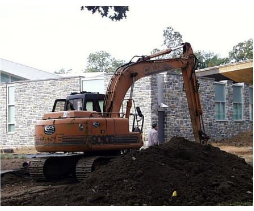
Task 4 Texture and Color:

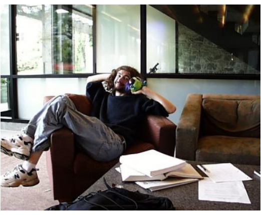


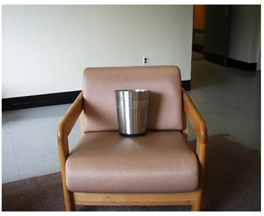
```
Processing directory ../ComputerVision/olympus

Top 3 matches for task 4 pic.0535.jpg:
File Name:../ComputerVision/olympus/pic.0733.jpg
File Name:../ComputerVision/olympus/pic.0712.jpg
File Name:../ComputerVision/olympus/pic.0626.jpg
```

The top three matches for the target image pic.0535.jpg are pic.0733.jpg, pic.0712.jpg, pic.0626.jpg


target pic.0535.jpg

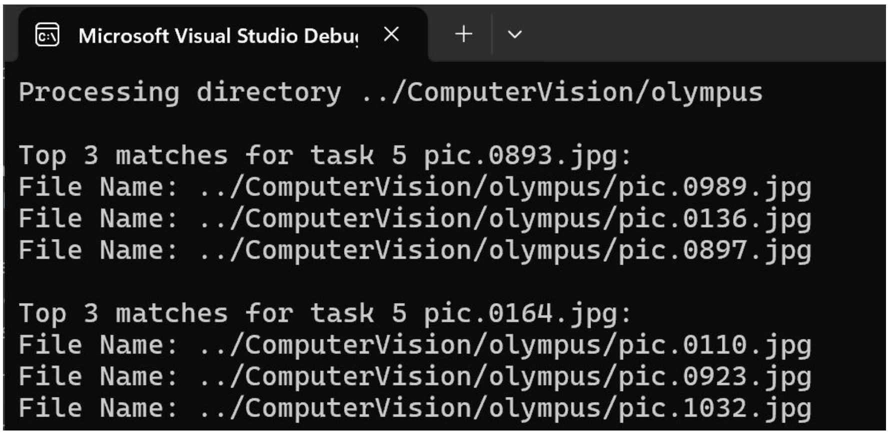

pic.0733.jpg


pic.0712.jpg


pic.0626.jpg

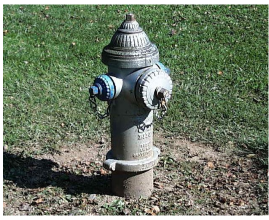
Compared to Task 2, which only considers color histograms and retrieves images with similar color distributions, Task 4 improves retrieval by incorporating texture, making it more effective for images with distinct structural patterns. Task 3, which uses multi-region histograms, accounts for spatial color variations but still lacks texture-based differentiation. As a result, Task 4 provides a more refined retrieval process by considering both color and texture, leading to different top matches that are more structurally similar to the target image rather than just color-wise.

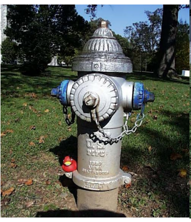
Task 5 Deep Network Embeddings:

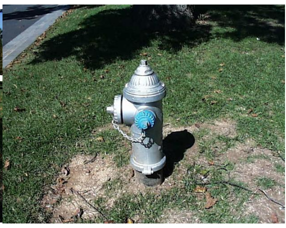


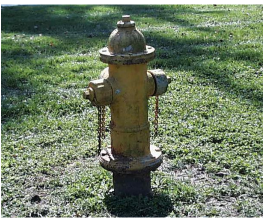
```
Processing directory ../ComputerVision/olympus

Top 3 matches for task 5 pic.0893.jpg:
File Name: ../ComputerVision/olympus/pic.0989.jpg
File Name: ../ComputerVision/olympus/pic.0136.jpg
File Name: ../ComputerVision/olympus/pic.0897.jpg

Top 3 matches for task 5 pic.0164.jpg:
File Name: ../ComputerVision/olympus/pic.0110.jpg
File Name: ../ComputerVision/olympus/pic.0923.jpg
File Name: ../ComputerVision/olympus/pic.1032.jpg
```

The top three matches for the target image pic.0893.jpg are pic.0989.jpg, pic.0136.jpg, pic.0897.jpg


target pic.0893.jpg


pic.0989.jpg


pic.0136.jpg

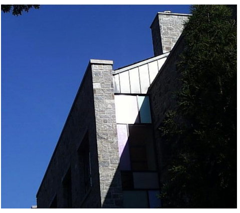

pic.0897.jpg

The top three matches for the target image pic.0164.jpg are pic.0110.jpg, pic.0923.jpg, pic.1032.jpg

target pic.0164.jpg

pic.0110.jpg

pic.0923.jpg

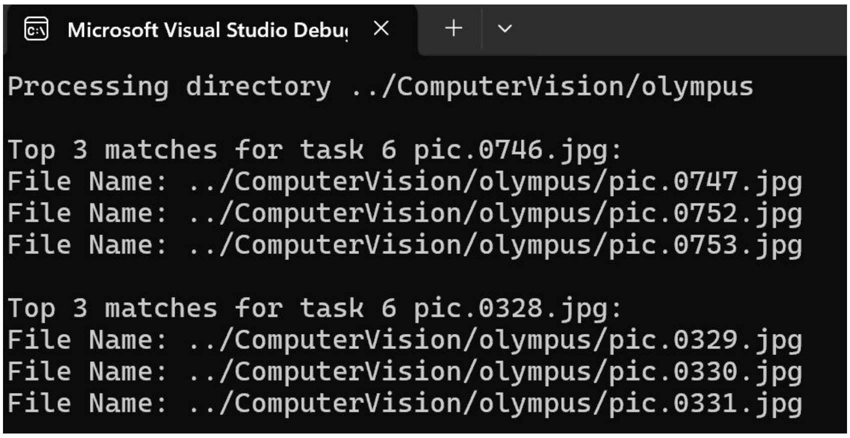

pic.1032.jpg

Comparison with Prior Methods:

Unlike histogram or texture-based retrieval, deep embeddings capture high-level semantic features learned from a large dataset (ImageNet). This means the retrieved images are likely to be more perceptually similar rather than just color or texture-wise similar.

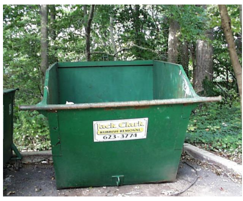
This method generally performs better when objects in images have similar structures but differ slightly in color or texture. However, in cases where color or texture is the dominant similarity factor (e.g., in abstract images), classic methods like histogram matching may outperform deep embeddings.

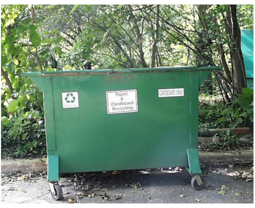
Task 6 Compare DNN Embeddings and Classic Features:

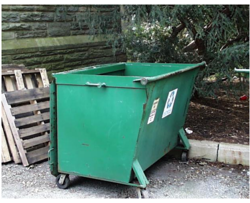


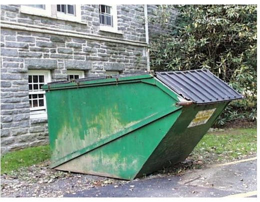
```
Processing directory ../ComputerVision/olympus

Top 3 matches for task 6 pic.0746.jpg:
File Name:  ../ComputerVision/olympus/pic.0747.jpg
File Name:  ../ComputerVision/olympus/pic.0752.jpg
File Name:  ../ComputerVision/olympus/pic.0753.jpg

Top 3 matches for task 6 pic.0328.jpg:
File Name:  ../ComputerVision/olympus/pic.0329.jpg
File Name:  ../ComputerVision/olympus/pic.0330.jpg
File Name:  ../ComputerVision/olympus/pic.0331.jpg
```

The top three matches for the target image pic.0746.jpg are pic.0747.jpg, pic.0752.jpg, pic.0753.jpg

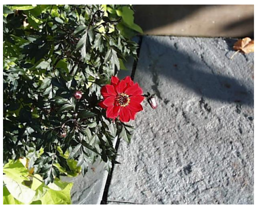

target pic.0746.jpg

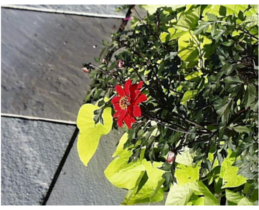

pic.0747.jpg

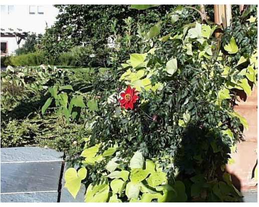

pic.0752.jpg

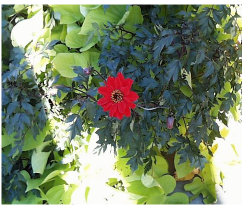

pic.0753.jpg

The top three matches for the target image pic.0328.jpg are pic.0329.jpg, pic.0330.jpg, pic.0331.jpg





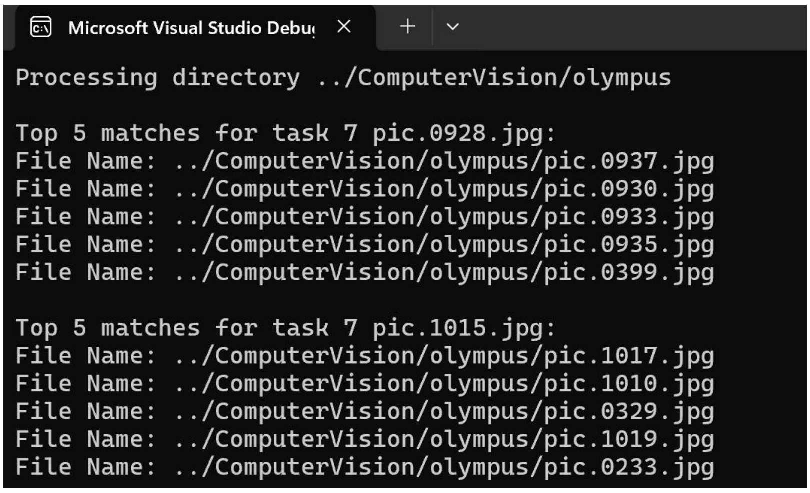




target pic.0328.jpg pic.0329.jpg pic.0330.jpg pic.0331.jpg

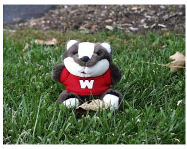
No, DNN embeddings are not always better, it depends on the task. If you are looking for structurally similar objects, DNN embeddings work very well because they capture shape and high-level features. But if color or texture is the key similarity (such as matching paintings or fabric patterns), then classic features such as histograms may actually perform better. The 2 examples above explain this very well.

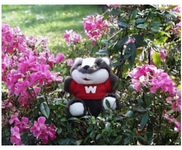
Task 7 Custom Design:

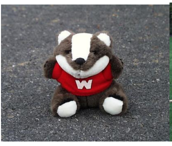


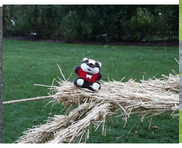
```
Processing directory ../ComputerVision/olympus

Top 5 matches for task 7 pic.0928.jpg:
File Name: ../ComputerVision/olympus/pic.0937.jpg
File Name: ../ComputerVision/olympus/pic.0930.jpg
File Name: ../ComputerVision/olympus/pic.0933.jpg
File Name: ../ComputerVision/olympus/pic.0935.jpg
File Name: ../ComputerVision/olympus/pic.0399.jpg

Top 5 matches for task 7 pic.1015.jpg:
File Name: ../ComputerVision/olympus/pic.1017.jpg
File Name: ../ComputerVision/olympus/pic.1010.jpg
File Name: ../ComputerVision/olympus/pic.0329.jpg
File Name: ../ComputerVision/olympus/pic.1019.jpg
File Name: ../ComputerVision/olympus/pic.0233.jpg
```

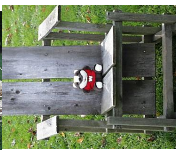
The top five matches for the target image pic.0928.jpg are pic.0937.jpg, pic.0930.jpg, pic.0933.jpg, pic.0935.jpg, pic.0399.jpg

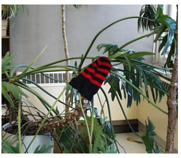

target pic.0928.jpg


pic.0937.jpg

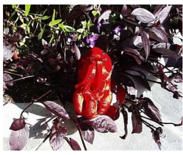

pic.0930.jpg

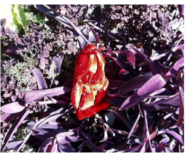

pic.0933.jpg

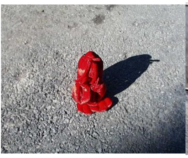

pic.0935.jpg

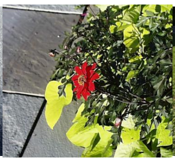

pic.0399.jpg

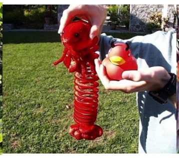
The top five matches for the target image pic.1015.jpg are pic.1017.jpg, pic.1010.jpg, pic.0329.jpg, pic.1019.jpg, pic.0233.jpg

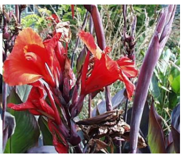

target pic.1015.jpg


pic.1017.jpg


pic.1010.jpg


pic.0329.jpg


pic.1019.jpg


pic.0233.jpg

For Task 7, I focused on retrieving images containing toy on grass and red flower in dark leaves using a combination of texture, color histograms, and deep embeddings.

Feature Extraction Approach:
pic.0928.jpg 4/5 correctness
Used Sobel-based texture histograms to capture gradient patterns in the grass. Applied histogram intersection for similarity measurement.

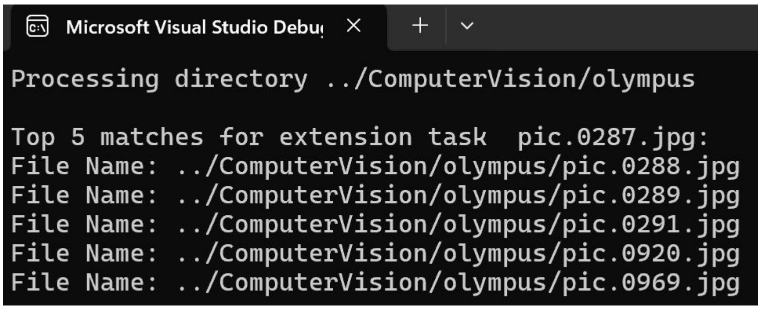
pic.1015.jpg 3/5 correctness

Extracted HSV color histograms, focusing on the red hue. Incorporated DNN embeddings from ResNet18 to enhance object recognition. Used weighted combination of color similarity and deep feature distance.

Conclusion:

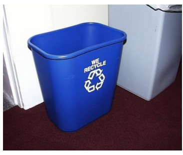
Texture-Based Retrieval worked well but struggled with mixed backgrounds (e.g., sky regions misclassified as grass). Color + DNN Retrieval was more robust but sometimes picked images with partial red objects instead of full statues. A hybrid method combining color, texture, and deep embeddings would improve accuracy further.

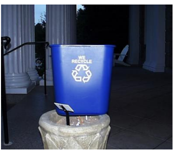
Extensions:

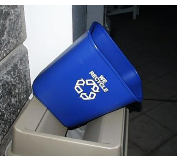


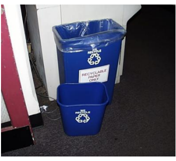
```
Processing directory ../ComputerVision/olympus

Top 5 matches for extension task  pic.0287.jpg:
File Name: ../ComputerVision/olympus/pic.0288.jpg
File Name: ../ComputerVision/olympus/pic.0289.jpg
File Name: ../ComputerVision/olympus/pic.0291.jpg
File Name: ../ComputerVision/olympus/pic.0920.jpg
File Name: ../ComputerVision/olympus/pic.0969.jpg
```

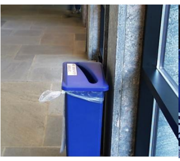
The top five matches for the target image pic.0287.jpg are pic.0288.jpg, pic.0289.jpg, pic.0291.jpg, pic.0920.jpg, pic.0969.jpg

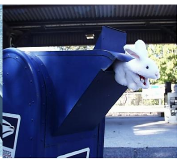

target pic.0287.jpg


pic.0288.jpg


pic.0289.jpg


pic.0291.jpg


pic.0920.jpg


pic.0969.jpg

I built a system to find blue trash bins using color histograms, shape analysis, and deep embeddings. It filters images by blue color, checks for rectangular shapes, and refines results with ResNet18 features. The system ranked images based on a weighted score, and most top matches were actual trash bins, while false matches were filtered out. Adding shape and deep features made retrieval much more accurate, and with depth filtering, it could get even better!

Reflection:

This project deepened my understanding of content-based image retrieval, highlighting the strengths and limitations of classic features (color histograms, texture descriptors) versus deep learning-based embeddings. I improved my C++ and OpenCV skills, optimizing feature extraction, caching, and similarity computations. Implementing different distance metrics reinforced the importance of feature selection in retrieval accuracy. Additionally, designing a modular system made it easier to compare methods and extend functionality. Exploring object-specific retrieval (e.g., detecting bananas) demonstrated practical applications of feature-based matching. Overall, this project enhanced my knowledge of computer vision, image analysis, and efficient retrieval techniques.

Acknowledgement:

I would like to thank our teaching assistants and classmates for their valuable support and guidance throughout this project. Their help in troubleshooting technical issues and providing feedback was essential in completing our work.

I also want to express our gratitude to our professor for providing clear explanations and insightful examples that deepened our understanding of computer vision. The resources and tutorials shared during the course were incredibly helpful in learning OpenCV and applying it effectively to this project.