Fast Depiction Invariant Visual Similarity for Content Based Image Retrieval Based on Data-driven Visual Similarity using Linear Discriminant Analysis

Y Wihardi*, W Setiawan, and E Nugraha

Department of Computer Science Education, Universitas Pendidikan Indonesia, Indonesia

*yayawihardi@upi.edu

Abstract. On this research we try to build CBIRS based on Learning Distance/Similarity Function using Linear Discriminant Analysis (LDA) and Histogram of Oriented Gradient (HoG) feature. Our method is invariant to depiction of image, such as similarity of image to image, sketch to image, and painting to image. LDA can decrease execution time compared to state of the art method, but it still needs an improvement in term of accuracy. Inaccuracy in our experiment happen because we did not perform sliding windows search and because of low number of negative samples as natural-world images.

1. Introduction

Content-based image retrieval (CBIR) is a process to search several images in image databases that similar to query image, which index images according to their content. This process is difficult, because in the real world, several images that visually similar may be very quiet dissimilar on the pixel level [1]. These images can be different in scale, orientation, illumination [2] and the visual domain/depiction [1]. So CBIR system needs to know which visual structures are important for a human observer and which are not. This phenomenon guides us to apply feature extraction that invariant to that difference structure. The study of smart classroom has been widely done.

Another challenge in this topic is how to build a good distance/similarity measure function. There are many approaches to build this function. The traditional approach using fixed similarity functions such as cosine similarity, Euclidean distance, and etc [3]. Recently, powered by availability of large internet image collections, introduced new approach that involve machine learning theory, that is learning distance function [1, 4, 5]. This approach has widely apply in domain object detections and recognitions [4, 6, 7], and image matching [1, 5]. This approach shows promising result.

Some applications of learning distance function is a Learning Per-exemplar Distances that shows in [4, 6, 7] and Data-driven Visual Similarity that show in [1]. The basic idea is that each image has a unique distance/similarity function to distinguish it from the other image. On the implementation, both Learning Per-exemplar Distances and Data-driven Visual Similarity, we can use the various classifier, such as Support Vector Machine. However, in its application to CBIR there are issues in terms of computing speed. This happens because every time the retrieval task was conducted, the distance function needs to be learnt. So it is need a fast classifier to learn the distance/similarity function.

In this research we use Linear Discriminant Analysis (LDA) as a classifier to learn the visual similarity function based on Data-driven Visual Similarity and combine it with some state of the art feature space. We use the LDA because of its simplicity, perhaps it can give a fast learning process of thousands negative image sample.

2. Related works

There are two main components that construct CBIR system: feature space and similarity/distance function.

2.1. Feature extraction

Study in feature extraction that invariant to scale, rotation, and illumination has done by several researchers:

2.1.1. Scale Invariant Feature Transform (SIFT). This method introduced by Lowe in 2004 [2]. It extracts distinctive invariant feature from image that are invariant to image scale, rotation, distortion, change in 3D viewpoint, and change in illumination. In general, this method detects scale space extrema by implement difference of Gaussian, then localize the key points and assign one or more orientation. Finally construct feature space into 128 length vector of each key points.

2.1.2. Histograms of Oriented Gradient (HOG). HOG was first introduced by Dalal and Triggs for human detection problem in 2005 [8]. This method normalizes gamma and color of input image, and then compute the gradients of it. After that, it does the weighted vote into spatial and orientation cells, and finally normalize contrast over overlapping spatial blocks. This method most similar to SIFT.

2.1.3. Speeded Up Robust Features (SURF). SURF is claimed as refinement method of SIFT. It introduced by Bay et al. in 2008 [9]. This method more precise and faster than others because of relying on integral images for image convolutions.

2.1.4. Binary Robust Invariant Scalable Key points (BRISK). BRISK is state of the art method in invariant scale and rotation feature extraction. This method is refinement of SIFT and SURF. It introduced by Leutenegger et al. in 2011 [10]. In general, this method detects scale space keypoints using saliency criterion, and finally descript to the BRISK feature. This method can be fast because using of quadratic function fitting to obtain location and the scale of each keypoint.

2.2. Learning distance/similarity function

The study in Learning Distance/Similarity Function was initiated by Malisiwicz et al. that introduced the Distance Learning Per-exemplar technique in 2008 on recognition domain problem [4]. The study continues by implementing this method on object detection problems in 2011 [6]. On this study they train a SVM classifier using single exemplar and the large of negative samples. Weight of the best hyperplane used to build distance function. This method shows a pretty good result compared to the Pascal VOC [11] which is state of the art at the time.

Then, they apply this method on image retrieval problem in [1, 7] known as Data-driven Visual Similarity. In this research they still use SVM classifier to obtain similarity function. Dot product with the SVM weight of optimal hyperplane used to define the visual similarity function.

Unfortunately, until this recent study the main problem is in terms of speed. Learning Similarity/Distance Function approach requires a great resource in implementation. This occur because every retrieval task for a query image, the distance/similarity function needs to learn from thousands negatives images sample. So this approach needs a fast classifier to face that problem.

3. Methods

The focus of this paper is how to compute visual similarity rapidly which introduced by Shrivastava et.al [1] and apply it on Content Based Image Retrieval (CBIR). We revisit the data driven uniqueness method [1] and reanalyze what they done. Data driven uniqueness approach do discriminant analyses, and use linear SVM as classifier to find the best hyperplane between query image and negative samples. Then the weight of hyperplane used to compute visual similarity. As we know, SVM need a complex computation, so the impact on slow speed. We propose a LDA as classifier because of its simplicity, so it can give a fast learning process of thousands negative image sample. Additionally, as original as data driven uniqueness approach, we use HoG [8] as feature representation, because of its robustness.

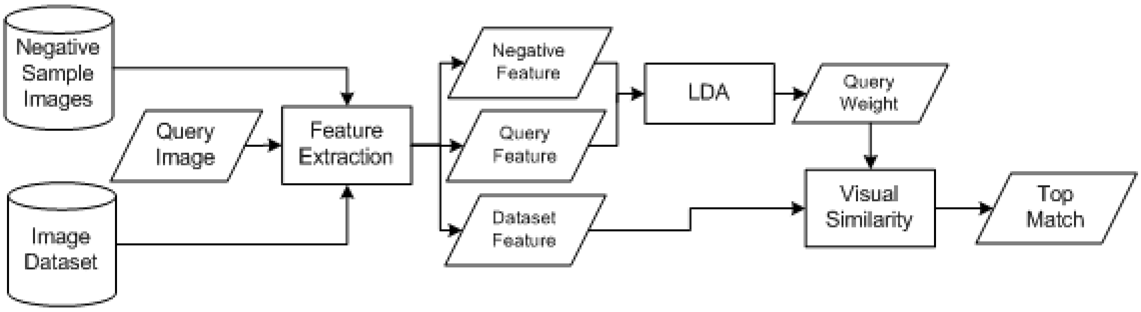

Figure 1. Flowchart of Our Method

Firstly, our methods do feature extraction using HoG descriptor (Fig.1). Then we find discriminant between query feature and negative samples using LDA classifier. Result weight from LDA model defines as query weight (Wq). To compute similarity S(Xq,Xi) between query feature (Xq) and each data (Xi) in dataset we use data driven uniqueness visual similarity formula (1):

$$S(Xq,Xi) = WqT.Xi \tag{1}$$

Based on result from that formula, we choose highest score as the best match.

3.1. HOG feature extraction

We use HoG feature because it still contains global information of image, different with keypoint based feature like SIFT, SURF, and BRISK. So we can do much treatment to its feature to recognize its pattern. On this research we use standard HoG descriptor based on [12]. In configuration, we use 9 bin histogram per cell and 3x3 pixel per cell.

Detail step for feature extraction describe bellow:

1. Color and gamma value normalization.
2. Compute gradient using kernel filter [-1,0,1] and $[-1,0,1]^T$
3. Spatial/orientation binning using 9 channels histogram between 0° to 180°.
4. Block normalization.

3.2. Linear discriminant analysis (lda)

To analyze the discriminant between query image and negative samples, we use LDA. We define image query as positive class (class-1) and negative samples as negative class (class-2). In this case, the LDA optimization problem define as:

$$J(\omega) = \frac{\sigma_{bet}^2}{\sigma_{wit}^2} = \frac{(\omega.(\mu_2-\mu_1))^2}{\omega^T(\Sigma_1+\Sigma_2)\omega} \tag{2}$$

where σ_{bet}^2 is the variance of between class, σ_{wit}^2 is the variance within class, ω is weight vector, μ_1, μ_2 are mean of each class (1 and 2), $\sum_1, \sum_2$ are covariance of each class.

To maximize the separation between data, so we solve the discriminant by using formula:

$$\omega = (\Sigma_1 + \Sigma_2)^{-1}(\mu_2 - \mu_1) \tag{3}$$

and then use that weight to compute visual similarity in formula (1).

4. Results and discussion

To validate our method, we do three typical experiments. First, image to image retrieval, second painting to image, and the last one sketch to image retrieval.

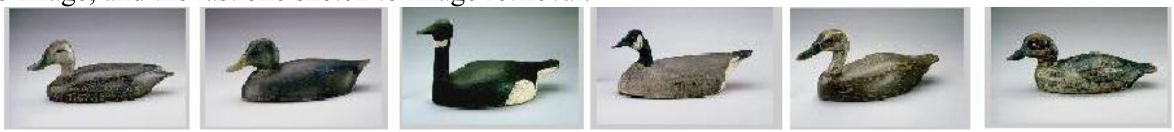

Figure 2. First left is query image on class ‘decoys’ and followed by Top-5 match

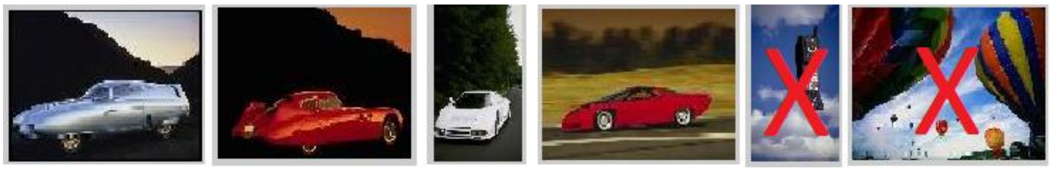

Figure 3. First left is query image on class ‘car’ and followed by Top-5 match

On our experiment we use 21 class of Corel Image Dataset [13] that contains 2720 images of 120x80 pixel dimension. As negative samples or natural-world images, we use 10.000 Random Flickr Images that used in [1].

Table 1. Precision for Top-n

Class	Top-5	Top-10	Top-50	Top-100
pumkin	0.29	0.19	0.11	0.09
aviation	0.27	0.20	0.13	0.11
baloon	0.27	0.18	0.12	0.10
bob	0.65	0.50	0.18	0.11
bonsai	0.34	0.27	0.19	0.17
bus	0.34	0.24	0.15	0.13
car	0.52	0.38	0.19	0.14
cards	0.77	0.70	0.36	0.23
decoys	0.86	0.82	0.57	0.38
dish	0.61	0.55	0.37	0.28
doll	0.48	0.41	0.28	0.22
door	0.70	0.62	0.41	0.31
easteregg	0.56	0.51	0.42	0.35
flags	0.62	0.52	0.30	0.23
mask	0.49	0.39	0.25	0.20
mineral	0.46	0.40	0.27	0.21
molecular	0.43	0.37	0.26	0.22
orbits	0.56	0.51	0.42	0.35
ship	0.43	0.37	0.26	0.22
steameng	0.62	0.52	0.30	0.23
train	0.34	0.27	0.19	0.17
mean	**0.51**	**0.42**	**0.27**	**0.21**

In the image to image retrieval experiment, we use all of the data in dataset as query image. The precision result of top-5, top-10, top-50, and top-100 for each class data is show in Table 1, and the success story for our top-5 retrieved images show in Figure 2 and Figure 3.

For painting to image experiment, we use 11 random painting of car which got from internet. The precision result for top-5, top-10, top-50 and top-100 sequentially are 0.38, 0.32, 0.14, and 0.12. We show the success story for this typical experimental result in figure 4 and figure 5.

In the last experiment, sketch to image retrieval, we use 16 random sketch of car which used as query image in [1]. We show the success story for top-5 retrieved images in figure 5 and figure 6. The precision for top-5, top-10, top-50, and top-100 are 0.35, 0.28, 0.16, and 0.13.

We do our experiment on single CPU with core i3 2.22 GHz processor and 2GB of RAM. Our method takes ±0.2 second time execution on single query images.

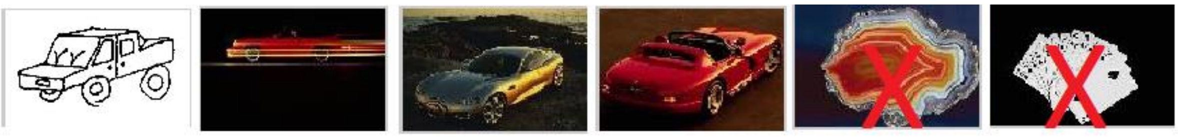

Figure 4. First left is query image on class 'decoys' and followed by Top-5 match

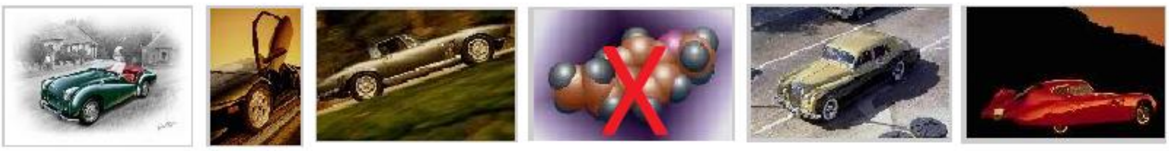

Figure 5. First left is query image on class 'car' and followed by Top-5 match

5. Conclusion

We have tried to use LDA to obtain exemplar learning similarity function for each query image and apply it in Invariant Depiction Content Based Image Retrieval System (CBIRS). It gives fast online computation with ± 0.2 second execution time on single query. It is faster than original method that using linear-SVM, which takes until three minutes' time execution on a parallelized 200node-cluster.

The precision of our method still needs an improvement. This result looks low because we did not perform sliding windows search as in [1, 8] . We did not do that, in purpose to reduce complexity and decrease the execution time. Besides that, inaccuracy can cause by the low number of negatives image as natural-world, so it did not distribute normally, since LDA can get best discriminant for normally distributed data.

References

[1] Abhinav S, Tomasz M, Abhinav G and Alexei A E 2011 *Proc. of the 2011 SIGGRAPH Asia Conf.* (Hong Kong)

[2] David G L 2004 *Int. Journal of Computer Vision* **60** 91-110

[3] Ritendra A, Dhiraj J, Jia L and James Z W 2008 *ACM Transactions on Computing Surveys*

[4] Tomasz M and Alexei A E 2008 *IEEE Conf. on Computer Vision and Pattern Recognition* (Anchorage) 1-8

[5] Liorf W, Tal H and Yaniv T 2009 *IEEE Int. Conf. on Computer Vision 12th* (Kyoto) 897-902

[6] Tomasz M, Abhinav G and Alexei A E 2011 *IEEE Int. Conf. on Computer Vision (ICCV)* (Barcelona) 89-96

[7] Tomasz M, Abhinav S, Abhinav G and Alexei E 2012 *Int. Conf. on Machine Learning* (Edinburgh)

[8] Navneet D and Bill T 2005 *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* (San Diego) 886 - 893

[9] Herbert B, Tinne T and Luc V G 2008 *Computer Vision and Image Understanding* **110** 346-359

[10] Stefan L, Margarita C, and Roland Y S 2011 *IEEE Int. Conf. Computer Vision (ICCV)* (Barcelona) 2548 - 2555

[11] Mark E, Luc V G, Christopher K I W, John W and Andrew Z 2010 *Int. Journal of Computer Vision* **88** 303-338

[12] Oswaldo L J, David D, Valter G and Urbano N 2009 *12th Int. IEEE Conf. On Intelligent Transportation Systems* (St. Louis) 432-437

[13] Jia L and James Z W 2003 *IEEE Trans. on Pattern Analysis and Machine Intelligence* 1075-1088