Landmark Recognition

A Project Report Submitted to the SCIS
in Partial Fulfillment and completion of the AOS course in the Degree of

Master of Technology

in

Artificical Intelligence

By

Pradeep Kumar Jakke

19mcmi33

School of Computer and Information Sciences
University of Hyderabad
Gachibowli, Hyderabad - 500 046
Telangana, India

November , 2019

CERTIFICATE

This is to certify that the Thesis entitled “**LANDMARK RECOGNITION**” submitted by **Pradeep Kumar Jakke** bearing Reg. No. 19MCMI33, in partial fulfillment of the requirements for the award of Master of Technology in Artificial Intelligence is a bonafide work carried out by him under my supervision and guidance.

The Thesis has not been submitted previously in part or in full to this or any other University or Institution for the award of any degree or diploma.

M. Nagamani
Supervisor
School of CIS
University of Hyderabad

Prof. Kavi Narayana Murthy
Dean
School of CIS
University of Hyderabad

ACKNOWLEDGEMENTS

A year long effort, rummaging the trails of research, has finally come to a close. Therefore let me take this opportunity to remember and thank those who made this possible.

This has been a transformative period in my life. A substantial part of it can be attributed to influence of one person - my project guide *M. Nagamani*. I thank hem for all the *gems* I gathered just by being in her vicinity. I am grateful to our Dean, *Prof. Kavi Narayana Murthy* for providing us the required research environment, and especially for allowing 24/7 access to lab facilities. i am also thankful to our M.Tech. coordinator *Dr. Anjeneya Swami Kare* for his help in academic and administrative work.

The thesis in your hand is result of weeks of endless toil. While caffeine was a regular companion through those sleepless nights, there were people willing to extend help.

Let me also remember my friends for all the days we spent together in the labs.

Finally the *three* people in my life without whom this wouldn't have happened in the first place - *my mom, dad and sisters.*

Pradeep Kumar Jakke

Abstract

Recent years have seen an exponential increase in the use of mobile devices. Since many of the mobile devices are equipped with a camera and are connected to the internet, localization in an urban environment using landmark images is gaining popularity. The idea is simple. A tourist takes images of a landmark where he or she is standing with a mobile camera which are then transmitted to a server where the image(s) are matched against a database of landmark images for that locality. If a match is found, relevant information such as background information on the landmark, nearby transit facilities or information on other important landmarks nearby is sent back. This type of application has tremendous potential as a mobile city guide or navigation aid. In this project, we investigate the use of local invariant shape features and global features such as colour and texture for the recognition task as evident from literature and present various retrieval techniques. A variety of descriptors for landmark recognition and scene classification are discussed. Insights into vocabulary building and weighting schemes for representing landmark images are provided that can help in boosting recognition rates.

Contents

List of Figures

List of Tables

1 Introduction

Automatic landmark recognition from images can enable better localization in an urban environment. Since most people nowadays carry mobile devices that are connected to the internet, mobile landmark identification is gaining popularity. The basic idea is a person captures some photos of the place where he or she is standing with a mobile device. These photos are then transmitted to a server over the internet where they are matched against a database of landmarks. If a match is found, background information about the landmark and other relevant information is sent back.

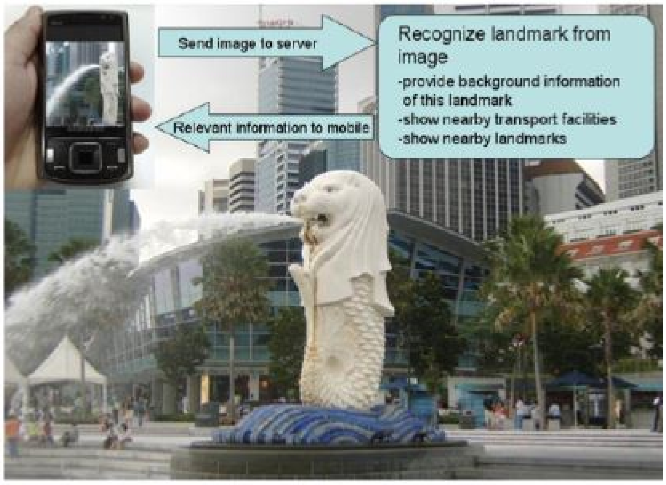

Send image to server
Relevant information to mobile
Recognize landmark from image
-provide background information of this landmark
-show nearby transport facilities
-show nearby landmarks

Figure 1: application of landmark recognition system

This type of application is immensely useful as a mobile city guide. Some of this information can even be overlaid on live camera frames thereby enabling augmented reality. For example, user focuses his camera on some important buildings near by and the names of the buildings pop up as annotations on the camera frame. This can be very useful for navigation in an urban setting as most people use landmarks as an important means of finding their way in a city.

The use of augmented reality based annotation on camera images in order to provide easy landmark-based navigation instructions from start to end point. A similar effort can be observed in, where the authors enable landmark-based navigation between two buildings in a university campus setting.

2 Literature review

we have used various resources for collecting the information about developing the machine learning model and gone through various research papers for identifying the best model that suits our requirements most of the research papers are from microsoft that are published at the image-net conference in 2015 that helped us a lot at coming to the coclusion of our model we have chosen. And also went through various blogs and web articles for finding a way through creating, cleaning and feature detection of the data.

2.1 Data collection

we have created our dataset both training dataset and testing dataset from kaggle which is a an online community of data scientists and machine learners, owned by Google. Kaggle allows users to find and publish data sets, explore and build models in a web-based data-science environment, work with other data scientists and machine learning engineers, and enter competitions to solve data science challenges. Kaggle got its start by offering machine learning competitions and now also offers a public data platform, a cloud-based workbench for data science, and short form AI education.

2.2 Algorithm Design

we have reffered various reasearch papers to compare the performance of the algorithms so as to choose the one with best performance there by came to conclusion to use ResNet(residual network).This ResNet architecture was more successful that traditional, hand-crafted feature learning on the ImageNet. Their DCNN, named AlexNet, contained 8 neural network layers, 5 convolutional and 3 fully-connected. This laid the foundational for the traditional CNN, a convolutional layer followed by an activation function followed by a max pooling operation, (sometimes the pooling operation is omitted to preserve the spatial resolution of the image).

Much of the success of Deep Neural Networks has been accredited to these additional layers. The intuition behind their function is that these layers progressively learn more complex features. The first layer learns edges, the second layer learns shapes, the third layer learns objects, the fourth layer learns eyes, and so on. Despite the popular meme shared in AI communities from the Inception movie stating that “We need to go Deeper”, He et al. [2] empirically show that there is a maximum threshold for depth with the traditional CNN model.

3 Theory

Many mobile applications nowadays use gps location and magnetic orientation of the mobile device to arrive at an estimate of whether a landmark is within viewing volume or not. Figure 2 shows a screenshot from an iPhone application which annotates historic landmarks based on gps information directly on the camera roll.While this may work reasonably well in an open area, the performance may be inferior in an urban setting where the gps reception is usually poor because of high rise buildings. Being able to visually recognize a building can be very advantageous in providing orientation in an urban setting.

A survey of landmark recognition using the bag-of-words framework

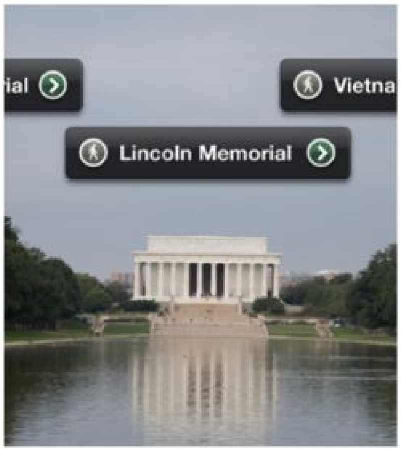


Figure 2: The relative suggestions for landmark

A survey of landmark recognition using the bag-of-words framework 3 Fig. 2 Augmented reality application (Image courtesy: National Park Service, USA).Figure 3 shows a general mobile landmark recognition system. In the training phase, the system performs a number of processing steps on the images in the land-mark database. The rst step involves extracting features from the images and representing them with descriptors. The features may be global or local or a combination of both.

A variety of feature extraction techniques can be found in literature which will be discussed in the next section. Once feature extraction is complete, clustering using k-Means is performed on all features combined in the second step and the cluster centres so obtained are called the visual words and collectively the bag-of-words. The bag-of-words approach, used commonly for text retrieval was rst used by for object and scene recognition. In the third step, each training image is represented as a histogram of visual word occurrences. There are several weighting schemes which we consider in this paper. For query image, the feature extraction and visual word assignment is done first based on the visual word nearest to a feature.Then the query image is represented as a vector using one of the selected weighting schemes.

Using a KNN classifier involves comparing the query vector representation to all the training images and then classifying the query image as belonging to that label to which majority of the k nearest neighbours of the query image vector belong.But this can be computationally expensive as the query image has to be matched with every training sample. An alternative to KNN classifier is to use a multi class support vector machine which uses a collection of binary classifiers based on the one-versus-all or one-versus-one strategy. It builds a model from the training dataset and classifies the query image based on the decision boundary. The classification is near real-time.

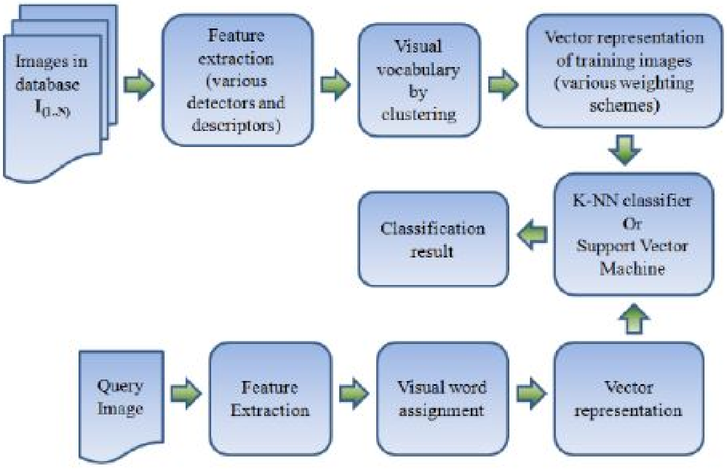

Images in database $I_{(1..N)}$
Feature extraction (various detectors and descriptors)
Visual vocabulary by clustering
Vector representation of training images (various weighting schemes)
K-NN classifier Or Support Vector Machine
Classification result
Query Image
Feature Extraction
Visual word assignment
Vector representation

Figure 3: A landmark recognition system using bag of words

There exist some challenges that need to be overcome by any mobile landmark recognition system. Since the camera on a mobile phone is usually not of high quality, the photos are of low to medium resolution. Secondly, the photos may be taken any time of the day, in very sunny or cloudy or even rainy conditions. So low contrast photos are expected. The recognition system has to be robust against large illumination variations and viewpoint changes. Also, in an urban environment, there will usually be a lot of background clutter. The system needs to well differentiate fore-ground from background.

Another challenge, albeit of a different nature, for landmark recognition is that there exists no readily available list of worldwide landmarks. Queries made on photo sharing websites such as Flickr based on landmark name often return irrelevant photos that must be filtered out. In, the authors have mined landmark photos from photo sharing websites such as picasa.google.com and panoramio.com,Google Image Search and travel guide articles from websites, such as wiki travel.com.They apply visual clustering on the noisy dataset to distinguish true landmarks from false ones. The premise is simple. The true images of a landmark tend to be visually similar and so they will form clusters that have high cohesion.

The rest of this paper is organized as follows. In section 2, information on global and local features and descriptors are presented. Also, feature selection strategies are considered. In section 3, vocabulary building and weighting schemes and their impact on recognition accuracy is discussed. Section 4 presents some popular image retrieval and classification techniques. Finally, section 5 presents some conclusions

3.1 Feature extraction

The first step in recognition as previously mentioned is feature extraction. The visual features used for landmark recognition can be broadly classified as global and local features. Global features are used to represent the entire image. All pixels in the image are considered. Local features usually identify certain interest points or regions in the image and only utilize image properties characterizing those regions or local neighbourhoods around those interest points for recognition. The rest of the pixels are ignored.

3.1.1 Global features

Different global features have been used for landmark recognition. In, the authors use energy spectrum, the squared magnitude of the windowed Fourier trans-form of an image. Given an image I, the discrete Fourier transform can be computed as

$$I(f_x, f_y) = \sum_{x,y=0}^{N-1} I(x,y).h(x,y).e^{-j2\pi(x.f_x+y.f_y)}$$

where h(x,y)is a circular hamming window to reduce the boundary effect. The amplitude component of the above term represents the spatial frequency spread every where in the image. It embeds un localized information about the image structure such as orientation, smoothness, length, and width of the contours that compose the entire scene in the image.

3.1.2 Local Features

Local features are relatively more robust to occlusion and viewpoint changes than global features and are more popular in the context of landmark recognition. The literature on local features is vast. For a detailed survey, the interested reader can refer to [40]. The local features most popularly used can be classified as scale in-variant and affine invariant. In the next two sections, we discuss the scale and affine invariant features

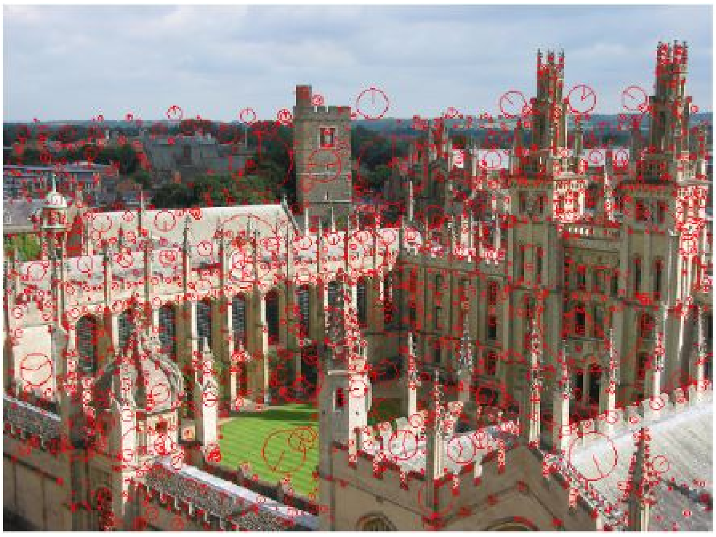


Figure 4: different orientations and scales for a landmark image.

3.2 Feature Descriptors

Feature descriptors compute certain properties of the image in local neighbour hoods centered at key points and are usually represented as a high-dimensional vector.

It measures gradient distribution in the detected regions as a histogram. It is computed as a set of orientation histograms on 4 ×4 pixel neighbourhoods in the gradient image. The contribution of each pixel to the location and orientation bins is weighted by the gradient magnitude.

The quantization of gradient locations and orientations makes the descriptor robust to small geometric distortions and small errors in the region detection. Each orientation histogram contains 8 bins

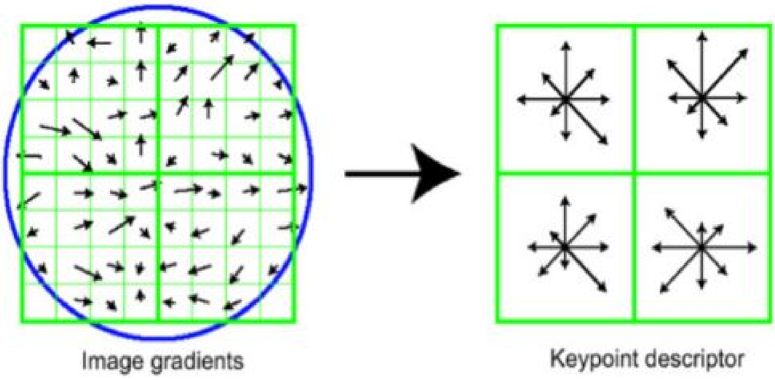


Figure 5: descriptor computation on a 2 ×2 grid with 8 orientation bins

3.3 Feature selection

Feature selection is frequently performed in text retrieval. Yang et al. found out that, when a good criterion is used, up to 98 percent of the unique words in the vocabulary can be removed without loss of text categorization accuracy. It will be interesting to determine if the same proportion of visual words can be eliminated without affecting recognition accuracy. With several thousand features detected per image.

The importance of feature selection for recognition applications cannot be overemphasized.Large number of features require very efcient clustering techniques and typically result in large vocabulary sizes that slow down the recognition performance. Also, a large proportion of features are just background clutter or noise which may confuse the classier. Despite the importance of feature selection, surprisingly few paper shave addresses this issue directly.

It will be interesting to determine if the same proportion of visual words can be eliminated without affecting recognition accuracy. With several thousand features detected per image.Also, a large proportion of features are just background clutter or noise which may confuse the classier. Despite the importance of feature selection, surprisingly few paper shave addresses this issue .

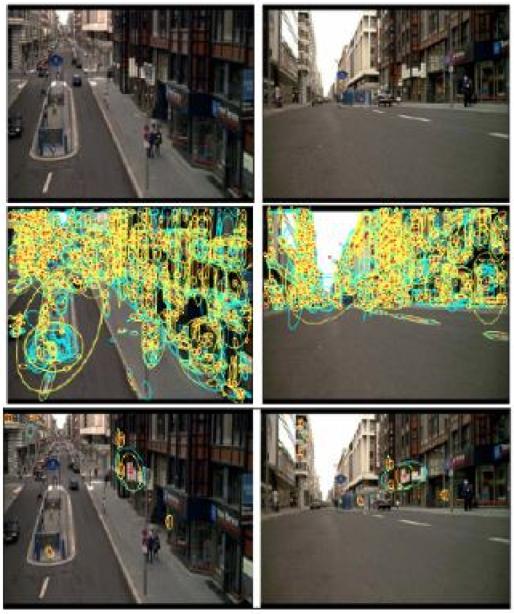


Figure 6: Two frames showing the same scene from very different camera viewpoints

The ve different criteria borrowed from text retrieval to limit the number of features. The rst is document frequency(DF) which is the number of images (documents) in which a visual word appears. Not knowing whether frequent visual words or rare ones are more informative for image classication, they adopt two opposite selection criteria based on DF: DF max chooses visual words with DF above a predened threshold, while DF min chooses visual words with DF below a threshold. The third criterion is the chi-square statistic which measures how strongly a visual word is correlated with a particular classication label.

the same proportion of visual words can be eliminated without affecting recognition accuracy. With several thousand features detected per image.Also, a large proportion of features are just background clutter or noise which may confuse the classier. Despite the importance of feature selection, surprisingly few paper shave addresses this issue.The rst is document frequency(DF) which is the number of images (documents) in which a visual word appears. Not knowing whether frequent visual words or rare ones are more informative for image classication.

Any visual word with a correlation lower than a threshold are eliminated. The fourth and fth measures are mutual information and point wise mutual information which are also statistical measures of the dependence between two random variables.

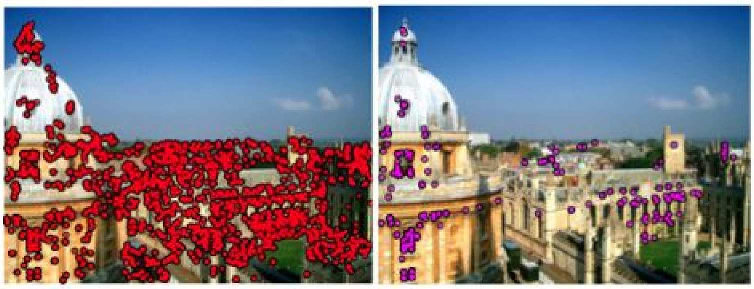


Figure 7: keypoint selection based on geometric verication

observe that when effective criteria like mutual information and chi-square are used, there is only minimum loss of MAP when the vocabulary size is cut by half. When the vocabulary is reduced by 70 percent, the MAP drops merely by 5percent , but after that it drops at a much faster rate. Another interesting observation is that DF max is signicantly better than DF min which implies that frequent visual words widely spread among images are more informative than rare visual words interms of discriminative power

3.4 ResNet

The core idea of ResNet is introducing a so-called "identity shortcut connection" that skips one or more layers, as shown in the following figure

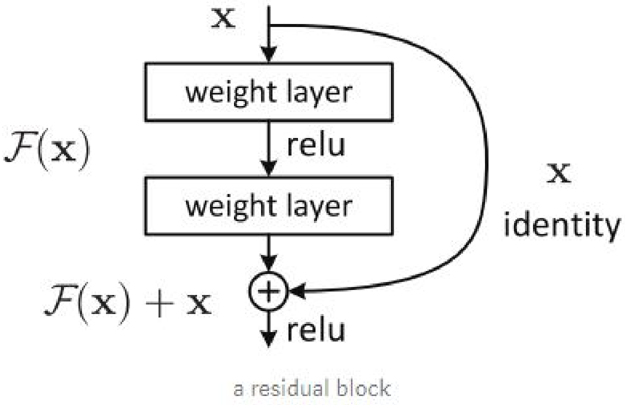

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu
a residual block

Figure 8: Identity shortcut

The stacking layers should not degrade the network performance, because we could simply stack identity mappings (layer that does not do anything) upon the current network, and the resulting architecture would perform the same. This indicates that the deeper model should not produce a training error higher than its shallower counterparts. They hypothesize that letting the stacked layers fit a residual mapping is easier than letting them directly fit the desired underlying mapping. And the residual block above explicitly allows it to do precisely that.

As a matter of fact, ResNet was not the first to make use of shortcut connections, Highway Network introduced gated shortcut connections. These parameterized gates control how much information is allowed to flow across the shortcut. Similar idea can be found in the Long Term Short Memory (LSTM) cell, in which there is a parameterized forget gate that controls how much information will flow to the next time step. Therefore, ResNet can be thought of as a special case of Highway Network.

However, experiments show that Highway Network performs no better than ResNet, which is kind of strange because the solution space of Highway Network contains ResNet, therefore it should perform at least as good as ResNet. This suggests that it is more important to keep these “gradient highways” clear than to go for larger solution space.

4 Conclusion

In this project we have observed that feature selection is an important step in the recognition pipeline that is often ignored or undervalued. Insights into vocabulary building and weighting schemes for representing landmark images are provided that can help in boosting recognition rates. Some effective approaches to landmark classification or finding images are explored which resulted in identifying a best algorithm that provides optimal results in terms of training and performance.

5 Execution

5.1 Generate Dataset

The dataset we used in the project is from kaggle. kaggle provides an API to get the data using which user can get the data from that particular location

```
kaggle competitions download -c landmark-recognition-2019
```

This data set contains the csv files which in turn contains the urls of the images which has to be regenerated along with the features.

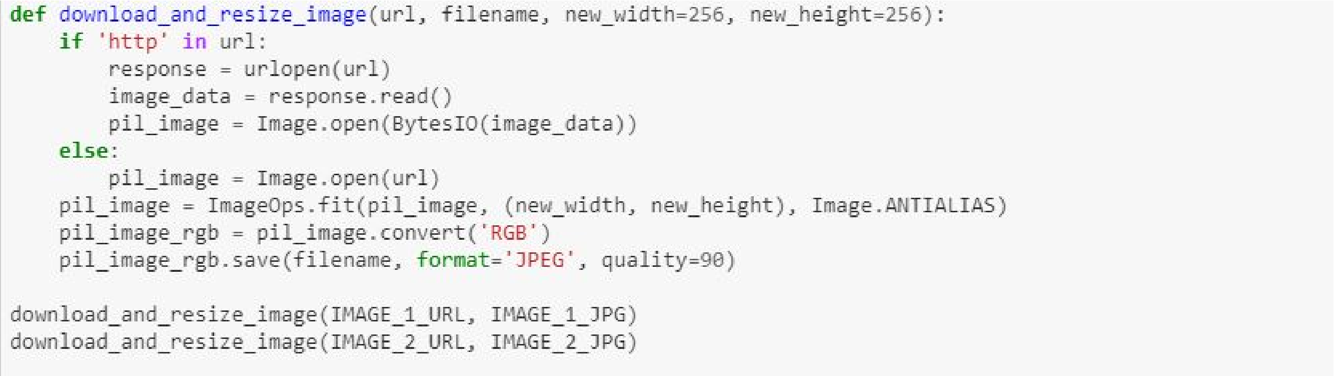
5.2 Sample code

In this we would like to present the sample code of our project.

1. python packages we have used

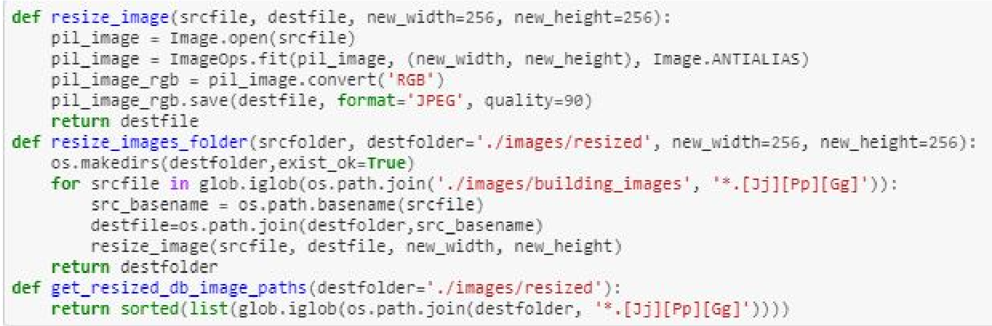
- matplotlib
- numpy
- panda
- pillow
- skimage
- tensorflow
- scipy
- opencv

2. resize all data base images

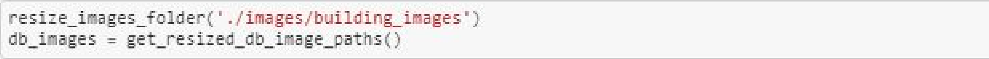


```
def download_and_resize_image(url, filename, new_width=256, new_height=256):
    if 'http' in url:
        response = urlopen(url)
        image_data = response.read()
        pil_image = Image.open(BytesIO(image_data))
    else:
        pil_image = Image.open(url)
    pil_image = ImageOps.fit(pil_image, (new_width, new_height), Image.ANTIALIAS)
    pil_image_rgb = pil_image.convert('RGB')
    pil_image_rgb.save(filename, format='JPEG', quality=90)

download_and_resize_image(IMAGE_1_URL, IMAGE_1_JPG)
download_and_resize_image(IMAGE_2_URL, IMAGE_2_JPG)
```

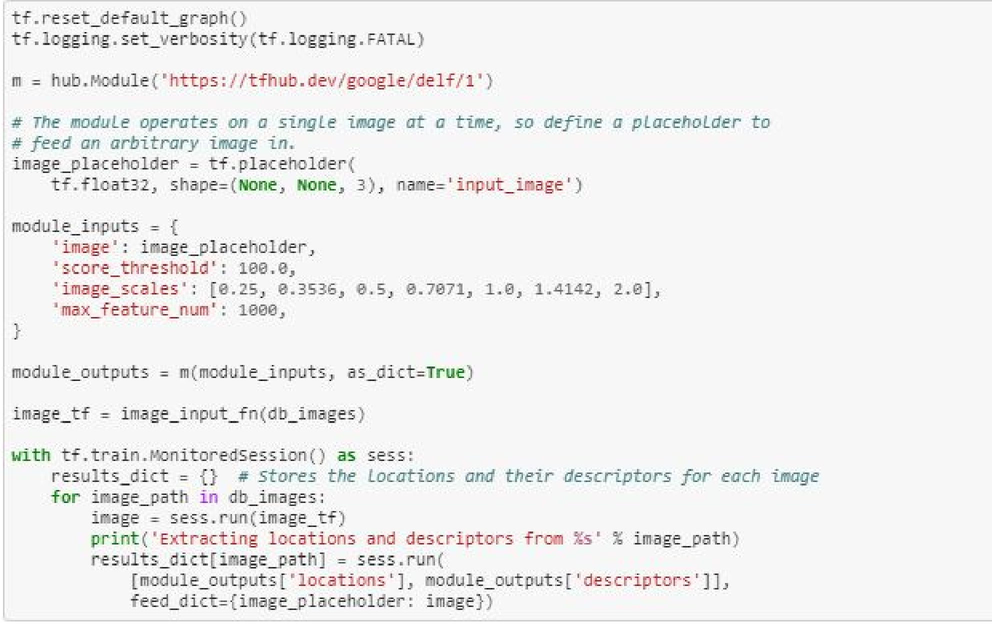
3. evaluate features



```
def resize_image(srcfile, destfile, new_width=256, new_height=256):
    pil_image = Image.open(srcfile)
    pil_image = ImageOps.fit(pil_image, (new_width, new_height), Image.ANTIALIAS)
    pil_image_rgb = pil_image.convert('RGB')
    pil_image_rgb.save(destfile, format='JPEG', quality=90)
    return destfile
def resize_images_folder(srcfolder, destfolder='./images/resized', new_width=256, new_height=256):
    os.makedirs(destfolder,exist_ok=True)
    for srcfile in glob.iglob(os.path.join('./images/building_images', '*.[Jj][Pp][Gg]')):
        src_basename = os.path.basename(srcfile)
        destfile=os.path.join(destfolder,src_basename)
        resize_image(srcfile, destfile, new_width, new_height)
    return destfolder
def get_resized_db_image_paths(destfolder='./images/resized'):
    return sorted(list(glob.iglob(os.path.join(destfolder, '*.[Jj][Pp][Gg]'))))
```

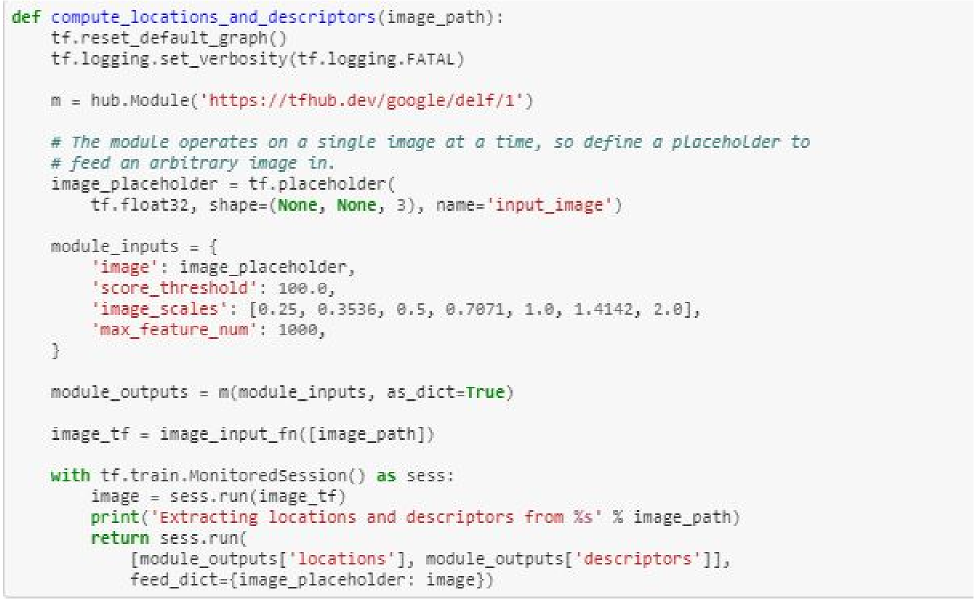


```
resize_images_folder('./images/building_images')
db_images = get_resized_db_image_paths()
```



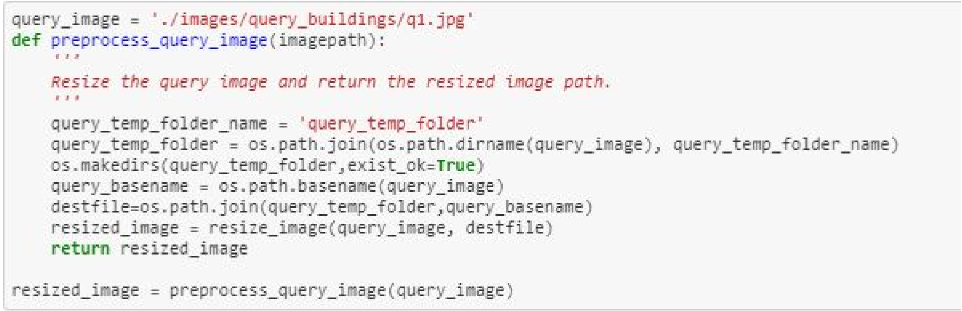
```
tf.reset_default_graph()
tf.logging.set_verbosity(tf.logging.FATAL)

m = hub.Module('https://tfhub.dev/google/delf/1')

# The module operates on a single image at a time, so define a placeholder to
# feed an arbitrary image in.
image_placeholder = tf.placeholder(
    tf.float32, shape=(None, None, 3), name='input_image')

module_inputs = {
    'image': image_placeholder,
    'score_threshold': 100.0,
    'image_scales': [0.25, 0.3536, 0.5, 0.7071, 1.0, 1.4142, 2.0],
    'max_feature_num': 1000,
}

module_outputs = m(module_inputs, as_dict=True)

image_tf = image_input_fn(db_images)

with tf.train.MonitoredSession() as sess:
    results_dict = {}  # Stores the locations and their descriptors for each image
    for image_path in db_images:
        image = sess.run(image_tf)
        print('Extracting locations and descriptors from %s' % image_path)
        results_dict[image_path] = sess.run(
            [module_outputs['locations'], module_outputs['descriptors']],
            feed_dict={image_placeholder: image})
```

4. compute locations and descriptors



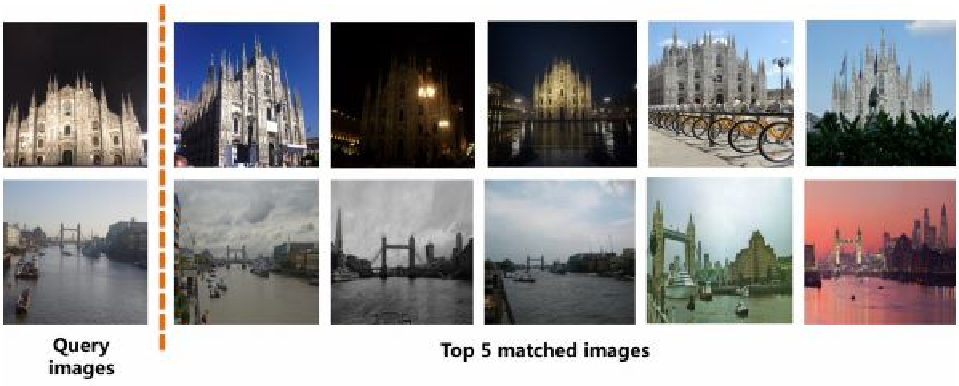
```
def compute_locations_and_descriptors(image_path):
    tf.reset_default_graph()
    tf.logging.set_verbosity(tf.logging.FATAL)

    m = hub.Module('https://tfhub.dev/google/delf/1')

    # The module operates on a single image at a time, so define a placeholder to
    # feed an arbitrary image in.
    image_placeholder = tf.placeholder(
        tf.float32, shape=(None, None, 3), name='input_image')

    module_inputs = {
        'image': image_placeholder,
        'score_threshold': 100.0,
        'image_scales': [0.25, 0.3536, 0.5, 0.7071, 1.0, 1.4142, 2.0],
        'max_feature_num': 1000,
    }

    module_outputs = m(module_inputs, as_dict=True)

    image_tf = image_input_fn([image_path])

    with tf.train.MonitoredSession() as sess:
        image = sess.run(image_tf)
        print('Extracting locations and descriptors from %s' % image_path)
        return sess.run(
            [module_outputs['locations'], module_outputs['descriptors']],
            feed_dict={image_placeholder: image})
```

5. Query the model with image



```
query_image = './images/query_buildings/q1.jpg'
def preprocess_query_image(imagepath):
    '''
    Resize the query image and return the resized image path.
    '''
    query_temp_folder_name = 'query_temp_folder'
    query_temp_folder = os.path.join(os.path.dirname(query_image), query_temp_folder_name)
    os.makedirs(query_temp_folder,exist_ok=True)
    query_basename = os.path.basename(query_image)
    destfile=os.path.join(query_temp_folder,query_basename)
    resized_image = resize_image(query_image, destfile)
    return resized_image

resized_image = preprocess_query_image(query_image)
```

6. output



References

1. Relja Arandjelovic and Andrew Zisserman. Three things ev- eryone should know to improve object retrieval. In 2012 IEEE Conference on Computer Vision and Pattern Recognition, pages 2911–2918. IEEE, 2012.

2. Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In European conference on computer vision, pages 404–417. Springer, 2006.

3. Ondrej Chum, James Philbin, Josef Sivic, Michael Isard, and Andrew Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In 2007 IEEE 11th International Conference on Computer Vision, pages 1–8. IEEE, 2007.

4. Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cedric Bray. Visual categorization with bags of keypoints. In Workshop on statistical learning in computer vision, ECCV, volume 1, pages 1–2. Prague, 2004.

5. Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. arXiv preprint arXiv:1801.07698, 2018.

6. Herve J egou and Ond rej Chum. Negative evidences and cooccurences in image retrieval: The benefit of pca and whitening. In European conference on computer vision, pages 774–787. Springer, 2012