State University of New York at Buffalo

CSE 473/573: Summer 2016
Computer Vision and Image Processing

PROGRAMMING ASSIGNMENT #3
Image Search Engine

-Prithvi Gollu Indrakumar
UB #: 50169089

Image Based Search Engine- Report

Introduction

In this project a new image dataset was created, which consisted of 5 images each from various angles of 5 landmarks in UB north campus: Lake Lasalle, Norton Hall, Capen Hall, Clemens Hall and commons. A total of 25 images were indexed and the working of the code was analyzed. The search engine used was written by Adrian Rosebrock. We display top 10 similar images for each query.

This search engine was built in 4 steps:

1. **Define descriptor**: Here 3D RGB histogram is used as a descriptor
2. **Index dataset**: A 3D RGB histogram of all the 25 images as described by the descriptor was extracted and stored in the form of a index dictionary
3. **Define similarity metric**: The color histogram of all the images are compared with each other using chi-squared distance and the results are stored in terms of relevancy.
4. **Searching**: First descriptor is applied on the query image, and then the distance metric ranks how similar the images in the dictionary are to the query image. These results are then sorted via similarity metric and displayed.

Accuracy

Accuracy of the search engine is calculated by using the following formula:

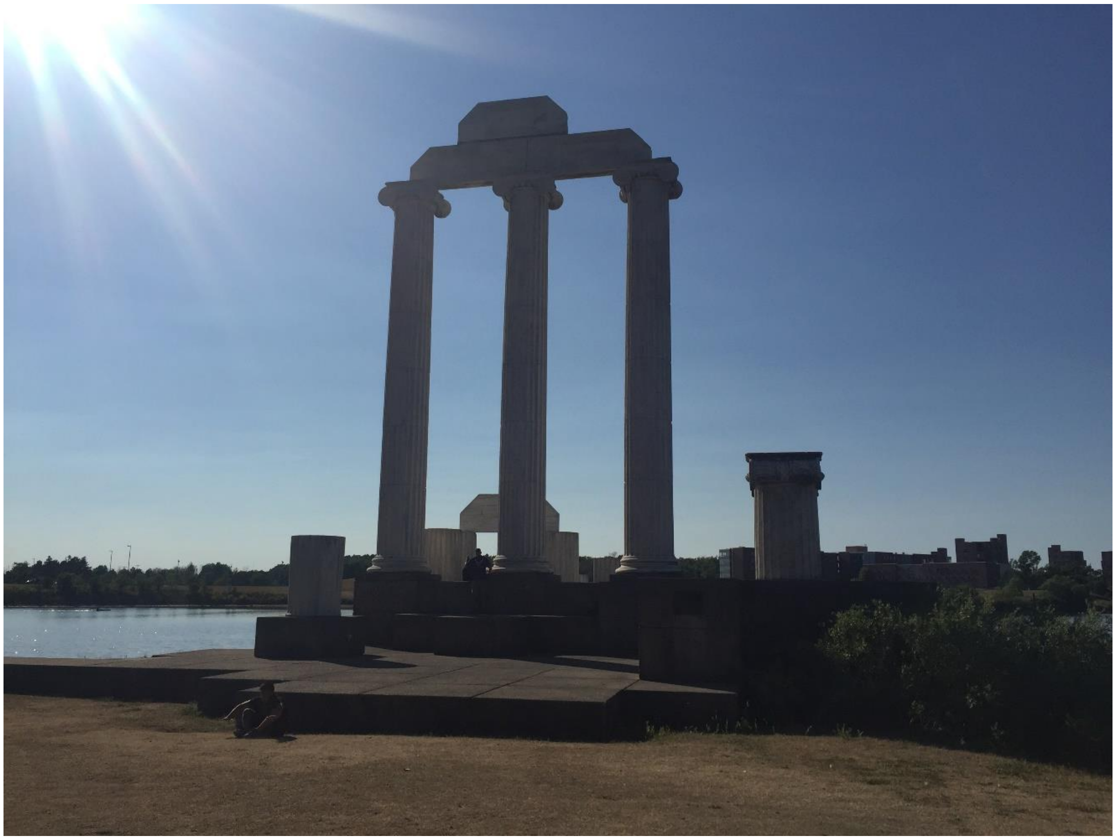
Accuracy = No of relevant results/Total returned results

Accuracy is calculated for top 5 as the dataset consists of only 5 relevant images and top 10 as we display top 10 similar images.

Analysis

- ***Lake Lasalle***

Query Image:



Results Snapshot: Top 10

Baldy Hall

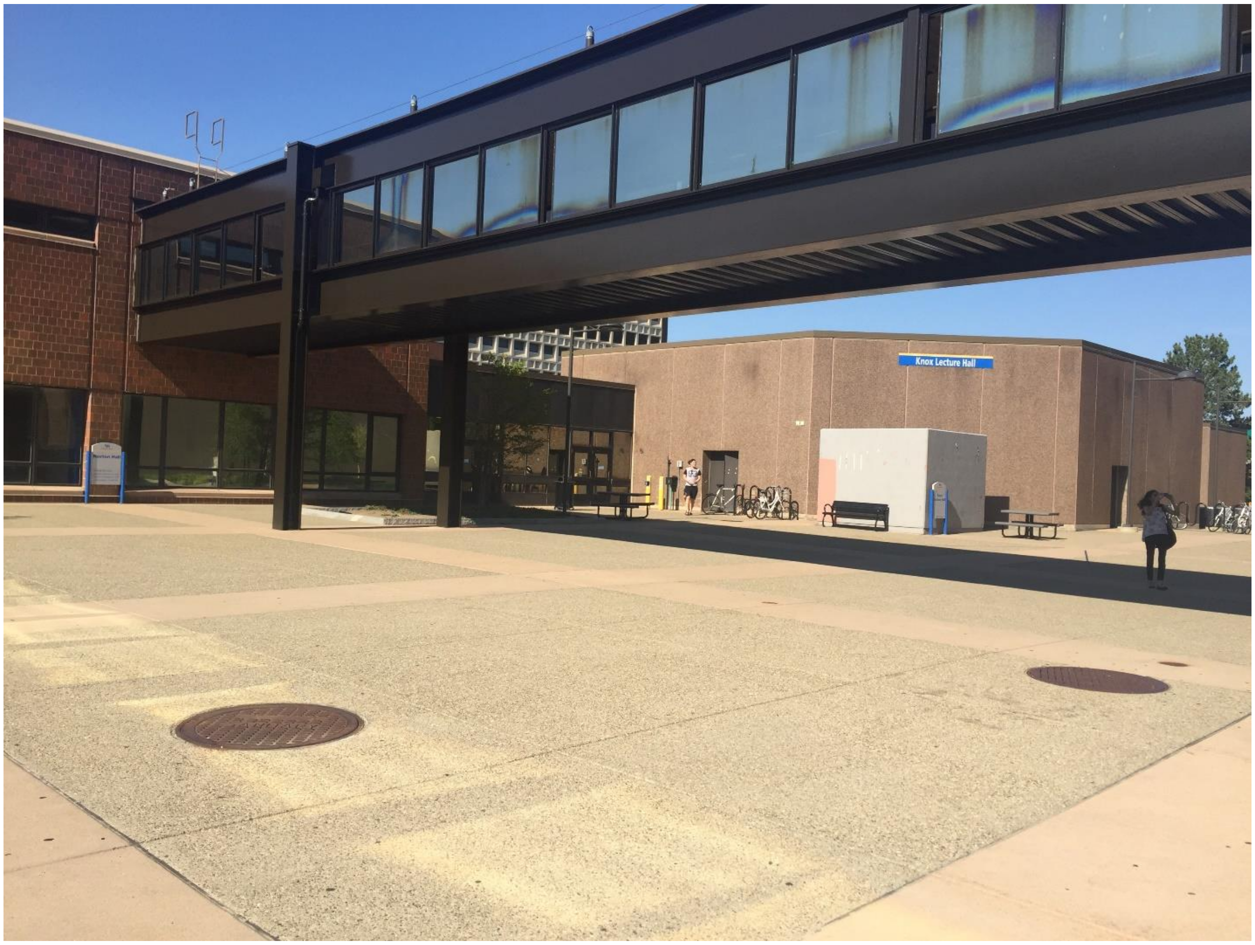
Accuracy

Accuracy when we consider only the top 5 search results: **3/5=0.6**

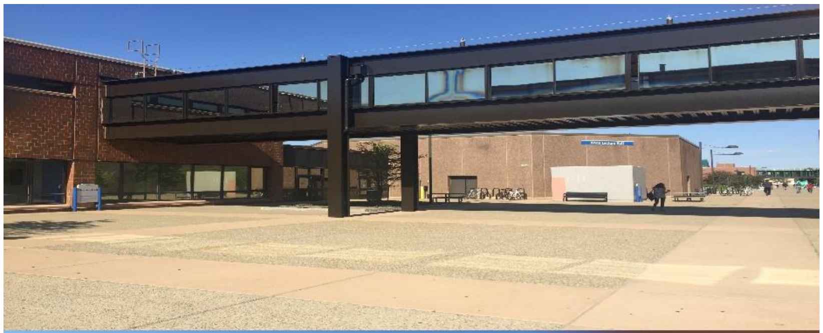
Accuracy when we consider the top 10 search results: **5/10=0.5**

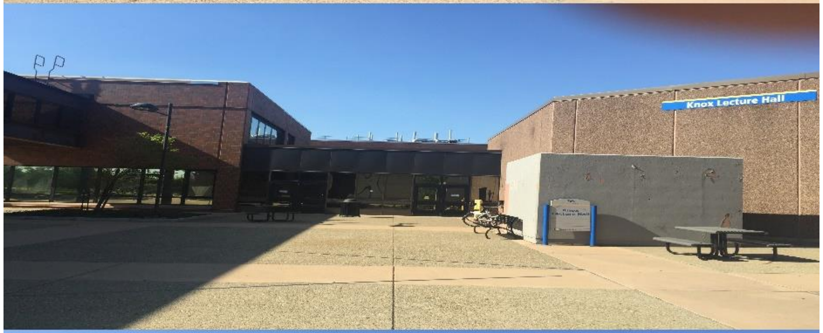
- ***Norton Hall***

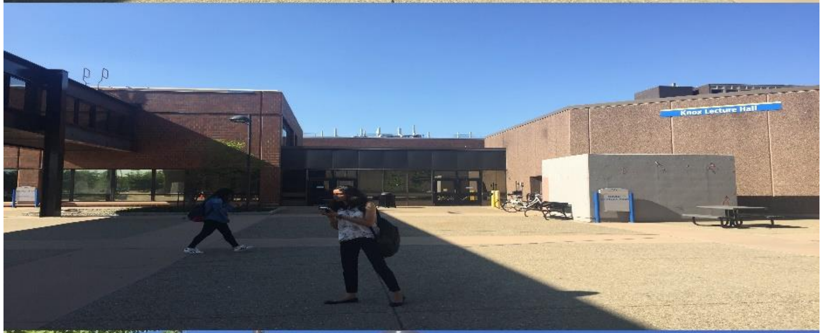
Query Image:

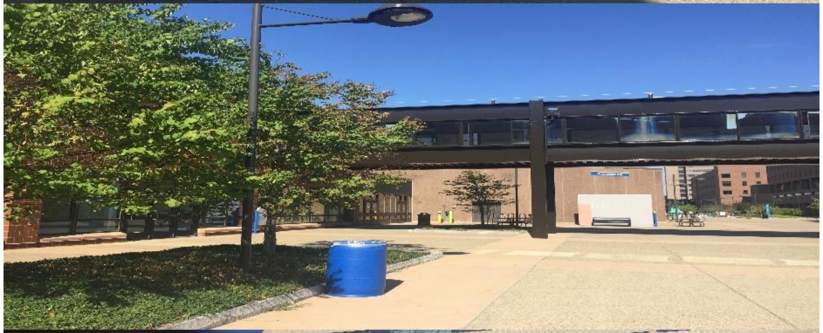


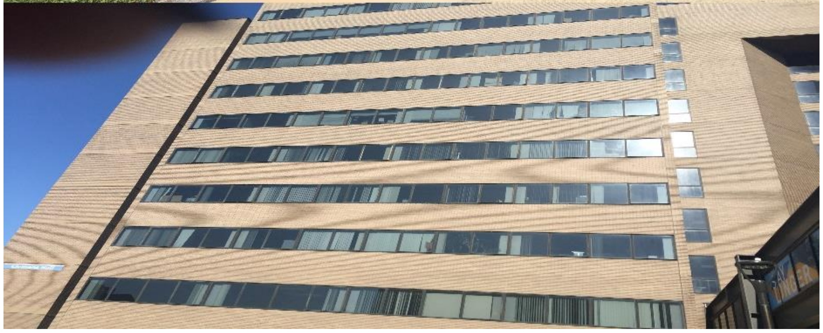
Results Snapshot: Top 10



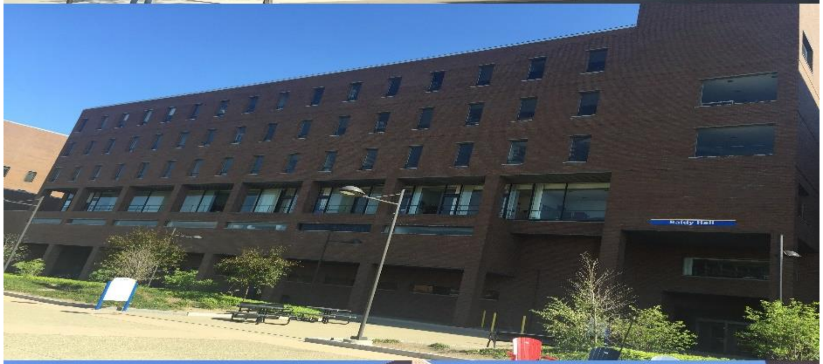








Knox Lecture Hall


Baldy Hall

THE COMMONS

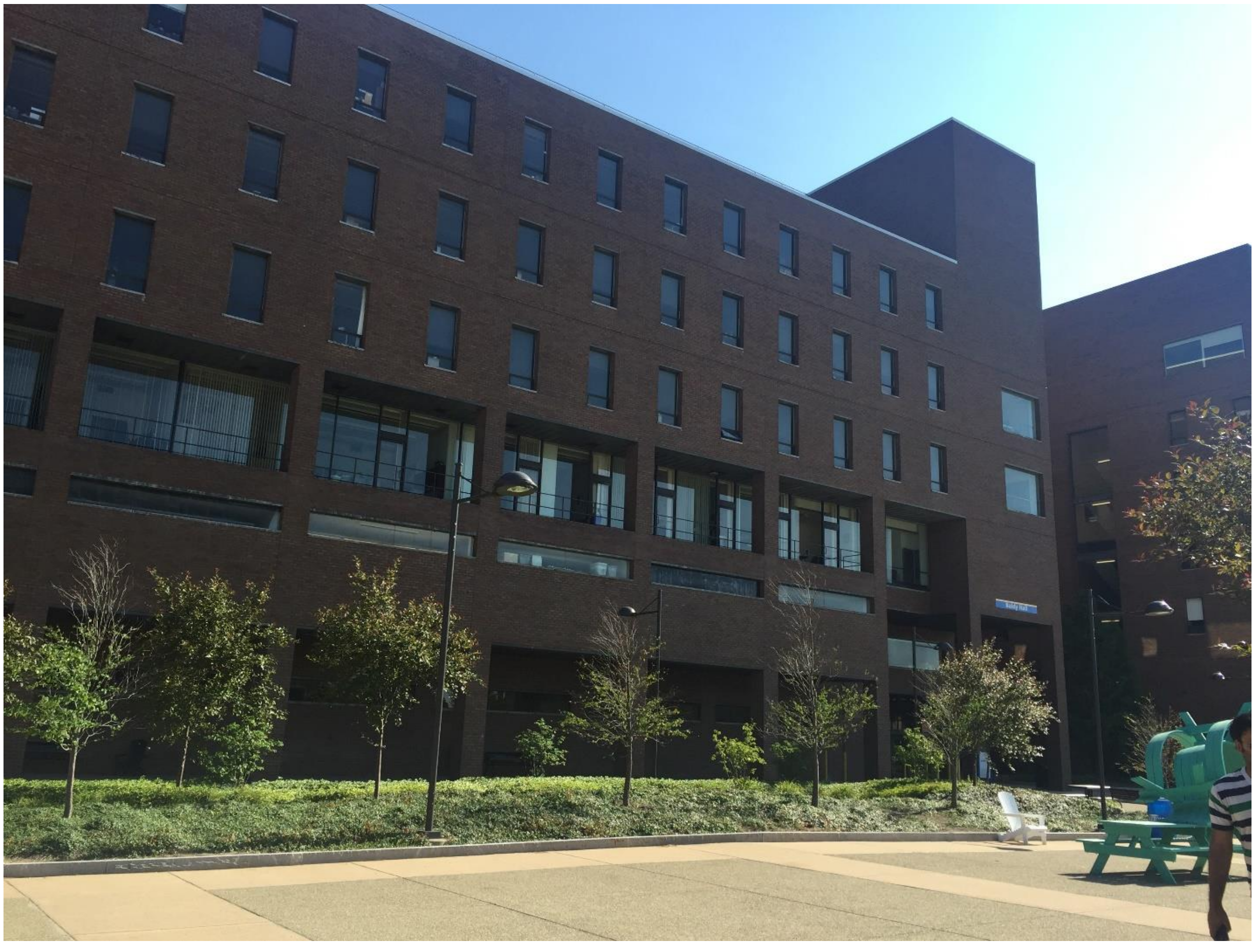
Accuracy

Accuracy when we consider only the top 5 search results: **4/5=0.8**

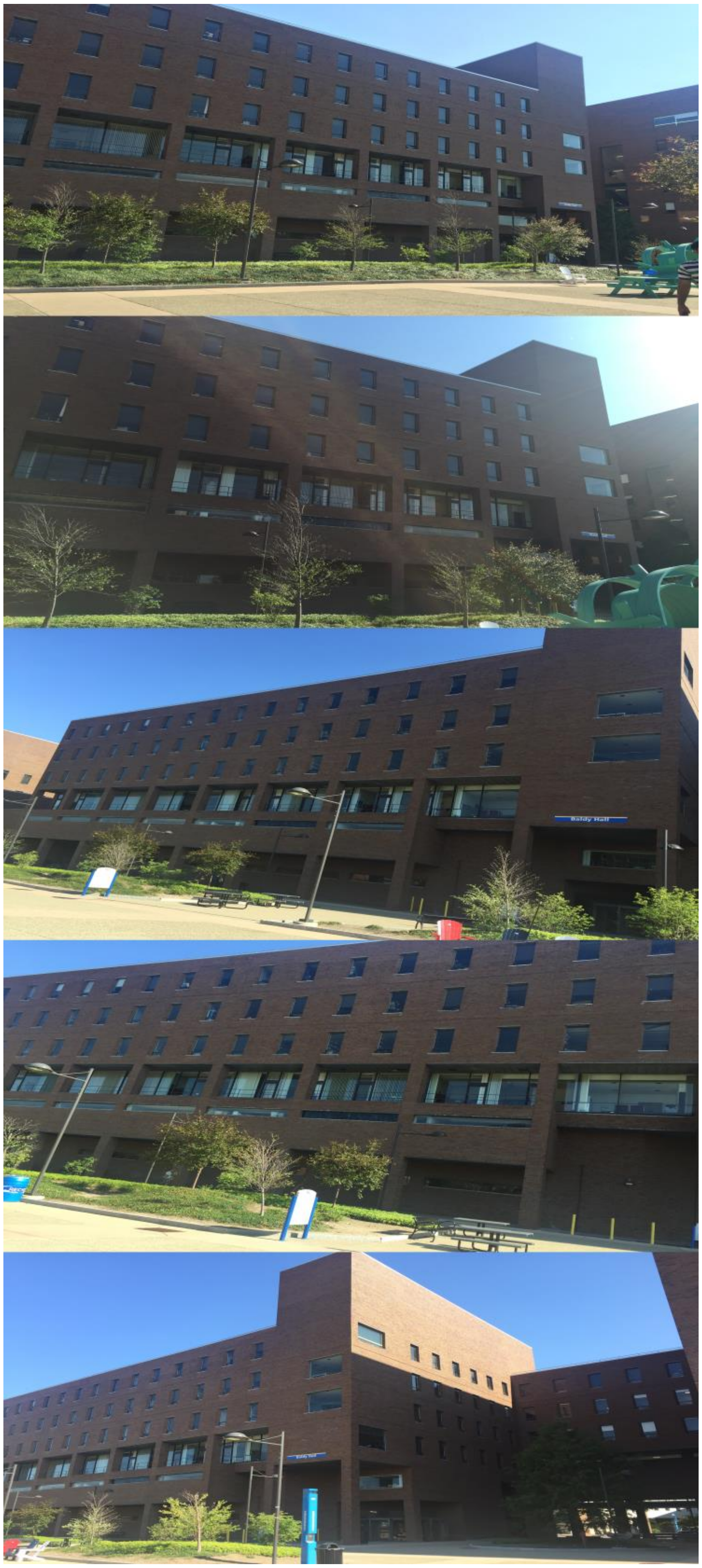
Accuracy when we consider the top 10 search results: **5/10=0.5**

- ***Capen Hall***

Query Image:



Results Snapshot: Top 10



Knox Lecture Hall

Knox Lecture Hall

THE COMMONS

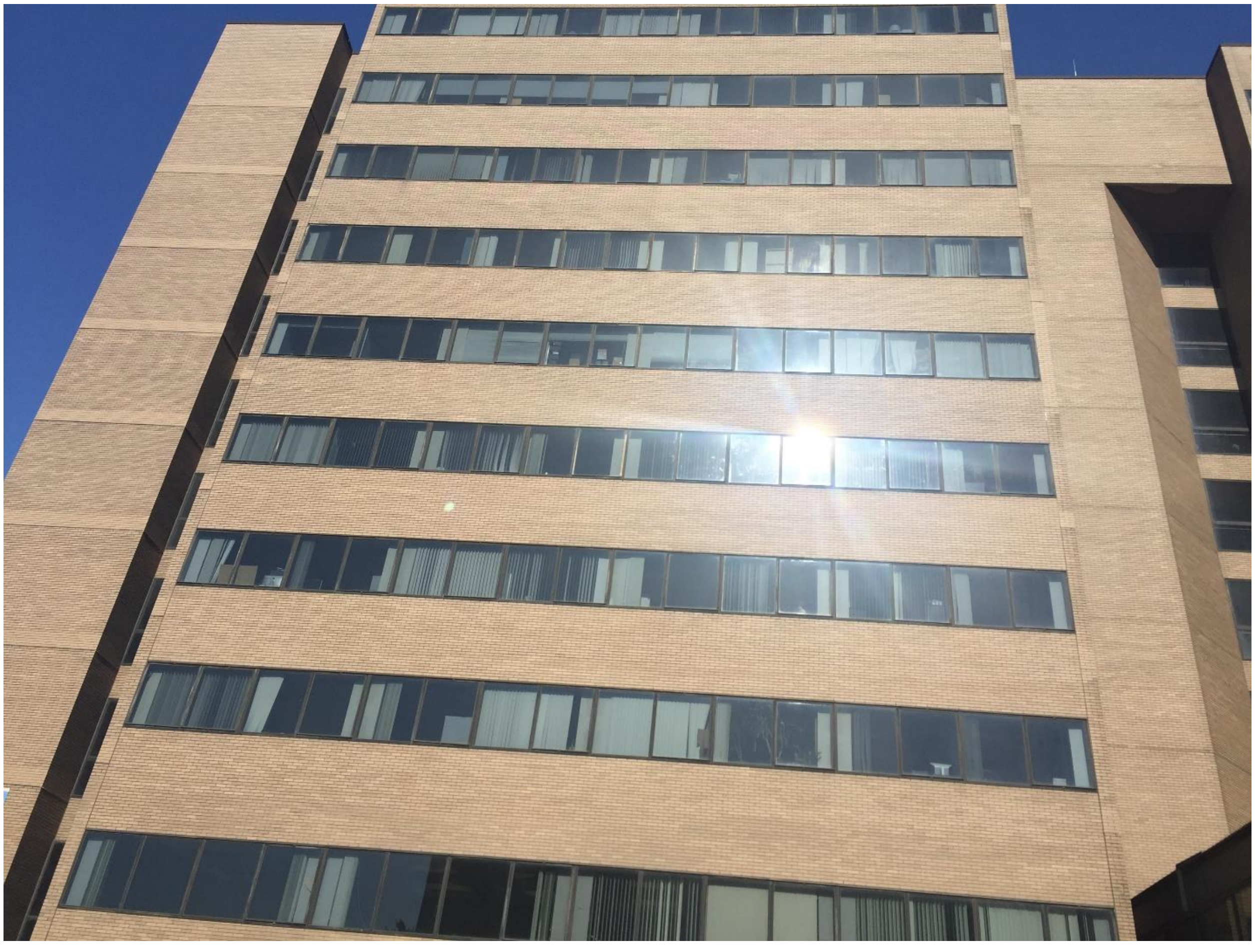
Accuracy

Accuracy when we consider only the top 5 search results: **5/5=1**

Accuracy when we consider the top 10 search results: **5/10=0.5**

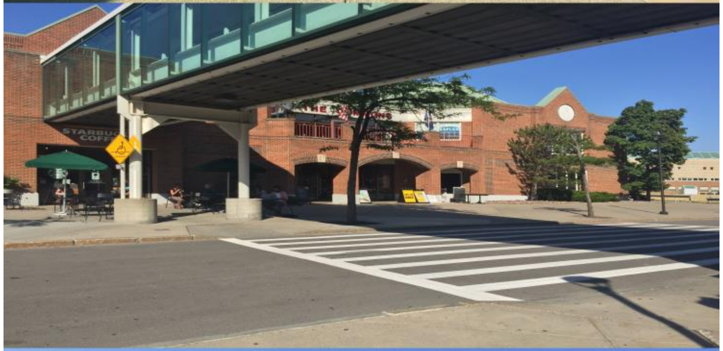
- ***Clemens Hall***

Query Image:



Results Snapshot: Top 10

Knox Lecture Hall


STARBU
COFF

Knox Lecture Hall

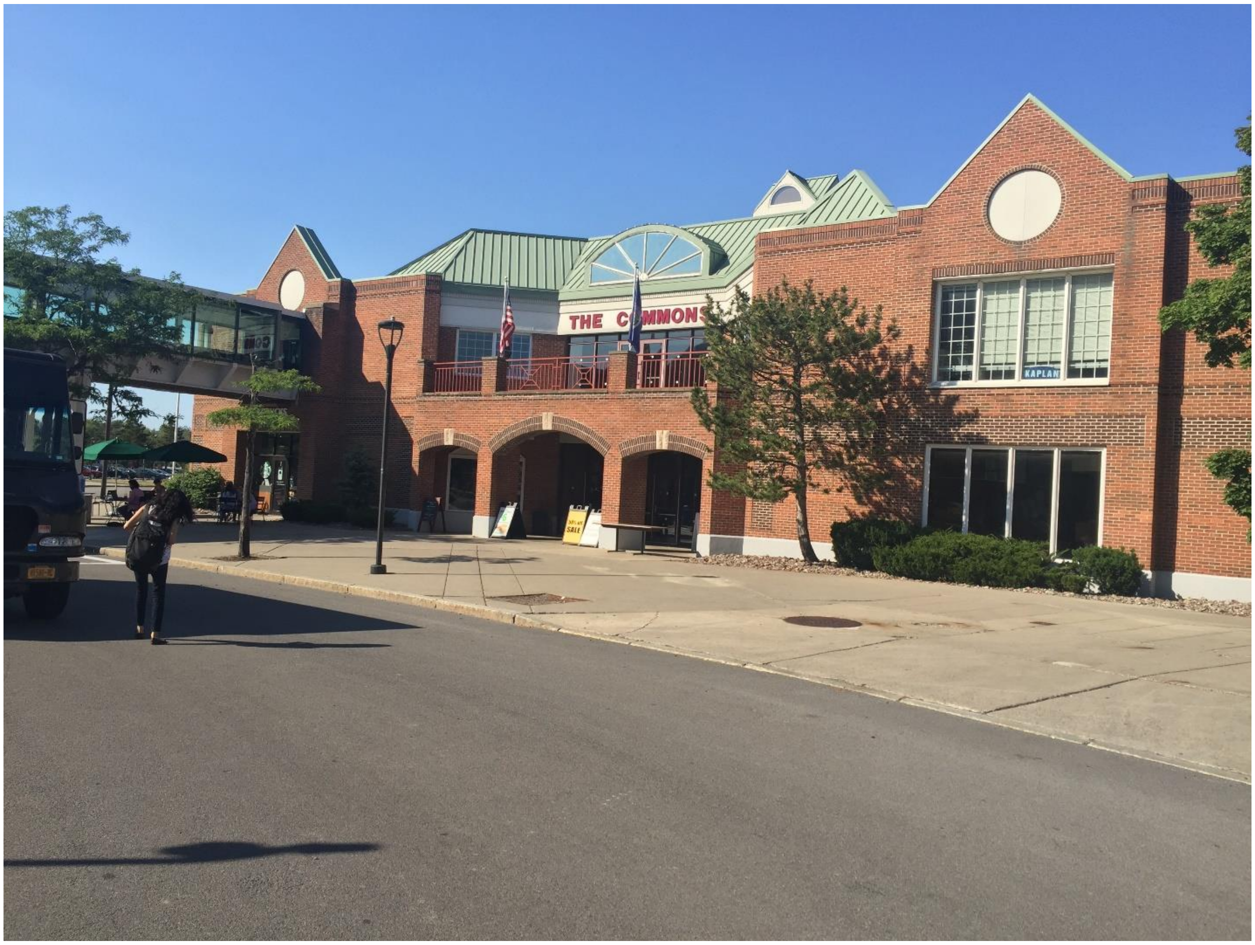
Accuracy

Accuracy when we consider only the top 5 search results: **4/5=0.8**

Accuracy when we consider the top 10 search results: **5/10=0.5**

- ***Commons***

Query Image:



Results Snapshot: Top 10

Knox Lecture Hall

THE COMMONS
SALE

Knox Lecture Hall

THE COMMONS
KAPLAN
SALE

Accuracy

Accuracy when we consider only the top 5 search results: **3/5=0.6**

Accuracy when we consider the top 10 search results: **5/10=0.5**

Total Accuracy of the Search Engine

Accuracy when we consider only the top 5 search results for the overall search engine: **19/25=0.76**

Accuracy when we consider the top 10 search results for the overall search engine: 2**5/50=0.5**

Since all the 5 images similar to the query image in the dataset are part of the top 10 results, accuracy when we consider the top 10 search results for the overall search engine is 1.

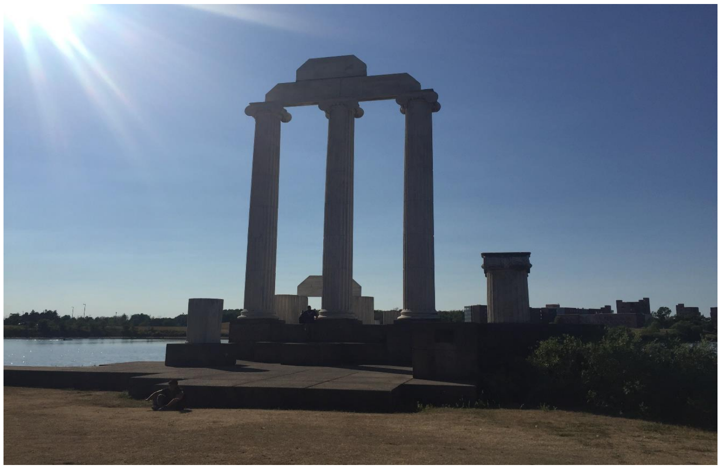
Search Engine Performance

Few of the relevant images come in top 6-10 like in the case of Commons, where 2 relevant images are at 7th and 10th position. It is the same for Lake Lasalle, Norton Hall and Clemens Hall.

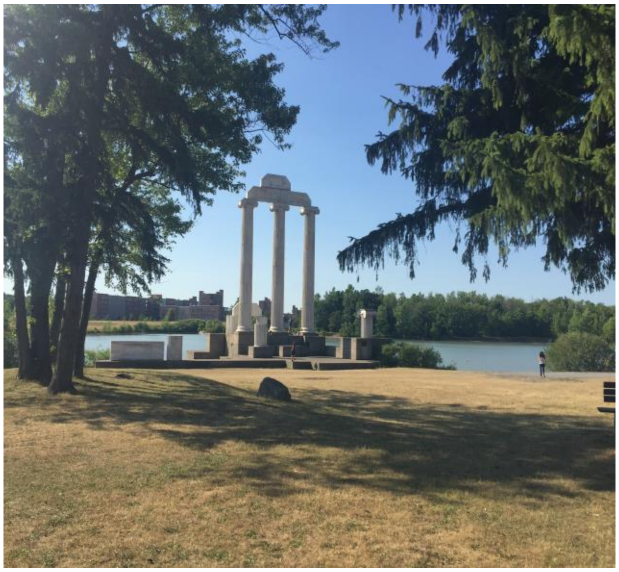
By observing the 2 accuracy measures we can say that this search engine is works well but not accurate enough. This is a small dataset of just 25 images and with an accuracy of only 0.76 for top 5 images. In real world an image dataset will generally have thousands of images in it, then the accuracy of this search engine it going to go down.

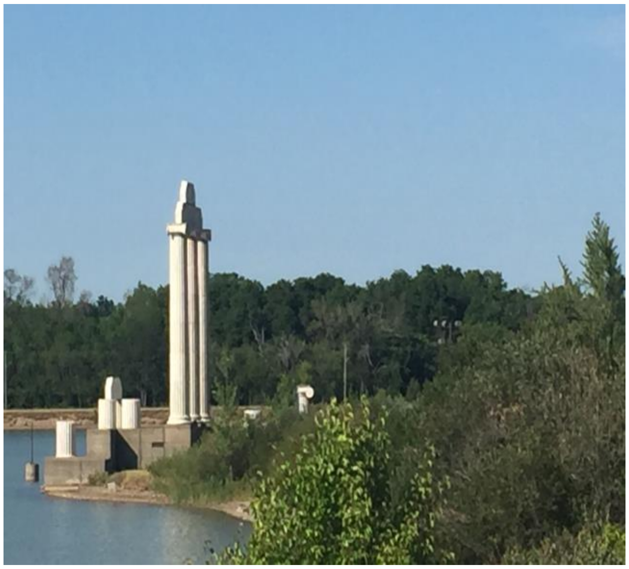
Conclusion

The descriptor used in this search engine is a very basic one. The 3D RGB histogram is sensitive to noise, where even a slight change in brightness or lighting when the image was captured is noted. By replacing 3D RGB histogram with different color space like LAB or HSV can overcome this.



Query Image: Lake Lasalle



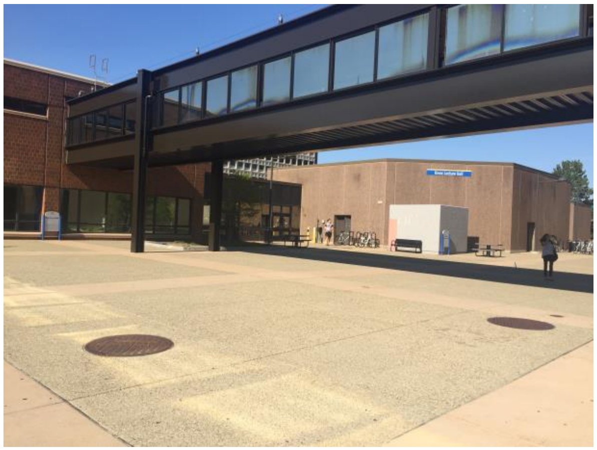


Result: Lake Lasalle 10th position and 2nd position

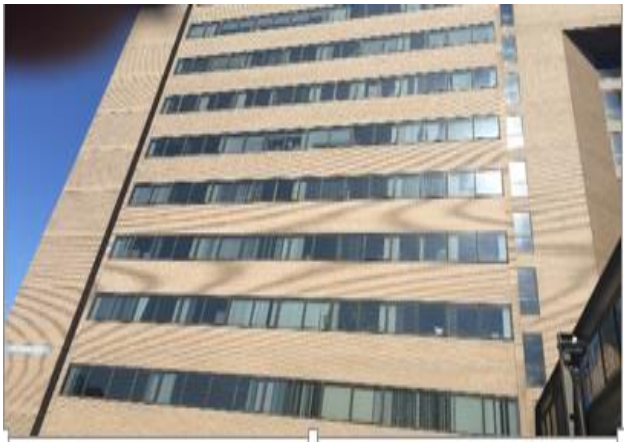
When there is some occlusion in the images, it does not consider them. When we query an image of Lake Lasalle, the following image with few occlusions is at the

10th position though there is a very prominent common feature between the 2 images. If we observe, the image returned in 2nd position has the most prominent feature at an angle.

Also the system takes into consideration only the color distribution in the image, this makes the search engine return any 2 buildings of the same color as similar. Take the case of Norton Hall. Here the image of Clemens hall is at 5th position. This is because there are a lot of common colors between the 2 images.



Query: Norton Hall



Result: Clemens Hall (5th position)

This limitation can be overcome by using other histograms such as Color correlogram, which along with the color considers spatial relationship amongst pixels. Also we can use descriptors which consider the features in the image such as shape, size, texture etc.

The takeaway from this project is that I got a basic idea of how the image search engine is designed. I also learnt about various image indexing descriptors that can be used in the search engines. Also how the similarity between 2 images are calculated.