| | | |
|---|---|---|
| **Spring 2018** | **Report#3 - 01/21/2018** | **Tianxiao Zhang** |

## I. Tasks achieved Last Week (***, **, *: order of priority)

■ Project: Content-Adaptive Saak Transform

Purpose: Using content-adaptive method to further improve the accuracy of the MNIST test data.

Method: Using clustering method to cluster the content and doing Saak transform in each cluster and then combine them together as feature for each image.

Note: The time cost for running the codes should be controlled because what we want to do is beat CNN, not just running the codes without considering the time. If the time cost is long, the method is not that good.

Slogan: Beat CNN!

## II. Feedback and Interaction

■ ***Prof. Kuo's feedback***

•

■ ***Discussion***

## III. Report

### 1. What The “Content” Should Be?

In this week, I clustered the 2-by-2 using the mean-shift clustering method. First, I clustered the 2-by-2 blocks using 0.1% of the total number of blocks and then predicted the other blocks based on the clustering. I got really bad results because the number of data used to be clustered seemed not enough. But we need to control the time at the same time. Then I thought a method that we could squeeze the total 2-by-2 blocks. There are plenty of same blocks in the total blocks such as all black one and all white one. These repetitions seemed to contribute nothing to the clustering. So what I was doing was only use one of each kind of block no matter how many times it appeared. Thus the number of samples reduced a lot. Then I can utilize 1% of the number of blocks to cluster and then predict the other blocks. By the way, using 1% of the first total blocks was much better than using 1% of random total blocks. Many experiments proved this.

| Percentage of total blocks | 1% | 2% | 3% | 4% | 5% |
|---|---|---|---|---|---|
| Train accuracy | 0.9923 | 0.9943 | 0.9951 | 0.9950 | 0.9954 |
| Test accuracy | 0.9715 | 0.9293 | 0.9333 | 0.9446 | 0.9637 |
| Total time | 8’59” | 11’6” | 16’58” | 25’44” | 41’10” |

We can obviously see that the can get the best test accuracy result using 1% of the total blocks. Even though we can get better and better results when increasing the percentage of the total blocks from 2%, the time cost increased rapidly and the results we got seemed still not good enough compared to the result we can get using the original Saak transform. So it seemed not a good way to do adaptive Saak transform.

Why we cannot get a much better results by clustering the 2-by-2 blocks? The reason I though was shown below:

In the above experiment, what we needed to cluster was 2-by-2 blocks. However, every image has 2-by-2 white block and moreover, each image also has 2-by-2 black block. If we cluster all white blocks together and all black blocks together, we get the same coefficients where the white blocks locate in each image and the same coefficients where the black blocks locate in each image too. So this seems no difference with no-clustering Saak transform because finally we need to fit the images to labels instead of blocks to labels. That’s why it did not work.

What about 4-by-4 blocks? Clustering 4-by-4 blocks, we should first resize the 32-by-32 images to 64-by-64 images based on the property of Saak transform. However, the results were still not good because 4-by-4 Saak transform only has 3 stages which were not enough because we lost much information with less stages.

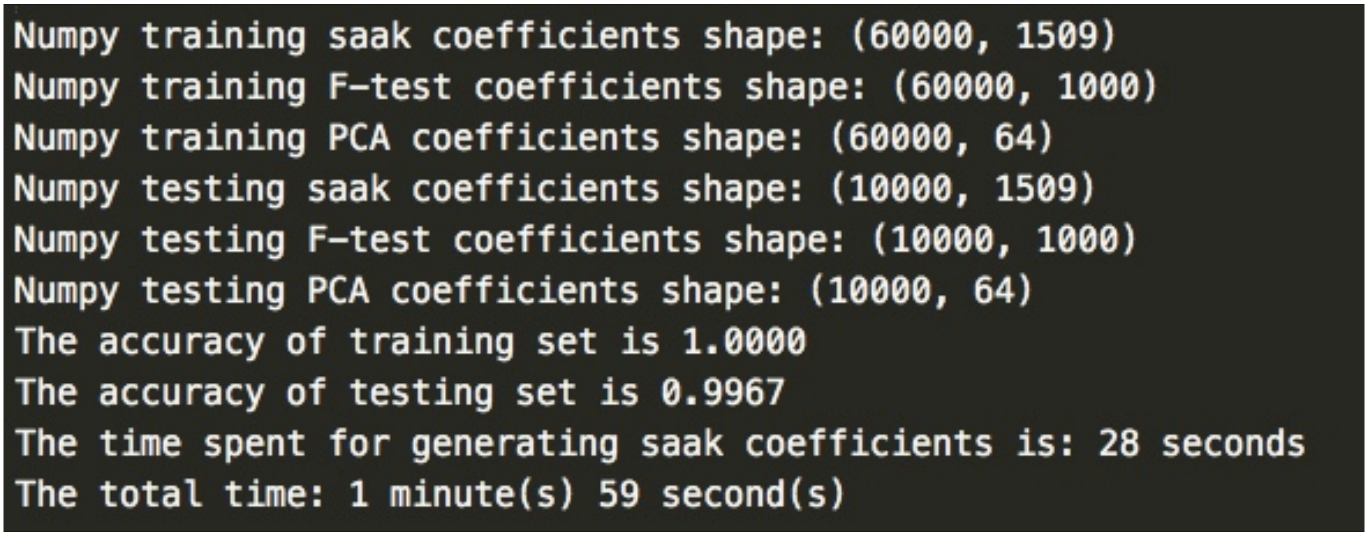
Saak transform is still in the area of “deep learning”, if CNN can improve the results by adding more layers, what will happen if we increase one more stage in Saak transform? I can improve the test accuracy by about 0.09% than the original Saak transform by resizing the images to 64-by-64 with 2-by-2 blocks and no clustering. By doing this, we can increase one more stage. But the improvement seemed not that meaningful for its trivial value.

## 2. Image-Adaptive Saak Transform?

What will happen if we change the “content” from blocks to images?

Then I did a very interesting experiment in which I utilized the labels to cluster the images.

I clustered the train and test images based their labels. For instance, all images labeled “0” were clustered together and so on. And finally I still fitted train images to train labels and utilized test images as test.

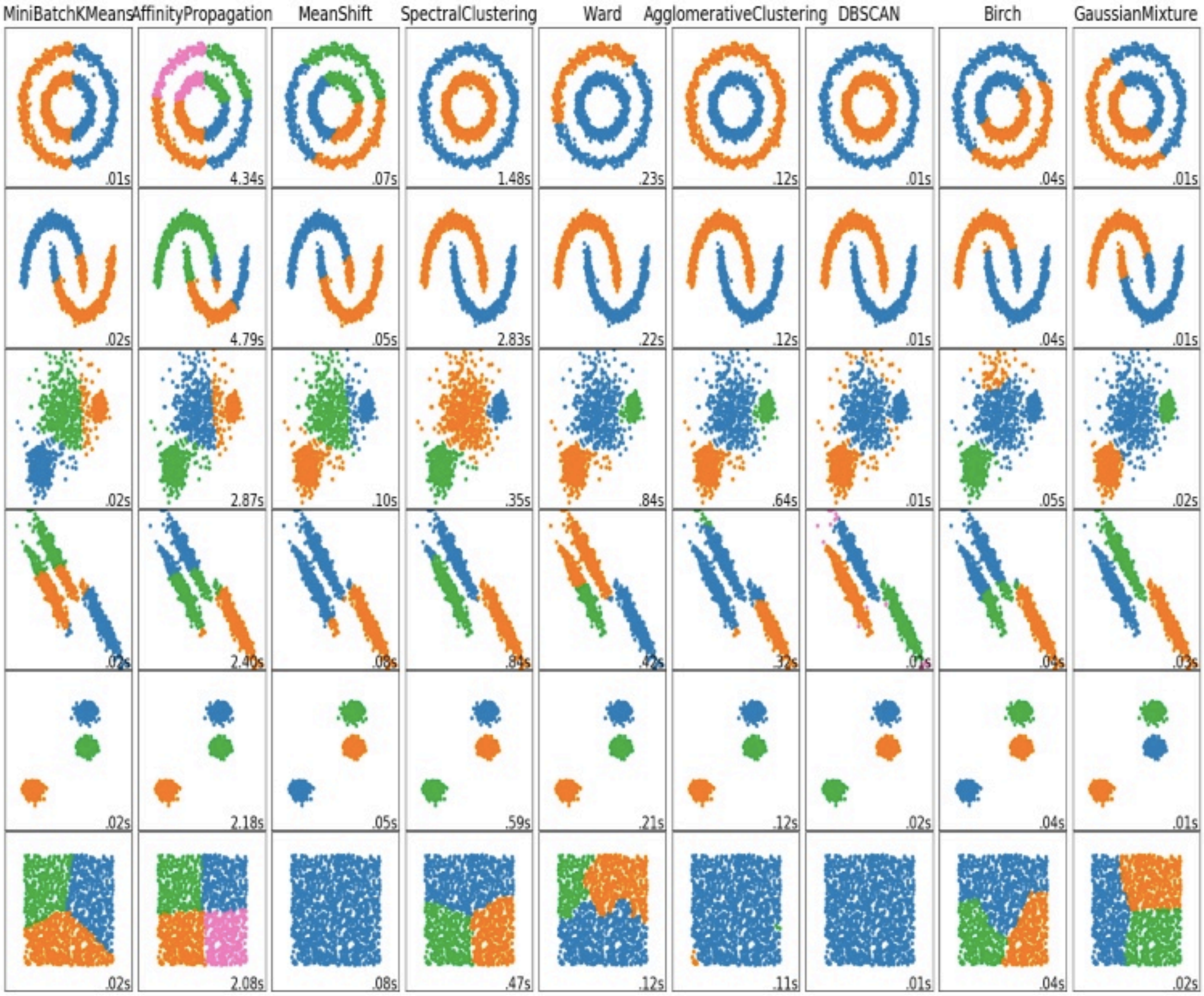


```
Numpy training saak coefficients shape: (60000, 1509)
Numpy training F-test coefficients shape: (60000, 1000)
Numpy training PCA coefficients shape: (60000, 64)
Numpy testing saak coefficients shape: (10000, 1509)
Numpy testing F-test coefficients shape: (10000, 1000)
Numpy testing PCA coefficients shape: (10000, 64)
The accuracy of training set is 1.0000
The accuracy of testing set is 0.9967
The time spent for generating saak coefficients is: 28 seconds
The total time: 1 minute(s) 59 second(s)
```

The results are truly great and we can get 100% accuracy for train images and 99.67% accuracy for test images. However, this was just a test because we can use train labels to cluster the train images but we cannot use test labels to cluster test images. So the results above are like entropy coding because we can find a coding method to be close or equal to the entropy, but we cannot exceed the entropy because the entropy is the value in theory.

## 3. Future Work

I think how to find a clustering method to get the clustering close to the clustering using labels is the key point for future work. I am still trying but it seemed that k-means and mean-shift are both not that good to do that.



Above shows the properties of various clustering method. I am still testing these methods to see which one is the best to cluster the images.

## 4. Conclusion

What we are doing is interesting and meaningful because no one did this before. We have almost no references to refer and what we need to do is thinking and exploring. Even Yann LeCun said that “deep learning is dead”. So in this post-CNN era, let’s contribute as much as we can to deep learning.

## References

[1] C.-C. Jay Kuo, “Understanding convolutional neural networks with a mathematical model,” the Journal of Visual Communications and Image Representation, Vol. 41, pp. 406-413, November 2016.

[2] C.-C. Jay Kuo, “The CNN as guided multi-layer RECOS transform,” the IEEE Signal Processing Magazine, Vol. 34, No. 3, pp. 81-89, May 2017.

[3] C.-C. Jay Kuo and Yueru Chen, “On data-driven Saak transform,” arXiv preprint arXiv: 1710.04176 (2017).

[4] http://scikit-learn.org/stable/modules/clustering.html

## IV. Plan for the next week (***, **, *: order of priority)

- Finding a method to cluster the images as close as using labels to cluster

## V. Milestone