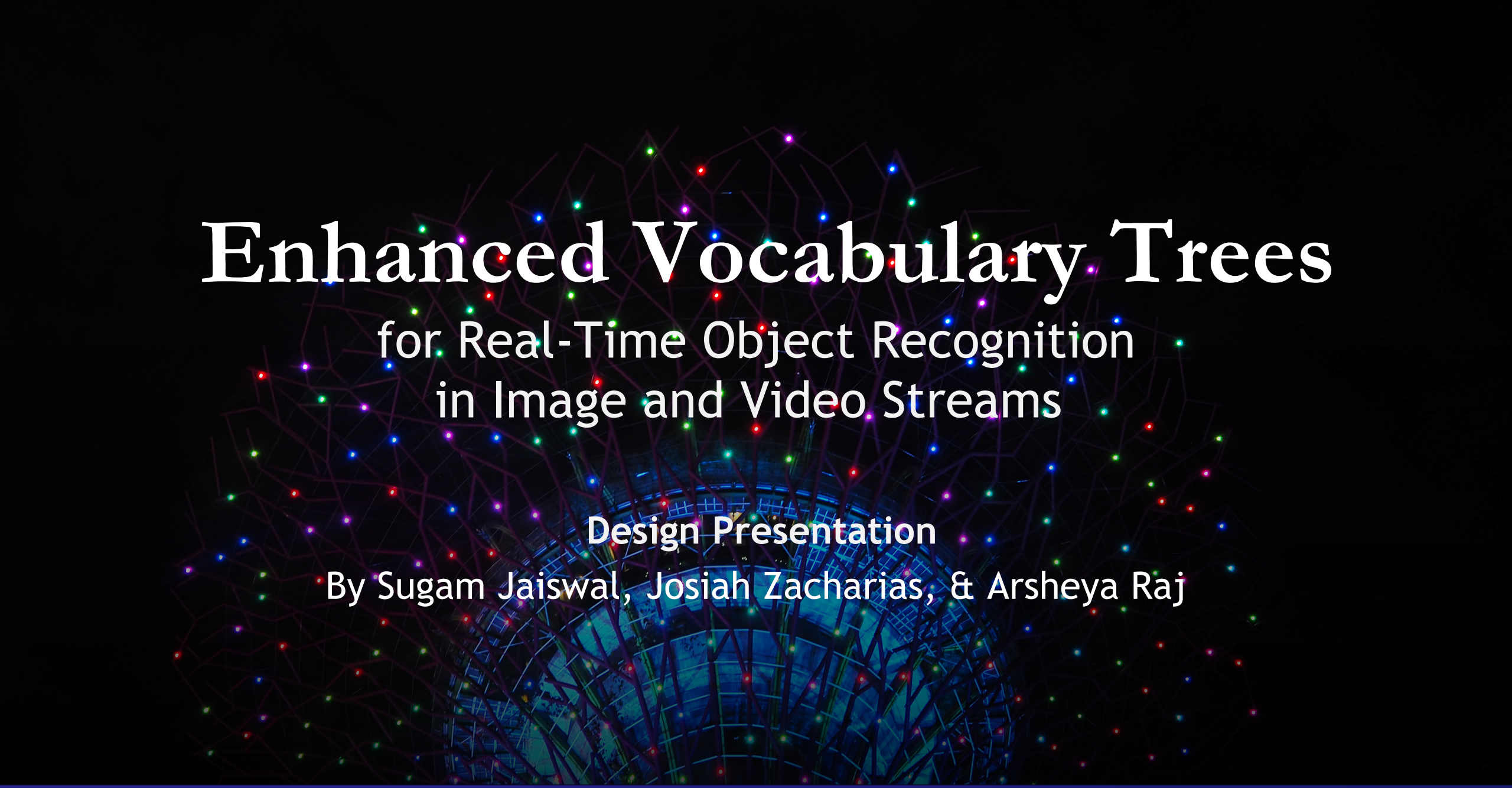


Enhanced Vocabulary Trees

for Real-Time Object Recognition in Image and Video Streams

Design Presentation

By Sugam Jaiswal, Josiah Zacharias, & Arsheya Raj

UNIVERSITY of WASHINGTON | BOTHELL
COMPUTING & SOFTWARE SYSTEMS

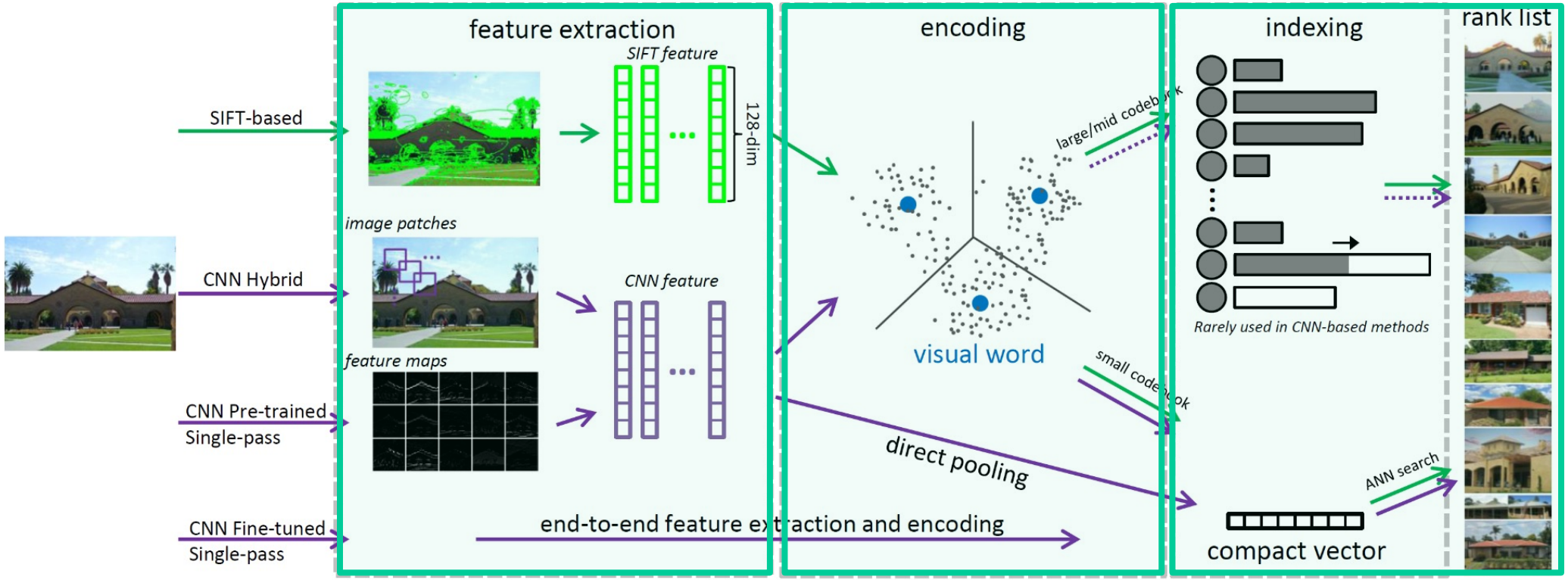
Contents

Project Overview

Goals:

1. Object Detection
   - Detect objects in large datasets of images and videos (e.g. COCO)
   - Achieve better performance than CNN (neural nets have poor performance on video frames)
2. Build Optimized Vocabulary Tree
   - Construct using hierarchical K-means clustering
   - Attempt other clustering algorithms and detector experimentation
3. Project as SaaS (hopeful)
   - Deploy to web application for service as business pitch
   - Only to be done once body of testing and performance is completed

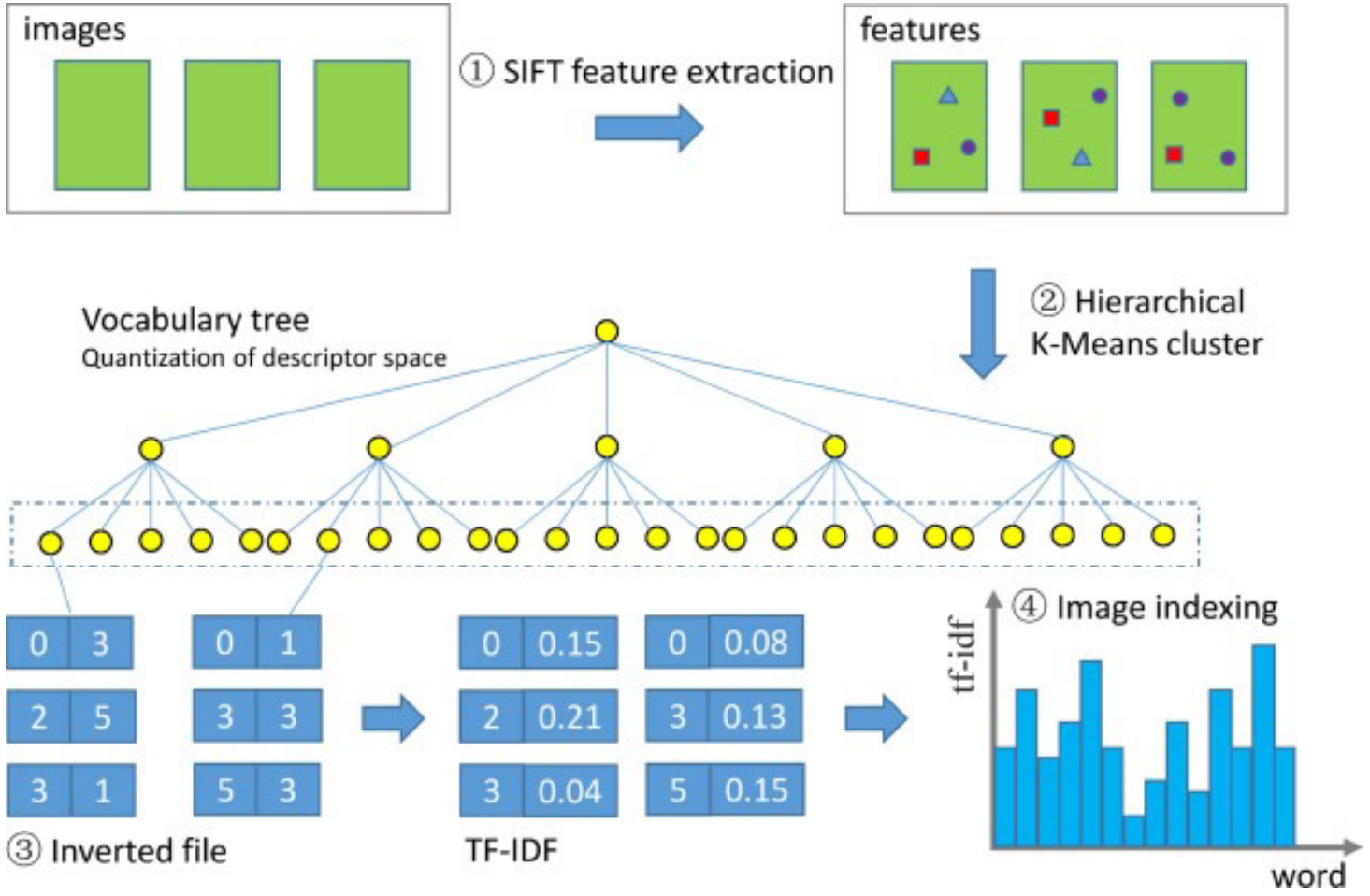
Reference Paper

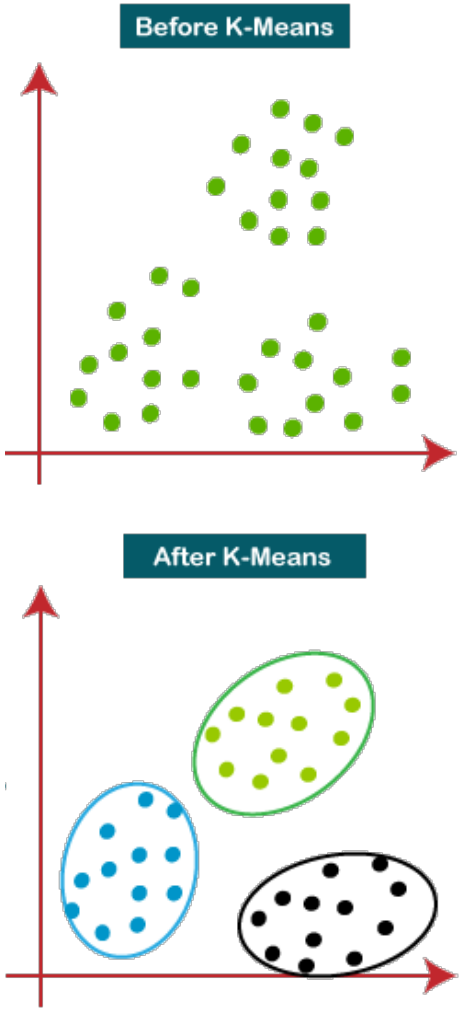


Nister, D., & Stewenius, H. (2006). Scalable Recognition with a Vocabulary Tree. In 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06) (Vol. 2, pp. 2161-2168). IEEE.
https://inst.eecs.berkeley.edu//~cs294-6/fa06/papers/nister_stewenius_cvpr2006.pdf

Input Datasets

K-Means Tree Algorithm



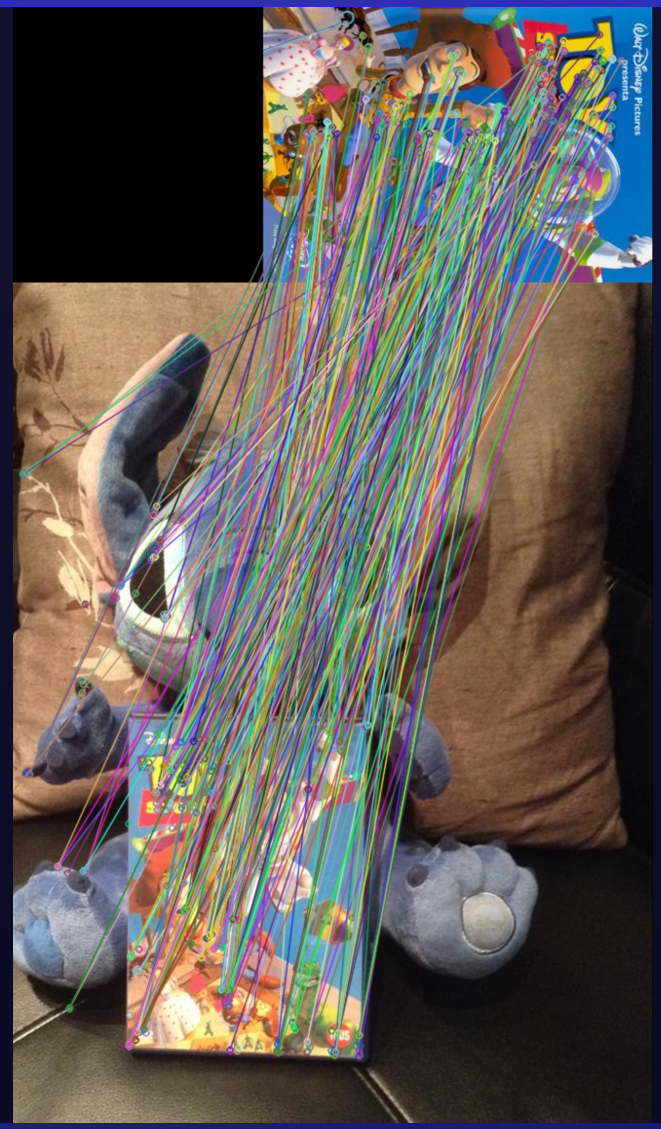


https://www.sciencedirect.com/science/article/abs/pii/S0924271622000727

Feature Detectors

Considerations:

1. Sift
   - Basis of original paper's approach
   - Robust to scaling, rotation, and changes in lighting
2. SURF (Speeded-Up Robust Features)
   - Improved version of SIFT—faster and more efficient
3. ORB (Oriented FAST and Rotated BRIEF)
   - Fast, invariant to rotation and robust to noise.
   - Combines FAST keypoint detector and BRIEF descriptor and is more computationally efficient than SIFT or SURF.



Project Scope / Application

Project scope :

1. Vocabulary Tree
   - Address challenges of large-scale visual recognition tasks using a vocabulary tree approach
2. Improved algorithms
   - Feature detection/extraction, image representation, indexing, and efficient search techniques within a hierarchical structure
3. Large visual datasets
   - Enhance recognition performance and scalability in handling large-scale visual datasets

Potential Applications:

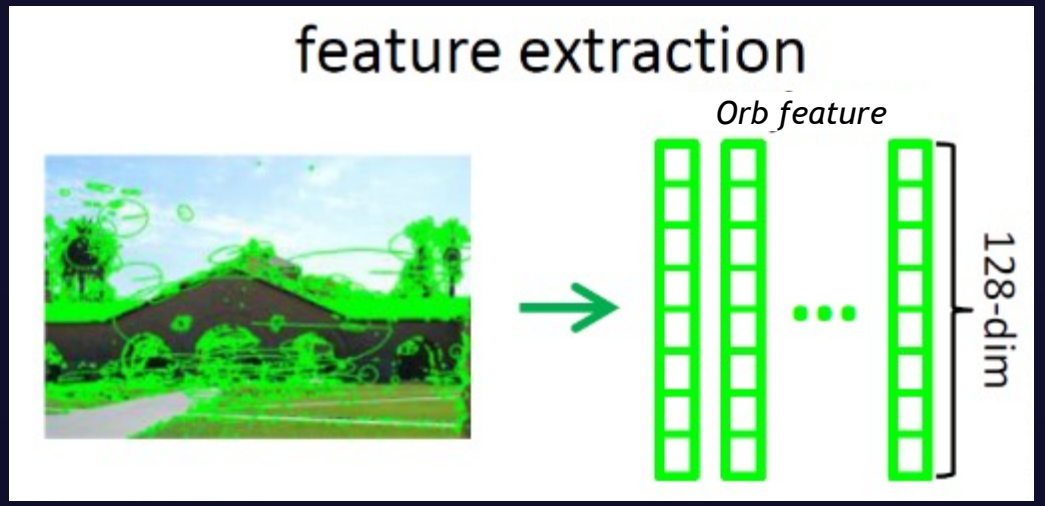
1. Object recognition
   - Identify /classify objects in images /videos, for automated image tagging, object detection, and visual search
2. Image retrieval
   - Applications like efficient content-based image retrieval and plagiarism detection from large databases
3. Scene recognition
   - Automated analysis and categorization of images based on their contextual information
4. Visual surveillance
   - Activities such as object / person tracking, anomaly detection, and event recognition
5. Augmented reality
   - Overlay of virtual objects onto the real-world environment in augmented reality applications
6. Medical imaging
   - Medical image analysis, assisting in tasks like tumor detection, organ segmentation, and disease classification

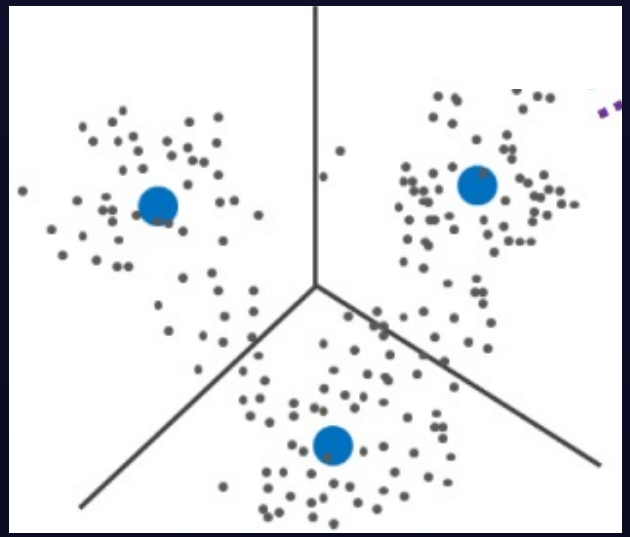
Data Structures

- OpenCV available structures
  - TermCriteria
  - Kmeans() function
  - FeatureDetector — ORB
- Non-OpenCV (Trees)
  - Tree implemented via Vector<Mat> in database
- Images from python Library called pycocotools.COCO dataset
  - Current sample 10 images, implemented with Python
- Test data and Testing methodology
  - The test data contains 10 images of planes and these planes are divided in the clusters. Another test contained 10 images from planes, people and clock divided into 3 clusters
- Score calculation in testing data
  - Linear algebra-based score calculation normalization

Program Design

- Input dataset installation
- ORB detector
- K-means clustering
  - Agglomerative clustering
- Max Voting
- Cluster Reassignment
- Database construction and indexing
- Score Calculation



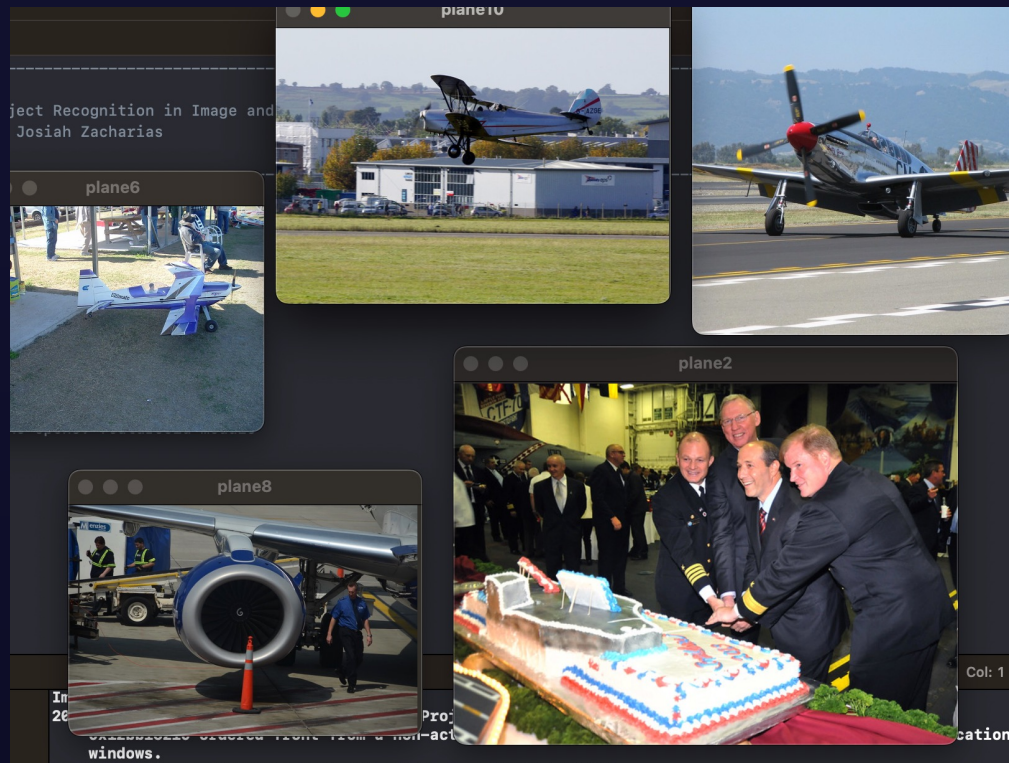


K=3 for k-means and we are running 50 iterations for the convergence. EPS value is 1.0 for the eigen values distance calculation.

Preliminary Results

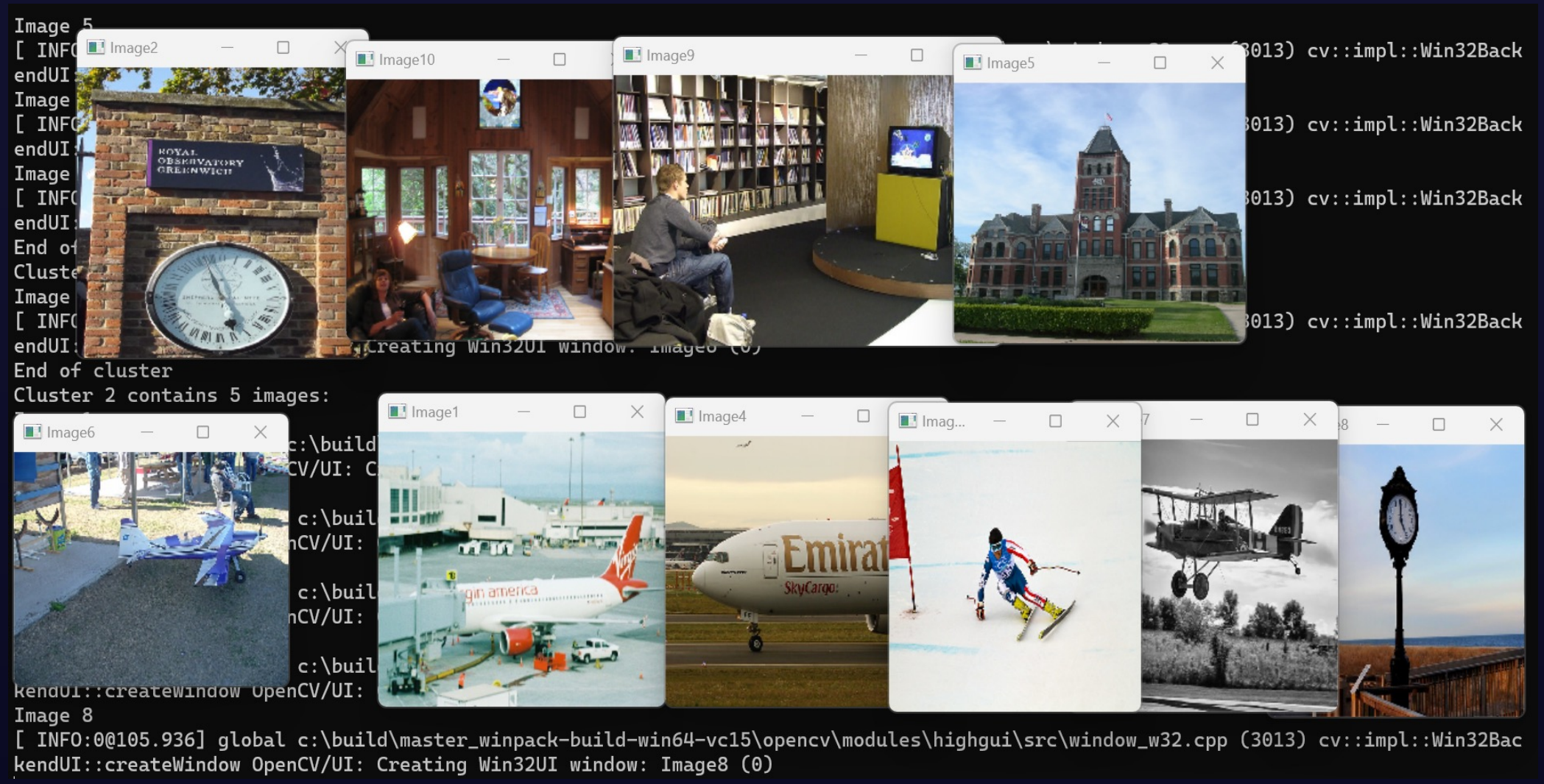

```
2023-05-13 15:06:03.071371-0700 CSS 587 Project[21613:3
2023-05-13 15:06:03.104031-0700 CSS 587 Project[21613:3
    Failed to retrieve device information! Invalid enum

Cluster 0 contains 1 images:
Image 8
2023-05-13 15:06:03.376404-0700 CSS 587 Project[21613:3
    0x12bb16210 ordered front from a non-active applica
    windows.
```


```
enumerated value!
Cluster 0 contains 1 images:
Image 8
2023-05-13 15:06:03.376404-0700 CSS 587 Project[21613:349126] [Window] Warning: Window CVWi
    0x12bb16210 ordered front from a non-active application and may order beneath the activ
    windows.
2023-05-13 15:06:31.338969-0700 CSS 587 Project[21613:349126]
    GetDYLDEntryPointWithImage(/System/Library/Frameworks/AppKit
    .framework/Versions/Current/AppKit,_NSCreateAppKitServicesMenu) failed.
End of cluster
Cluster 1 contains 4 images:
Image 0
Image 3
Image 4
Image 6
```



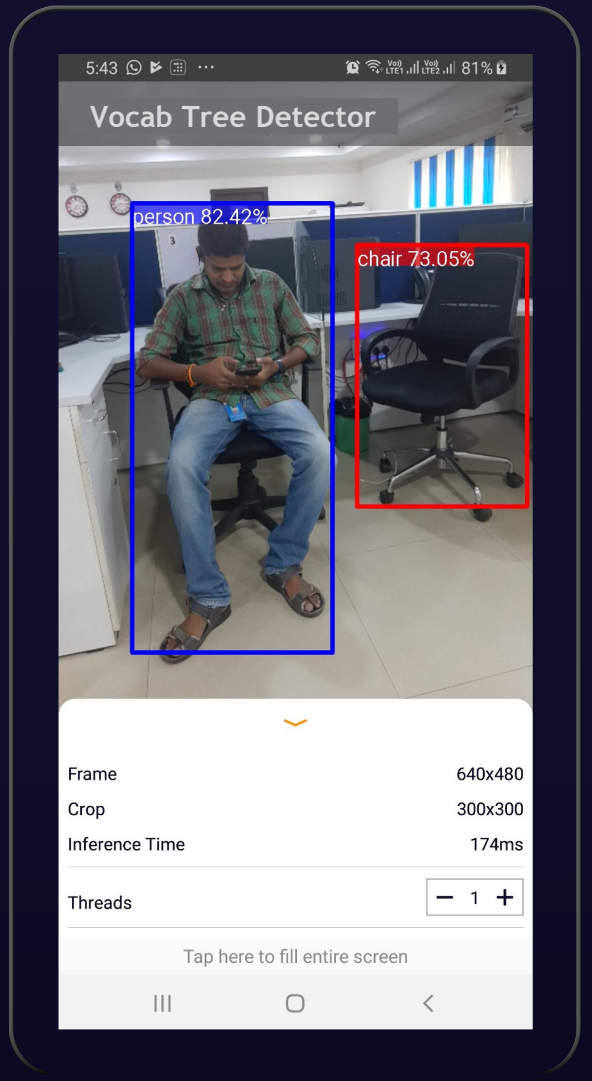

```
windows.
2023-05-13 15:06:31.338969-0700 CSS 587 Project[21613:349126]
    GetDYLDEntryPointWithImage(/System/Library/Frameworks/AppKit
    .framework/Versions/Current/AppKit,_NSCreateAppKitServicesMenu) failed.
End of cluster
Cluster 1 contains 4 images:
Image 0
Image 3
Image 4
Image 6
End of cluster
Cluster 2 contains 5 images:
Image 1
Image 2
Image 5
```

Preliminary Results



Future Work

1. Try with the bigger dataset of COCO
2. Try different Feature Detectors
3. Try different clustering algorithms (agglomerative clustering)
4. Reduce Noise and get a good score
5. Compare results with research paper
6. SaaS implementation and User Interface



References

Nister, D., & Stewenius, H. (2006). *Scalable Recognition with a Vocabulary Tree*. In 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06) (Vol. 2, pp. 2161-2168). IEEE._

https://inst.eecs.berkeley.edu//~cs294-6/fa06/papers/nister_stewenius_cvpr2006.pdf

We will use the OpenCV library for implementing the vocabulary tree and object recognition system.

Additionally, we plan to utilize the code and ideas provided in the GitHub repository "Scalable Recognition with a Vocabulary Tree" as a starting point.

Fifty One integrations: The open-source tool for building high-quality datasets and computer vision models.

https://docs.voxel51.com/