

An Image Classification Algorithm based on AdaBoost and Different Feature Extraction Methods

Mengfan Wang
The Bradley Department of Electrical
and Computer Engineering
Virginia Tech
Blacksburg, VA
Email: mengfanw@vt.edu

Yuxian Ye
The Bradley Department of Electrical
and Computer Engineering
Virginia Tech
Blacksburg, VA
Email: herexian@vt.edu

Abstract—This proposal proposes a new image classification algorithm. Different feature extraction methods, such as SIFT, LBP and HOG are combined to complement one other's disadvantages. PCA is used to reduce the dimension of features extracted by SIFT.

Index Terms—image classifier, SIFT, LBP, HOG, PCA, bag of words, K-means, SVM, AdaBoost

I. INTRODUCTION

With the development of computer vision technology, more and more feature extraction algorithms are proposed. These algorithms have different advantages and disadvantages, which are applied in many areas. For example, features extracted by SIFT are rotation, illumination and scale invariant, and not sensitive to shades. However, SIFT algorithm is time-consuming and widely used for image matching and 3-D model construction rather than real-time image processing. HOG, another algorithm, ignores the influence of illumination and color to images, and reduces the dimension of data. But it's sensitive to shades and noises, and time-consuming, too. LBP algorithm needs little computation requirement, which is applied to face recognition and image classification. This algorithm is not sensitive to illumination but sensitive to direction and rotation.

Therefore, combine different feature extraction algorithms is regarded as a useful method to enhance the advantages of features and improve the accuracy of classification when implementing a classifier. On the other hand, previous work shows the effectiveness of boosting algorithms in the fields of image classification, such as AdaBoost or gradient boosting. But most of them focused on using different weak learners, in which case using similar learners but different features is a new idea worth trying.

II. DATA COLLECTION

The dataset is a part of Awa2 (Animals with Attributes 2) [1]. Awa2 consists of 37322 images of 50 animals classes, which provides a platform to benchmark transfer-learning algorithms. The image data was collected from public sources, such as Flickr, in 2016.

Because some classes of the dataset contain few images, and taking our poor computing resources into account, only 4

classes, dolphin, giraffe, rabbit and sheep, with more than 900 images are used. Fig. 1 shows some images of these classes. We are going to select 900 images from each class, divide them into training set and test set with a ratio of 4:1, which means 720 images in the training set and 180 images in the test set for each class.

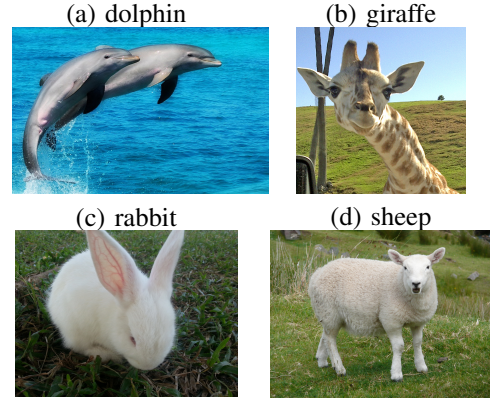


Fig. 1. Images from 4 different classes.

III. METHODS

A. Weak Classifier based on SIFT

First of all, SIFT features is the most difficult one to deal with, because the number of feature points varies from one to another image, and the dimension of features is very high. There may be hundreds of feature points in a image after

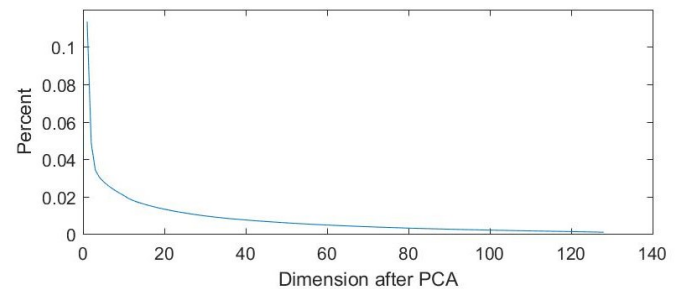


Fig. 2. The average percentage of each dimension's component of the sheep class.

filtering, and each feature point is a 128-dimensional vector. So, the first step is reducing the dimension of feature points to 40 by PCA, because the first 40-dimensional components takes the majority for any image's feature points. Fig. 2 shows the average percentage of each dimension's component, which is sampled from the sheep class. This step is based on the idea of PCA-SIFT but not quite same. Suppose $\{\mathbf{x}_{ij}\}_{j=1}^{n_j} \in \mathbb{R}^{128}$ are n_j SIFT feature points extracted from image i , this step equals to solve the problem:

$$\min_{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_{40}} \sum_{j=1}^{n_j} \|\mathbf{x}_{ij} - \mathbf{m} - \sum_{k=1}^{40} \mathbf{e}_k a_{jk}\|^2 \quad (1)$$

$$s.t. \|\mathbf{e}_1\|^2 = \|\mathbf{e}_2\|^2 = \dots = \|\mathbf{e}_{40}\|^2 = 1 \quad (2)$$

The solution to this problem is:

$$\mathbf{m} = \frac{1}{n_j} \sum_{j=1}^{n_j} \mathbf{x}_{ij} \quad (3)$$

$$a_{jk} = \mathbf{e}_k^T (\mathbf{x}_{ij} - \mathbf{m}) \quad (4)$$

And $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_{40}$ are the eigenvectors corresponding to the top 40 eigenvalues of $(\mathbf{X}_i - \mathbf{m})(\mathbf{X}_i - \mathbf{m})^T$, while $\mathbf{X}_i = [\mathbf{x}_{i1} \ \mathbf{x}_{i2} \ \dots \ \mathbf{x}_{in_j}]$.

Therefore, the feature after reduction is:

$$\hat{\mathbf{X}}_i = [\mathbf{e}_1 \ \mathbf{e}_2 \ \dots \ \mathbf{e}_{40}]^T (\mathbf{X}_i - \mathbf{m}), \quad (5)$$

while $\hat{\mathbf{X}}_i = [\hat{\mathbf{x}}_{i1} \ \hat{\mathbf{x}}_{i2} \ \dots \ \hat{\mathbf{x}}_{in_j}]$.

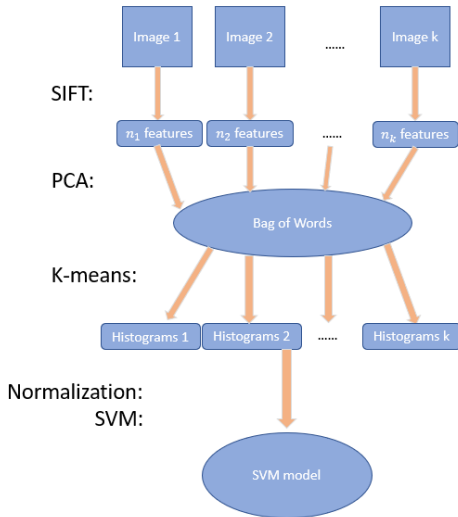


Fig. 3. The flow chart of the weak classifier based on SIFT algorithm.

After dimension reduction, all feature points extracted from the training set should be put into the bag of words model together, which forms a set $\{\hat{\mathbf{x}}_{ij} | 1 \leq i \leq Nc, 1 \leq j \leq n_i\}$. c is the number of classes, N is the number of images in each class (Suppose the numbers are same for the sake of clarity), and n_i is the number of feature points in the image i . To divide

these points in K clusters, we want the sum of residual sum of squares is the least:

$$\min_{C_1, C_2, \dots, C_K} \sum_{m=1}^K \sum_{\hat{\mathbf{x}}_{ij} \in C_m} \|\hat{\mathbf{x}}_{ij} - \mathbf{y}_m\|^2 \quad (6)$$

$$s.t. C_i \cap C_j = \emptyset \ \forall i, j \in \{1, 2, \dots, K\}, i \neq j \quad (7)$$

$$C_1 \cup C_2 \cup \dots \cup C_K = \{\hat{\mathbf{x}}_{ij}\}, \quad (8)$$

while C_1 to C_K are clusters and \mathbf{y}_m is the representative point for cluster C_m . The constraints make sure that every points can be and only be assigned to one cluster. When the objective function gets minimum its derivative equals to zero:

$$\mathbf{y}_m = \frac{1}{n_m} \sum_{\hat{\mathbf{x}}_{ij} \in C_m} \hat{\mathbf{x}}_{ij}, \quad (9)$$

while n_m is the number of points in the cluster m , and \mathbf{y}_m is the mean of this cluster.

Unfortunately, it's a NP-hard problem and we can not guarantee an optimal solution. K-means algorithm is monotonically decreasing, which moves the cluster centers around in space in order to minimize RSS, and we can only prove it will eventually arrive at a local minimum. Several effective heuristics can be used to improve the results of K-means algorithm, including (i) excluding outliers from the training set; (ii) trying out multiple starting points and choosing the clustering with lowest RSS ; (iii) obtaining initial cluster centers from another method such as hierarchical clustering [2].

The number of clusters K can not be too large or too small. In fact, if there are n points in the model, $K = \sqrt{n}$ is the best choice [3]. When the cluster centers are determined, feature points of each image can be assigned to these clusters according to the Euclidean distances between feature points and cluster centers. For the image i , a histogram vector $\mathbf{z}_i = [z_{ij}] \in \mathbb{R}^K$ counts the distribution of feature points, while z_{ij} represents the number of feature points assigned to the cluster j in the image i . Because the number of feature points is different from one to another image, the histogram vector need to be normalized:

$$\hat{\mathbf{z}}_i = \frac{\mathbf{z}_i}{n_i}, \quad (10)$$

while $n_i = \sum_{j=1}^K z_{ij}$. By these steps, an image can be transformed to a fixed-length vector.

The last step on this weak classifier is using one-versus-all (OVA) SVM [4] to classify the training set. Because the dimension of features $\hat{\mathbf{z}}$ is much bigger than the number of classes, linear kernel is enough for this problem. Suppose there are c classes $\{1, 2, \dots, c\}$, c binary classifiers are needed totally. For the m_{th} binary SVM classifier, the training set is $\{[\hat{\mathbf{z}}_i, g(y_i, m)]\}_{i=1}^{Nc}$, while y_i is the label of image i and $g(y_i, m)$ is a logical function representing the label of $\hat{\mathbf{z}}_i$. In this training set, $g(y_i, m) = 1$ if $y_i = m$ otherwise -1 . Images in class m are regarded as the positive samples and images

in other class are regarded as the negative samples. Taking outliers and noises into account, the optimization problem is:

$$\min_{\mathbf{w}, b} \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^{N_c} \xi_i \quad (11)$$

$$s.t. \ g(y_i, m)(\mathbf{w}^T \hat{\mathbf{z}}_i + b) \geq 1 - \xi_i \quad (12)$$

$$\xi_i \geq 0 \quad (13)$$

The dual problem is:

$$\max \sum_{i=1}^{N_c} a_i - \frac{1}{2} \sum_{i=1}^{N_c} \sum_{j=1}^{N_c} a_i a_j g(y_i, m) g(y_j, m) \hat{\mathbf{z}}_i^T \hat{\mathbf{z}}_j \quad (14)$$

$$s.t. \ 0 \leq a_i \leq C \quad (15)$$

$$\sum_{i=1}^{N_c} a_i g(y_i, m) = 0 \quad (16)$$

The problem can be solved by SMO and KKT conditions. Repeat c times to get classifiers

$$h_m(\mathbf{z}) = \frac{\mathbf{w}_m^T \mathbf{z} + b_m}{\|\mathbf{w}_m\|}, \quad (17)$$

while $y = \mathbf{w}_m^T \mathbf{x} + b_m$ is the hyperplane and the absolute value of $h_m(\mathbf{z})$ is the distance between \mathbf{z} and the hyperplane.

For a new sample \mathbf{z} from the test set,

IV. MODEL EVALUATION

V. FUTURE WORK

VI. CONCLUSION

REFERENCES

- [1] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-Shot Learning - A Comprehensive Evaluation of the Good, the Bad and the Ugly," in *CVPR*, 2017.
- [2] C. D. Manning, P. Raghavan, and H. Schtze, "Introduction to information retrieval," New York: Cambridge University Press, 2009, pp.360-365.
- [3] X. Wang, "A fast exact k-nearest neighbors algorithm for high dimensional search using k-means clustering and triangle inequality," *The 2011 International Joint Conference on Neural Networks*, 2011.
- [4] C. Hsu and C. Lin, "A comparison of methods for multiclass support vector machines," in *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 415-425, Mar 2002.