Bag of Words Classification for MNIST Fashion and Homography Computation

Naman Goyal
2015CSB1021@iitrpr.ac.in

Indian Institute of Technology Ropar,
Punjab, INDIA - 140001

1 Bag of Words Classification for MNIST Fashion Dataset

Fashion-MNIST is a dataset of Zalando's [3] article images consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image, associated with a label from 10 classes.

The process consists of 5 steps

```
1. Extracting the features from Train and Test
   Images.
2. Create Dictionary based on K-Means clustering
3. Compute Histogram on all Train and Test
   Images based on cluster centers and feature
   vector using soft assignment.
4. Each Test Image histogram is then matched
   with all Training Data histogram and majority
    vote of class of k-nearest neighbour is
   assigned.
```

Listing 1: Bag of Words Classification

1.1 Feature Extraction

Three methods were tested and results are summarized based on same.

1. Use SURF Descriptor patchwise.
2. Intensity information (appended vector) patchwise
3. First detect Harris Interest Points and then use SURF Descriptor as a feature

Observation It was observed that SURF Descriptor patchwise gave best performance. Also other descriptors which worked on patch performed better than extracting features from interest points.

Explanation Since the image size is 28*x*28 and images have lot of noise, detecting interest points gives quite redundant interest points.

Observation Neglecting the orientation gave better result i.e. neglecting rotation information.

Explanation Since images are upright, capturing rotation information is redundant.

Interest Point/Patch	Descriptor	Test Accuracy
Harris	SURF	0.67
Patch	Intensity Vector	0.72
Patch	SURF	0.80

Henceforth, Rest of the discussion assumes Patchwise SURF Descriptor unless stated otherwise.

1.2 K-Means clustering

The K-Means was implemented by the iterative Lloyd's algorithm [1], which consists of these steps.

```
1. Randomly assign cluster centers.
2. In each iteration, first assign the nearest
   cluster to each the data points based on
   distance.
3. Recalculate, the centroids of each cluster
   based on data points.
4. Stop when no data points changes its
   assignment.
```

Listing 2: Iterative Refinement K-Means

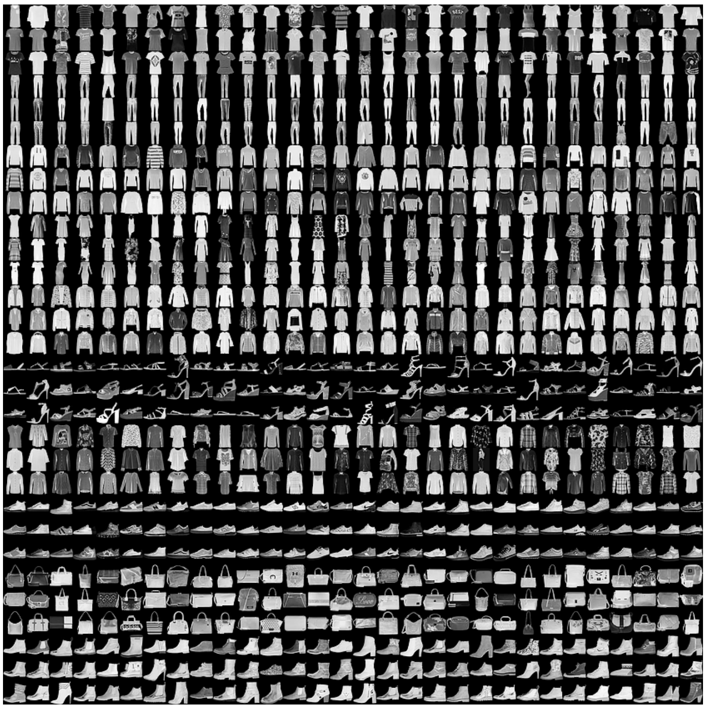

Figure 1: MNIST Fashion Dataset - Zalando Research

K-Means usually takes very long no of iterations to converge, also it may converge to sub-optimal solution.

To improve the K-Means, the Random data point assignment is replaced by spreading out initial cluster centers. The algorithm K-Means [2] is as follows

1. Randomly select one data points has first center.
2. For each data point calculate distance $D(x)$, the distance between the data points and nearest already chosen cluster center.
3. Pick the next cluster center from the data points based on probability distribution of $D(x)^2$.
4. Repeat step 2 and 3 until desired no of cluster centers are chosen.
5. Now that initial centers have been chosen, apply standard K-Means

Observation K-Means++ on average resulted in better cluster solution with less no of iterations.

Explanation Since the initial cluster are spread across data points, the objective function is minimized.

For determining the no of clusters a k-fold cross validation was done.

Listing 3: k-Fold Cross validation for no of clusters

```
1. Partition the training data into disjoints K
   groups.
2. In each of K trials, use a different group as
    validation set, and rest of the groups
   combined as training set.
3. On each trial, calculate the best no of
   clusters empirically  based on the accuracy
   on validation dataset.
4. Take average across all trials, over of the
   no of cluster.
```

5. Use this value and learn the model over whole training dataset.

Observation The best no of cluster came around 32.

Explanation Decreasing the no of cluster beyond a threshold leads to less degree of freedom while a large no of cluster size, result sin redundant cluster centers.

1.3 Histogram Computation

Observation Hard assignment gave poor performance over soft assignment

Explanation A particular feature vector, may be closer to many cluster centers and hard assignment fails to model this, by just assigning to nearest cluster center. It evens breaks the ties arbitrarily. On the other hand, soft assignment consider all neighbour and assigns weights based on distance i.e. more weights to near ones and smaller weights to farther ones.

1.4 Prediction based on k-nearest neighbours

For each test image histogram, rather than assigning class of the nearest neighbour training image histogram. Consider k nearest neighbours and take a majority vote of their classes.

Observation k value came around 3 based on cross validation.

Explanation Large k values includes more farther points and smaller k values ignores many neighbours; hence a optimal value lies which balances both.

2 Homography Computation

The process consists 2 parts first is computing homography.

1. Get the 4 corresponding pair of points from user in both the images from a GUI using ginput.
2. Construct a matrix $8x9$ A, where each pair of points corresponds to 2 rows.

3. Compute Singular Value Decomposition of A.
4. Take the last column of V in $A = U\Sigma V^T$.

Listing 4: Homography Computation

Second part is warping images to build mosaic.

1. Warp image A on B using imwarp.
2. Calculate the output view i.e world coordinates of both warped image A and original image B.
3. Take minima over top left point and maxima over bottom right corner to calcualte new world coordinates and again warp both images A and B using the computed Homography for A and Identity Homography for B in the new world coordinates.
4. Take mamixa of the warped images and output the mosaic obtained.

Listing 5: Warping Image to Create Mosaic

Observation Learned Homography depends on correctness of chosen interest points.

References

[1] K-Means *https://en.wikipedia.org/wiki/K-means_clustering*

[2] K-Means++ *https://en.wikipedia.org/wiki/K-means%2B%2B*

[3] MNIST Fashion *https://github.com/zalandoresearch/fashion-mnist*

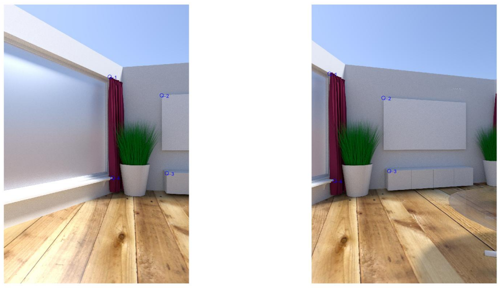

Figure 2: Interest Points chosen in both the images

Figure 3: Final mosaic by warping left image on right image

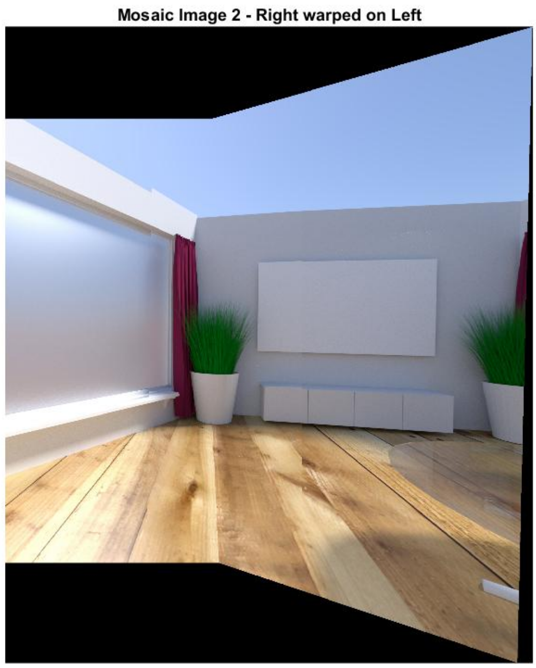

Figure 4: Final mosaic by warping right image on left image