VEHICLE LOGO RECOGNITION WITH REDUCED-DIMENSION SIFT VECTORS USING AUTOENCODERS

Reyhan Kevser Keser, Esra Ergün, Behçet Uğur Töreyin

Istanbul Technical University, Istanbul, Turkey

ABSTRACT

Vehicle logo recognition has become an important part of object recognition in recent years because of its usage in surveillance applications. In order to achieve a higher recognition rates, several methods are proposed, such as Scale Invariant Feature Transform (SIFT), convolutional neural networks, bag-of-words and their variations. A fast logo recognition method based on reduced-dimension SIFT vectors using autoencoders is proposed in this paper. Computational load is decreased by applying dimensionality reduction to SIFT feature vectors. Feature vectors of size 128 are reduced to 64 and 32 by employing two layer neural nets called vanilla autoencoders. A dataset consisting of Medialab vehicle logo images [9] and other vehicle logo images obtained from the Internet, is used. The dataset may be reached at [10]. Results suggest that the proposed method needs less memory space than half of the original SIFT based method's memory requirement with decreased processing time per image in return of a decrease in the accuracy less than 20%.

Index Terms— Vehicle logo recognition, SIFT, dimension reduction, autoencoders

1. INTRODUCTION

The well-known feature extraction method SIFT [1] is used for large number of tasks in computer vision that requires object recognition and point matching between different scenes. SIFT descriptors are translation, rotation and scale invariant, robust to illumination variations and very useful on real-world tasks. But matching process is expensive because SIFT feature vectors are 128 dimensional and calculating distance between these vectors is very time consuming. Several methods are proposed to improve comparing process. [2], [3], [4], [5]. In this study we represented 128 dimensional SIFT vectors in 32 and 64 dimensions with vanilla autoencoders [11].

We propose a SIFT method which is more time and memory efficient than traditional SIFT method for vehicle logo recognition.

2. SIFT

SIFT (scale invariant feature transform) is a successful method to find interest points in an image. It basically consists of four steps:

2.1. Scale Space Extrema Detection

Laplacian of Gaussian (LoG) which is a blob detector can be used to obtain different sized blobs, via different sigma values. But LoG is a costly function, hence a similar function which is Difference of Gaussians(DoG) is used with different sigma values. The difference of The Gaussian blurred images with different sigma values are obtained in DoG process. These new images are formed a pyramid over scale space. In this pyramid, local extrema are marked as possible keypoints.

2.2. Keypoint Localization

The extrema points consist of strong interest points, edge points and points that have low contrast. In order to get rid of the low-contrast points, points are firstly localized with more precision and then the points whose intensity value are less than a threshold, are removed. In order to get rid of the edge points, an algorithm that is similar to Harris corner detector is used.

2.3. Orientation Assignment

In order to have rotation invariant keypoints, the orientations are defined for each keypoint according to their neighborhood. This process consists of determining neighborhood, computing orientation histogram which covers 360 degrees, weighting the histogram, then taking the highest peak and values more than 80% of it in the histogram and finally computing orientation. Hence different oriented keypoints at the same location and scale are obtained.

2.4. Keypoint Descriptor

At the previous steps, the keypoints that have scale, orientation and image location information were obtained. In other words, the keypoints are now invariant to these variables. In this step descriptor for the local image region which is invariant to remaining variables such as illumination change and local shape distortion too, is calculated.

The obtained descriptors are 128 dimensional vectors. Information in the keypoint descriptor comes from 16x16 neighborhood of the keypoint. 8 bin orientation histogram value is computed for each 4x4 sized sub-blocks which are the pieces of 16x16 neighborhood.

3. AUTOENCODERS

Autoencoders are neural networks that tries to generate its input with minimum difference. In other words, if weight and bias sets are W and b and input is x, the autoencoder tries to map input x to output x' with $h_{W,b}(x) = x'$. The goal is finding best set of parameters (W,b) that minimizes difference x-x'. Like other neural networks applications, trying to minimize x-x' is a convex optimization problem and solved with gradient based methods.

Autoencoder neural network is an unsupervised learning algorithm, it takes no label. When autoencoder is shallow and vanilla type, it learns similar to PCAs. Stacking more layers of neurons makes it easier to find correlation between different components because each layer adds a nonlinear operation with activation function. However, if capacity of autoencoder is too large, network just copies instead of finding useful features.

Generally, there are several error functions used in autoencoders. Eq. (1) and (2) are two frequently used error functions where (1) is L_2 norm and (2) is cross entropy loss that is used when the input is bit probability,

$$L = \|x - x'\|^2 \tag{1}$$

$$L_H(x, x') = -\sum_{k=1}^{d} [x_k log x'_k + (1 - x_k) \log(1 - x'_k)] \tag{2}$$

where k is the index of hidden unit. In this study L_2 norm as error function is used. There are several autoencoder types such as sparse autoencoders, denoising autoencoders, and variational autoencoders. In this study we used vanilla autoencoders.

4. METHOD

Our dataset consists of 90 cropped vehicle logo images which is obtained from 9 car brand [9].

In [1], after SIFT vectors are obtained, to compare vectors and find matchings, cosine distances were calculated. Instead of calculating cosine distance between 128 dimensional vectors, we use cosine distance between 64 dimensional vectors which are reduced vectors by Autoencoders.

Figure 1 shows the Autoencoder architecture we implemented. This is a traditional autoencoder model which has 2 encoder and 2 decoder layers with sigmoid activation function. Left-most rectangle is input SIFT feature vector with 128 dimensions. Right-most rectangle represents generated SIFT feature vector. Encoder layers encodes 128 dimensional input vectors to 64 dimensional vectors while decoder layers decode this vector back to 128 dimensions. Cost function is squared distance between original input and generated output. Our purpose is to find suitable representation for these vectors with lower number of components. This is a symmetrical network, first and second encoder layers have 128 and 64 number of units, respectively. Encoder and decoder layers are symmetric; first decoder layer has 64 number of unit while second has 128.

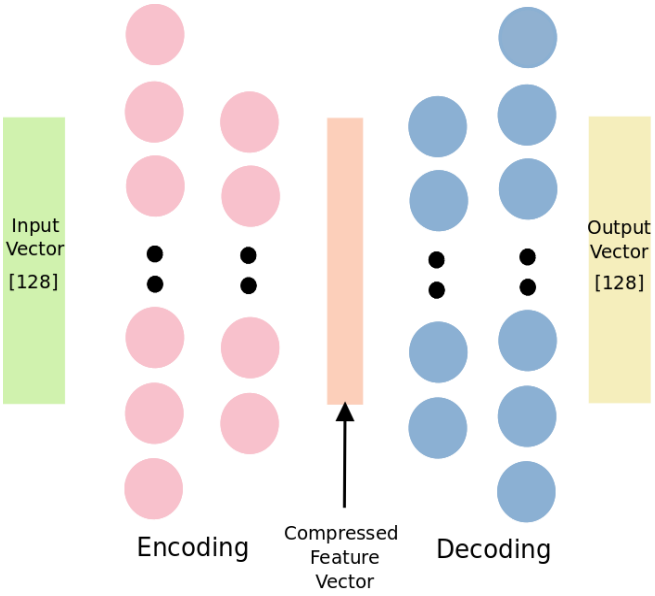

Figure 1. Our autoencoder structure

5. EXPERIMENTAL RESULTS

We used two programming languages for different parts of project. Firstly, logo images are cropped manually from all vehicle images in the dataset. Then SIFT vectors are obtained with [8] in MATLAB. The autoencoder is implemented on python with TensorFlow [6] and reduced vectors are compared on MATLAB.

Implemented autoencoder architecture is trained with 5621 SIFT vectors through 30,000 iterations. Learning rate and batch size are set to 10^{-4} and 256 respectively. Loss function is optimized with Adam [7] optimizer. Plot of loss

obtained from Eq. (1) vs iteration number can be seen from Figure 2. During training process, a single CPU is used.

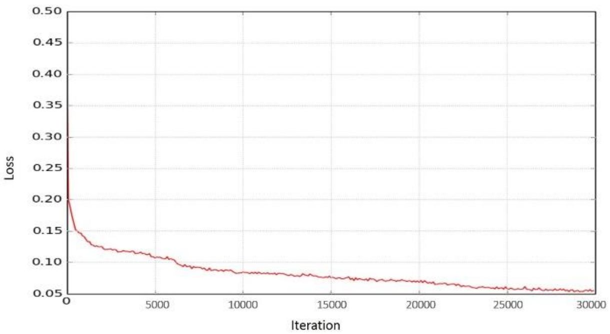

Figure 2. Loss results per iteration

The recognition performance of the proposed method is measured using the "accuracy" metric, A, defined by:

$$A = T_M / N \quad (3)$$

where T_M is the number of true matches and N is the number of all images in the data set.

81% accuracy is achieved after training through 30,000 iterations while matching process with 64 dimensional vectors. In addition to this, 32 dimensional vectors are obtained and tested. Memory usage which is the occupied memory by vectors obtained from all images in data set and logo data set, is computed The results are shown in Table 1.

Table 1. Results that are obtained with 128, 64 and 32 dimensional feature vectors.

Dimension	Process time per image (msec)	Accuracy	Memory usage (Mbit)
128	10.42	1	25.05
64	9.63	0.81	12.53
32	8.22	0.78	6.26

Table 2. Examples of input logo images, corresponding true matches and algorithm results obtained utilizing (a.) 128, (b.) 64 and (c.) 32 dimensional SIFT feature vectors

Input	a.	b.	c.	True match
				
				
				
				

6. CONCLUSIONS AND FUTURE WORK

A reduced-dimension SIFT features based vehicle logo detection method is proposed using autoencoders. The dimension reduction of features is achieved with twolayer neural network structures called vanilla autoencoders.

Results indicate that by employing the proposed dimension reduction technique, an accuracy decrease of less than 20% yields a memory space saving of more than 100% along with a reduced processing time requirement per image.

Future work consists of 4,8 and 16 dimensional reduced vectors, quantization and binarization of these vectors and using different similarity measures, such as Jaccard and Manhattan measures.

7. REFERENCES

[1] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. International journal of computer vision, 60(2), 91-110.

[2] Chen, C. C., & Hsieh, S. L. (2015). Using binarization and hashing for efficient SIFT matching. Journal of Visual Communication and Image Representation, 30, 86-93.

[3] Zhou, X., Wang, K., & Fu, J. (2016, December). A Method of SIFT Simplifying and Matching Algorithm Improvement. In Industrial Informatics-Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII), 2016 International Conference on (pp. 73-77). IEEE.

[4] Yu, L. L., & Dai, Q. (2011). Improved SIFT feature matching algorithm. Computer Engineering, 2, 074.

[5] Zhao, J., Xue, L. J., & Men, G. Z. (2010, July). Optimization matching algorithm based on improved Harris and SIFT. In Machine Learning and Cybernetics (ICMLC), 2010 International Conference on (Vol. 1, pp. 258-261). IEEE.

[6] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Ghemawat, S. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467.

[7] Kingma, D., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[8] David G. Lowe, "Method and apparatus for identifying scale invariant features in an image and use of same for locating an object in an image" U.S. Patent 6,711,293 (March 23, 2004). Provisional application filed March 8, 1999. Asignee: The University of British Columbia.

[9] Medialab LPR dataset, July. 2017[online]. Available: http://www.medialab.ntua.gr/research/LPRdataset.html

[10] ITU Logo Dataset, July. 2017[online]. Available: https://kovan.itu.edu.tr/index.php/s/o36RNe6ac6cahli

[11] Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams. Learning internal representations by error propagation. No. ICS-8506. California Univ San Diego La Jolla Inst for Cognitive Science, 1985.