Group Project 1: VLAD vs BoVW

Brahma Kulkarni (IMT2017011)
Kaustubh Nair (IMT2017025)
Sarthak Khoche (IMT2017038)
Vaishnavi Dhulipalla (IMT2017514)

29 March 2020

1 Aim

The aim of this exercise is to compare the Bag of **Visual Words (BoVW)** and **Vector of Logically Aggregated Descriptors (VLAD)** approaches.

2 Dataset

For both approaches, we used the CIFAR10 dataset. This dataset has 50,000 train and 10,000 test images. The train images are split into 5 batches. Initially, we wrote our own programs to load the CIFAR10 dataset (load_cfar10_train() and load_cfar10_test() using pickle to load the batch files). However, as we needed greater resources, we shifted to Google Colaboratory and directly loaded the dataset using the 'keras.datasets' module.
Here's a link to the dataset: https://www.cs.toronto.edu/~kriz/cifar.html

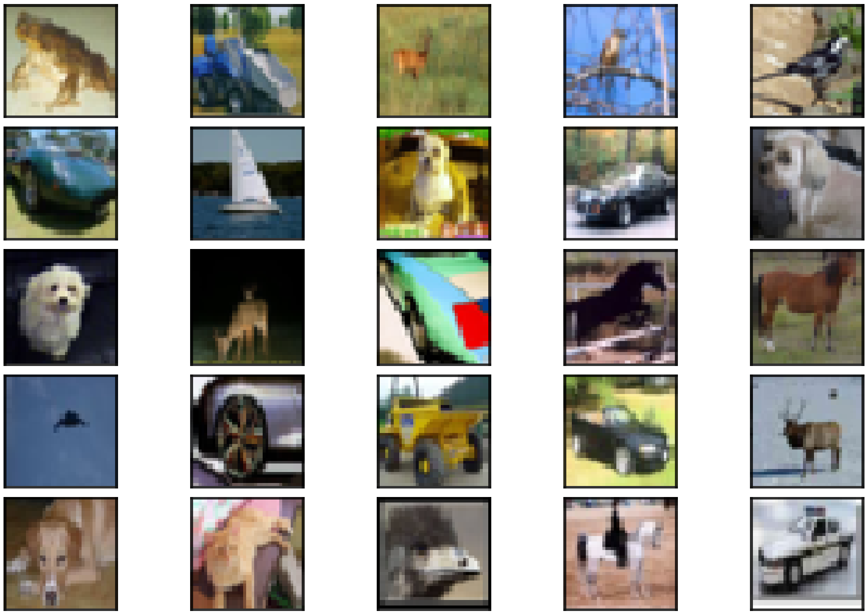

Figure 1: Dataset

3 Pre-processing

We converted every image to gray scale and reshaped them to the size of 150 × 150. INTER_AREA interpolation is used, which resamples using pixel-area relations.

4 Bag of Visual Words (BoVW) approach

In this approach, for each image, we compute the SIFT descriptors and run the KMeans clustering algorithm on them. This gives us labels and cluster centers for those descriptors. We generate a histogram on the descriptors of each image and train a model based on all of the histograms generated. Hence, we generate a model based on the frequencies of occurrence of the labels of the descriptors of the images.

Steps:

- **Generating SIFT feature descriptors**: The functions used were **extractSIFTIndividual()** and **extractSIFTFeatures()**.
- **Building a KMeans model on the SIFT descriptors**: The functions used was **kMeans()** and this used the MiniBatchKMeans function from 'sklearn.cluster' module.
- **Generating frequency histograms:** The function used was **aggregateDescriptors()**.
- **Using PCA to reduce dimensionality.**
- **Training a model and generating predictions:** For this purpose we used the **SVM model** (imported from 'sklearn' library).

Result

Having used the above approach, we obtained an average accuracy of **19.71%** on the aforementioned CIFAR10 dataset.

5 Vector of Logically Aggregated Descriptors (VLAD) approach

To start off this approach, it is the same as BoVW. The SIFT descriptors are computed and KMeans clustering is run on them. Now, for each image, each descriptor is taken and the closest cluster center to it is subtracted from it. Now all of these differences are added up (aggrgated) for each image and these form our new VLAD features.

Advantage over BoVW approach

In the VLAD approach, as opposed to the BoVW approach, we wish to learn the difference between descriptors and their corresponding cluster centers. In other words, we can think of it as learning the distance of descriptors from their corresponding cluster centers. This gives the model that we're training a higher discriminating power.

Steps:

- **Generating SIFT feature descriptors**: The functions used were **extractSIFTIndividual()** and **extractSIFTFeatures()**.
- **Building a KMeans model on the SIFT descriptors**: The functions used was **kMeans()** and this used the MiniBatchKMeans function from 'sklearn.cluster' module.
- **Generating the VLADs:** The function used was **aggregateDescriptors()**.
- **Using PCA to reduce dimensionality.**
- **Training a model and generating predictions:** For this purpose we used the **SVM model** (imported from 'sklearn' library).

Result

Having used the above approach, we obtained an average accuracy of **45.2%** on the aforementioned CIFAR10 dataset.

6 Difficulties faced

- We noticed that, while generating SIFT features, some descriptors were empty (or of None type). There were constantly 6 images had this problem. To tackle this problem we returned a list of labels corresponding to only those descriptors that weren't of type None and returned this with the feature vector. After which, this new list of labels was used for training and predicting.
- This exercise was computationally quite intensive. To tackle this problem, we migrated the code to Google Colaboratory and only worked when we were alloted 25.5 gb of RAM and 68.4 gb of disk space.

7 Conclusion

As you can see, using the same SVM model, we got accuracies of 19.71% using the BoVW approach and 45.2% using the VLAD approach. Hence, we can conclude that the VLAD approach is better as it clearly provides more promising results on, in this case, the CIFAR10 dataset.

8 References

BoVW

- https://towardsdatascience.com/bag-of-visual-words-in-a-nutshell-9ceea97ce0fb

VLAD

- **Paper:** https://github.com/ameya005/VLAD-Implementation
- **Tutorial:** https://github.com/ameya005/VLAD-Implementation
- **Implementations:**
  - https://github.com/ameya005/VLAD-Implementation
  - https://github.com/jorjasso/VLAD
  - https://github.com/lixuan0023/VLAD