School of Computing Science & Engineering

Course: Content Based Image and Video Retrieval (CSE3018)

Faculty: Dr. Vijayarajan V.

PROJECT REPORT

CONTENT BASED IMAGE RETRIEVAL TO CLASSIFY DOG BREEDS

Group Members:

SREYA DRONAMRAJU	15BCE0971
AKANKSHA LAL	15BCE2087
SHUBHAM	16BCE0808
SAGARIKA PURI	16BCE0798

TABLE OF CONTENTS

1. ABSTRACT

In our project, we used feature extraction of commonly used computer vision features such as HOG, SIFT, GIST and Color, for classifying a set of images. We took the training dataset of various dog images which are labelled by their breed. Most often, we may come across a dog on the street or at someone’s house and wish to identify the breed. In such cases, it would be useful to capture a photo of the dog and the content based image retrieval system would give the classification of the dog breed using the trained dataset that is already given to the system beforehand. The system extracts and compares features and descriptors in similar image locations to be able to identify common characteristics in the same breed of dogs. We have used two dog breeds – Afghan Hound and Appenzeller, which are mostly differentiated by their color, body shape and texture. As it is a small scale project, five training images for each breed were provided with different angles and lighting. Test images from google were provided, and the cbir system used nearest neighbor classifier with the training dataset to predict the class of test images.

2. INTRODUCTION

All image classification systems usually follow a common procedure in which a defined set of features are extracted from a photo and fed to a classifier. These highlights are regularly separated at bland areas or keypoints inside the picture, inspecting both question and foundation, with the expectation that these areas will uncover something about the class. The Content Based Image Retrieval utilizes an image or photo’s matter to look and recover advanced pictures from gigantic database of pictures. Content-based picture recovery frameworks were presented to take care of the issues of content based picture recovery. Content based image recovery is an arrangement of methods for recovering semantically-applicable photos from a database in light of naturally inferred picture highlights.

The image processing is done in sections and is fully parallelized on an isolated machine, and can be easily distributed across multiple machines with a common file system (the standard cluster setup in many universities). The features are extracted in a bag-of-words manner ('color', 'hog2x2', 'hog3x3', 'sift', 'ssim') implemented using Locality-Constrained Linear Coding to allow the use of a linear classifier for speedy training and testing. Content-based Image Retrieval (CBIR), otherwise called query by image content (QBIC) is the utilization of PC vision methods to the picture recovery issue, that is, the issue of hunting down advanced pictures in huge databases. Content-based picture recovery is against conventional idea based methodologies. The system portrayed in earlier works can be used for discovering pictures having specific harmonies and complexities in picture databases, for the most part utilized by architects and craftsmanship students. When we look at two impacts then no one but we can watch the effect of differences in the pictures. At the point when these distinctions accomplish maximal esteems at that point, polar difference comes into picture. We can detect differentiating impacts just by looking at two situations or pictures.

3. LITERATURE SURVEY

It is characterized by (Manjunath, 2002) that, in PC vision, visual descriptors or picture descriptors are characterized as the portrayals of the visual highlights of the substance in pictures, recordings, or calculations or applications that deliver such depictions. They depict basic attributes, for example, the shape, the shading, the surface or the movement, among others. It is portray by (Wu, 2010), that visual descriptors are separated in two principle gatherings: General data descriptors, which they contain low level descriptors which give a depiction about shading, shape, areas, surfaces and movement, and particular space data descriptors which they give data about articles and occasions in the scene.

In [1], an approach to fine-grained image classification is done in which areas from different classes share mutual parts but consist of broad variation in shape and appearance. Dog breed identification was used as a test case to display that extracting related parts boosts classification performance. This field is especially challenging since the appearance of related parts can differ drastically, e.g., the faces of bulldogs and dachshunds are very different. To determine accurate similarities, based geometric and appearance models were constructed of dog breeds and their face parts. Part correspondence allows us to extract and compare descriptors in like image locations. A hierarchy of parts such as face and nose, and breed-specific part localization were also implemented with a 67% recognition rate. It was concluded that accurate part localization clearly expands order execution contrasted with cutting edge approaches. The wide variety of dog breed represents a huge issue to the individuals who might be occupied with getting another canine buddy. Strolling down the road or sitting in a bistro, one may see a neighborly, alluring pooch and stand amazed at its family. By and large, it is difficult to get some information about the breed, and as a rule, the proprietor themselves will be either uncertain or off base in their appraisal. Unless the dog can be categorized as one of a couple of generally referred to and particular breeds, for example, the brilliant retriever, Siberian imposing, or German Shepherd to give some examples, it may demonstrate hard to distinguish one's optimal friend without a lot of research or experience (] O. M. Parkhi, 2012) . The information relating to non-well known pooch breed, be that as it may, isn't accessible in bigger scales. This prevents the usage of convolutional neural systems, because of its prerequisites of substantial measure of picture information for preparing. Exchange adapting along these lines turns into a workable alternative to handle the issue of dog picture arrangement, where the summed up highlights from a pre-prepared model can be utilized to test characterization on related datasets.

A conventional strategy is to utilize distinctive descriptor extraction calculations, and to run a direct classifier on the highlights that are removed. Khosla et. al, who made the Stanford Dogs dataset, could accomplish 22% precision utilizing SIFT descriptors for grouping on Stanford Dogs (A. Khosla, 2011). The current technique for accomplishing the most elevated precision 52% on the Stanford Dogs dataset is by utilizing Selective Pooling Vectors, which encodes descriptors into vectors, and chooses just those that are beneath a specific edge of quantization mistake, regarding the codebook which is used to surmised the nonlinear capacity used to decide the characterization probabilities of different classes (G. Chen, 2015). Another approach is to "restrict" different historic points inside the specific class, and to co-enroll these points of interest and perform correlations on them. Both regulated and unsupervised learning strategies have been connected here. Once more, include extraction strategies, (for example, Filter) can be utilized to limit canine faces previously characterization is performed. Unsupervised learning techniques have been produced to learn "format" shape designs which normally re-happen in all pictures being sorted, and have possessed the capacity to appear to 38% exactness on the Stanford Dogs dataset.

4. PROPOSED METHODOLOGY

We are to use various feature extraction methods to classify a set of images. For this purpose, any set of images can be used as dataset. We have chosen dataset of dog images as we can aim to classify dogs by their breed. Dogs are the most photographed creatures after human beings, and as important as it is to detect and identify humans, it is as easy to do so for dogs. Dogs are characterized by their unique features such as color, fur, patches on the skin, shape of the ears or nose, size, etc. We are going to train the system with a set of images of dogs of specifically chosen breeds with their breed as the label. The features in the images are to be extracted using bag-of-words manner and compressed utilizing linear coding to permit linear classifier for speedy training and testing. We use Matlab to implement our content based image retrieval system, though it can also be used in Octave, with some minor compatibility problems that may arise. The datasets_feature function could be executed on numerous machines simultaneously to fasten feature extraction. The function takes control of the entire pipeline of piling up a dictionary (for bag-of-words features), compressing features to the dictionary, and concatenating them together to form a spatial pyramid. We could use a single or many datasets as explained. A new different folder will be created for every set of data and a different dictionary will be grasped for the dataset. We include various features in the datasets_feature function such as color, gist, hog2x2, hog3x3, lbp, sift, and ssim.

DESCRIPTION OF FEATURES

Color: Convert the image to color nomenclature [7,8] and extract thick covering patches of different sizes in the shape of a histogram of different names of colors. Then the bag-of-words and spatial pyramid pipeline is to be applied.

Gist: GIST descriptor encompasses the spatial outline of the image [9]

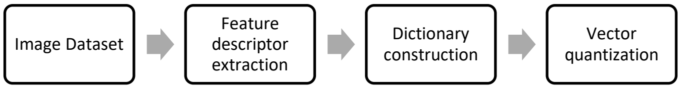
Dense HOG2x2, HOG3x3: Extract HOG [10] in an intensified manner on a grid [11] and join 2x2 or 3x3 cells to obtain a descriptor at every grid location. Then the bag-of-words and spatial pyramid pipeline are applied.

LBP: Extract non-uniform Local Binary Pattern [12] descriptor, and join 3 stages of spatial pyramid to gain resulting feature vector.

Dense SIFT: Extract SIFT [13] descriptor in a intensified manner on a grid in various patches, and then implement the bag-of-words and spatial pyramid pipeline.

SSIM: Extract Self-Similarity Image Matching [14] descriptor in an intensified way and we then use the bag-of-words and spatial pyramid pipeline to gain resulting feature vector.

BAG-OF-WORDS MODEL



IMPLEMENTATION

- First we specify name of datasets
- Specify lists of train images
- Specify lists of test images
- Specify feature to use
- Load the config structure
- Perform feature extraction
- Load train features
- Load test features

SAMPLE CODE TO EXTRACT COLOR FEATURES

```
addpath(genpath(pwd));
% Initialize variables for calling datasets_feature function
info = load('images/filelist.mat');
datasets = {'demo'};
train_lists = {info.train_list};
test_lists = {info.test_list};
feature = 'color';
% Load the configuration and set dictionary size to 20 (for fast demo)
c = conf();
c.feature_config.(feature).dictionary_size=20;
% Compute train and test features
datasets_feature(datasets, train_lists, test_lists, feature, c);
% Load train and test features
train_features = load_feature(datasets{1}, feature, 'train', c);
test_features = load_feature(datasets{1}, feature, 'test', c);
```

```
% Below is a simple nearest-neighbor classifier
% The images have a border color of black and white to indicate the two
% different classes in the demo dataset.
% Display train images in Figure 1
train_labels = info.train_labels; classes = info.classes;
unique_labels = unique(train_labels);
numPerClass = max(histc(train_labels, unique_labels));
h = figure(1); set(h, 'name', 'Train Images'); border = 10;
for i=1:length(unique_labels)
    idx = find(train_labels==unique_labels(i));
    for j=1:length(idx)
        subplot(length(unique_labels), numPerClass, j+(i-1)*numPerClass);
        im = imread(train_lists{1}{idx(j)});
        im = padarray(im, [border border], 255*(i-1)/(length(unique_labels)-
1)); imshow(im);
        title(sprintf('Example: %d, Class: %s', j,
classes{unique_labels(i)}));
    end
end
% Display test images and nearest neighbor from train images in Figure 2
test_labels = info.test_labels; classes = info.classes;
numPerClass = max(histc(test_labels, unique_labels));
h = figure(2); set(h, 'name', 'Test Images'); border = 10;
[~, nn_idx] = min(sp_dist2(train_features, test_features));
for i=1:length(unique_labels)
    idx = find(test_labels==unique_labels(i));
    for j=1:length(idx)
        subplot(length(unique_labels), numPerClass*2, 2*(j-1)+1+(i-
1)*numPerClass*2);
        im = imread(test_lists{1}{idx(j)});
        im = padarray(im, [border border], 255*(i-1)/(length(unique_labels)-
1));
        imshow(im);
        title(sprintf('Example: %d, Class: %s', j,
classes{unique_labels(i)}));

        subplot(length(unique_labels), numPerClass*2, 2*(j-1)+2+(i-
1)*numPerClass*2);
        im = imread(train_lists{1}{nn_idx(idx(j))});
        im = padarray(im, [border border],
255*(train_labels(nn_idx(idx(j)))-1)/(length(unique_labels)-1));
```

```
        imshow(im); title(sprintf('Nearest neighbor, predicted class: %s',
classes{train_labels(nn_idx(idx(j)))}));
    end
end
```

5. RESULT ANALYSIS

We have provided different images of two dog breeds – Afghan hound and Appenzeller. The Afghan hound is characterised by the long hair, body shape and fur color. The Appenzelleris characterised by its black color and characteristic white or lighter patches on the nose area. We have given test images which contain other objects in the background and different angles. We can see that nearest neighbor classifier used the features from training dataset and classified the images with the dog breed correctly. The first figure [Fig 2] shows the different images for training set for Afghan Hound. The second figure [Fig 3] shows the images provided for training set for Appenzeller. In the third and fourth figures [Figs 4,5] we supply test images from google, which are not pre labelled with the breed. The system successfully matches the features of the test images with the training images and classifies the dog breed as 'Afghan Hound' or 'Appenzeller' by predicting the nearest class neighbor.

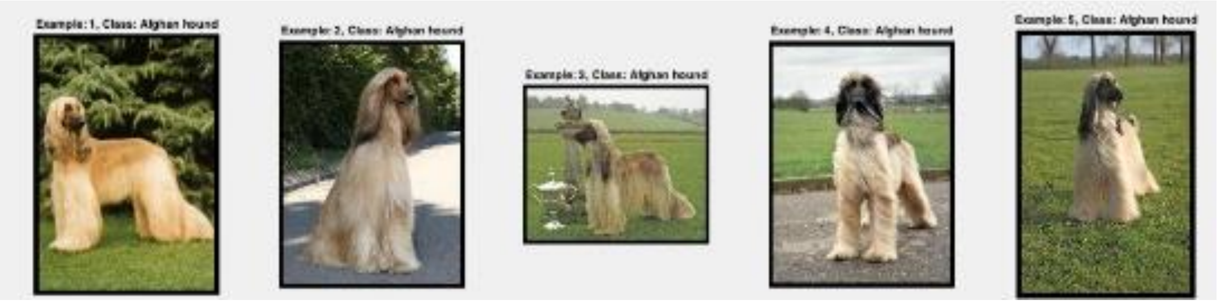

Example: 1, Class: Afghan hound
Example: 2, Class: Afghan hound
Example: 3, Class: Afghan hound
Example: 4, Class: Afghan hound
Example: 5, Class: Afghan hound

Figure 1 Training Set for Afghan Hound

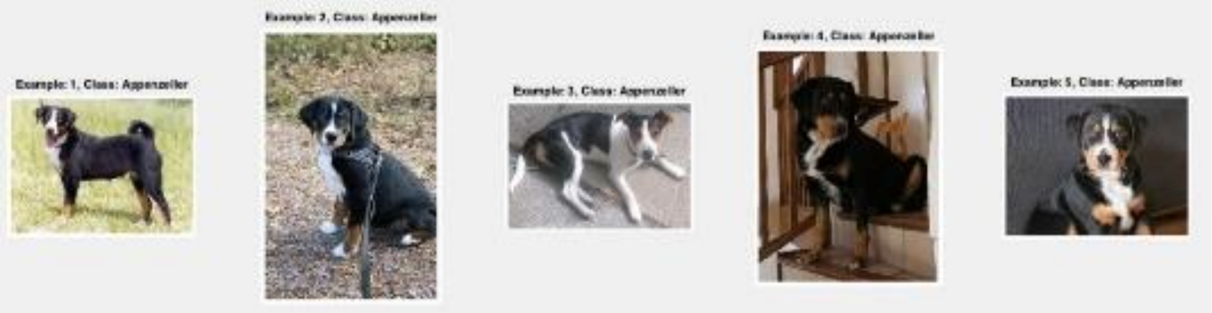

Example: 1, Class: Appenzeller
Example: 2, Class: Appenzeller
Example: 3, Class: Appenzeller
Example: 4, Class: Appenzeller
Example: 5, Class: Appenzeller

Figure 2 Training Set for Appenzeller

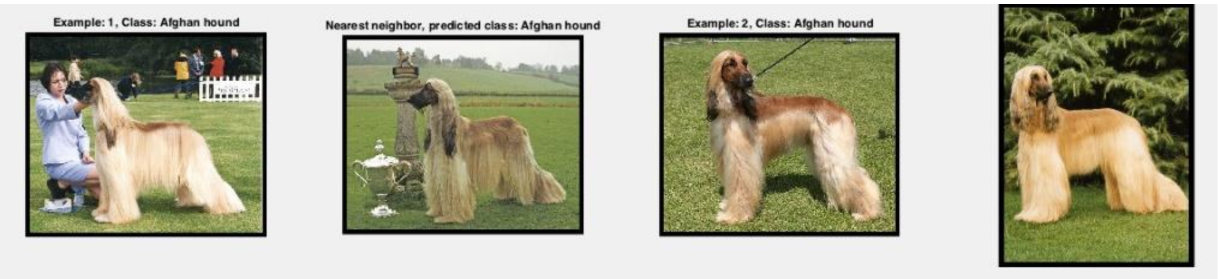

Example: 1, Class: Afghan hound
Nearest neighbor, predicted class: Afghan hound
Example: 2, Class: Afghan hound

Figure 3 Testing Set for Afghan Hound

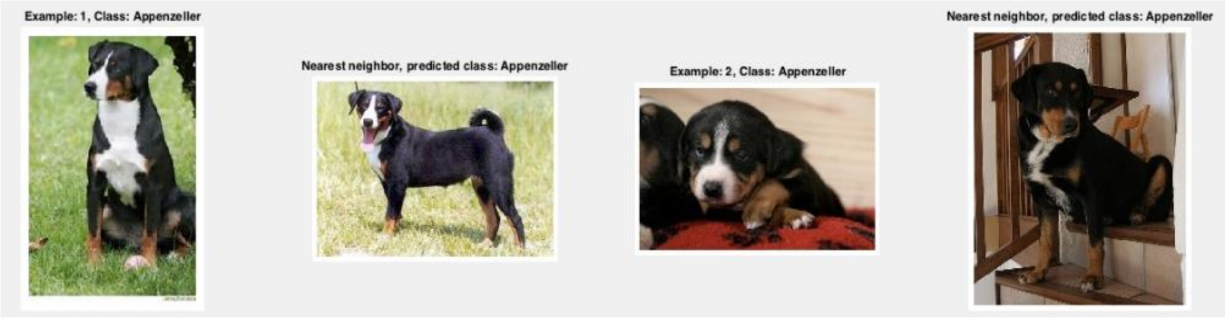

Example: 1, Class: Appenzeller
Nearest neighbor, predicted class: Appenzeller
Example: 2, Class: Appenzeller
Nearest neighbor, predicted class: Appenzeller

Figure 4 Testing Set for Appenzeller

6. CONCLUSION

In this project, we implemented various computer vision feature extraction methods to classify a set of images. We were able to classify the breed of a dog in a provided image with the help of various types of dog images of that same breed that were stored as training set. We demonstrated an application of color feature extraction where we can see that the nearest neighbor classifier utilized the features from training dataset and predicted the images with the dog breed correctly. In future, we wish to implement these feature extraction methods on a larger scale for a database of more than 10,000 images and hundreds of dog breeds. We could also use Ensemble learning, which combines the outputs of multiple classification models and compare the results. The current system focuses on the entire image of the dog, however more accuracy could be achieved if we focused on only the facial features an divided them into individual small regions with feature extraction methods separately for the eyes, nose, ears, etc. The same algorithm and code that we used for this project can also be implemented for an art classification system which categorizes art images into their type- such as abstract, traditional, oil painting, water color painting. We can use the same concept of color feature extraction to predict the art form.

7. REFERENCES

[1] Liu J., Kanazawa A., Jacobs D., Belhumeur P. (2012) Dog Breed Classification Using Part Localization. In: Fitzgibbon A., Lazebnik S., Perona P., Sato Y., Schmid C. (eds) Computer Vision – ECCV 2012. ECCV 2012. Lecture Notes in Computer Science, vol 7572. Springer, Berlin, Heidelberg

[2] Manjunath, B. ., Salembier, P., & Sikora, T. (2002). Visual Descriptors. Wiley and Sons.

[3] Wu, J., & M, J. (2010). CENTRIST: A Visual Descriptor for Scene Categorization. IEEE Transactions on Pattern Analysis and Ma, 33(8), 1489 – 1501.

[4] O. M. Parkhi, A. Vedaldi, A. Zisserman and C. V. Jawahar, "Cats and dogs," 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, 2012, pp. 3498-3505. doi: 10.1109/CVPR.2012.6248092

[5] A. Khosla, N. Jayadevaprakash, B. Yao and L. Fei-Fei. “Novel dataset for Fine-Grained Image Categorization”. First Workshop on Fine-Grained Visual Categorization (FGVC), IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011

[6] G. Chen, J. Yang, H. Jin, E. Shechtman, J. Brandt, and T. Han, “Selective Pooling Vector for Fine-Grained Recognition”, Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on. IEEE, 2015.

[7] J. van de Weijer, C. Schmid, J. Verbeek Learning Color Names from Real-World Images, CVPR 2007

[8] R. Khan, J. van de Weijer, F. Khan, D. Muselet, C. Ducottet, C. Barat, Discriminative Color Descriptors, CVPR 2013

[9] A. Oliva, A. Torralba, Modeling the shape of the scene: a holistic representation of the spatial envelope, IJCV 2001

[10] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, CVPR 2005

[11] B. C. Russell, A. Torralba, K. P. Murphy, W. T. Freeman, LabelMe: a database and web-based tool for image annotation, IJCV 2008

[12] T. Ojala, M. Pietikäinen, T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with Local Binary Patterns, PAMI 2002

[13] D. Lowe, Distinctive image features from scale-invariant keypoints, IJCV 2004

[14] E. Shechtman, M. Irani, Matching Local Self-Similarities across Images and Videos, CVPR 2007