Visual-to-Semantic Hashing for Zero Shot Learning

1st Xin Li
East China Normal University
Shanghai, China
51174500103@stu.ecnu.edu.cn

2nd Xiaoyue Wen
Engineering Research Center of Intelligent Transport of Zhejiang Province
Enjoyor Co., Ltd, Hangzhou, China
wenxiaoyue@enjoyor.net

3rd Bo Jin
East China Normal University
Shanghai, China
bjin@cs.ecnu.edu.cn

4th Xiangfeng Wang
East China Normal University
Shanghai, China
xfwang@sei.ecnu.edu.cn

5th Junjie Wang
East China Normal University
Shanghai, China
jasonwang.ecnu@gmail.com

6th Jinghui Cai
East China Normal University
Shanghai, China
51174500002@stu.ecnu.edu.cn

***Abstract*—Hashing-based multimedia retrieval are facing the problem of the dramatic increase of data, especially new unseen categories. It is time-consuming, expensive, and sometimes impractical to label new samples and retrain the hashing model. Recently, several zero-shot hashing methods are proposed to generate the hash function with good generalization for unseen classes, via exploring semantic information and similarity relationship. However, the performance of existing solutions is still not satisfying. Therefore, we propose a modified two-stage framework, called Visual-to-Semantic Hashing (VSH). To fully exploit the semantic information, visual feature is firstly mapped to the semantic space, and then generate its hash codes. To transfer supervised knowledge from seen classes to unseen classes, a margin-based ranking loss is employed to learn the semantic structure. To boost the discriminability of semantic mapping, a classification module is adopted to distinguish between different semantic mapping vectors. Plenty of experiments demonstrate that the proposed VSH is superior to state-of-the-art methods.**

***Index Terms*—Hashing, zero shot, cross-domain, multimedia retrieval**

I. Introduction

Due to the dramatic increase of the scale of multimedia data, hashing-based method have attracted more and more attention in the field of multimedia retrieval [1]. The aim of hashing methods is to encode the images into binary codes, named hash codes [2]. Through learned hash function, the whole image data is projected into a discrete binary space, so called hamming space. Compared with the real-value calculation, hamming distance can be efficiently calculated in a very short time owing to its binarization and low bits.

Generally, hashing methods mainly fall into two categories: unsupervised hashing [3]–[5] and supervised hashing [6]–[8]. The unsupervised hashing often takes into account the distribution and manifold structure of the samples, while the latter makes full use of the semantic information, like class label or pair-wise relation to learn similarity-preserving hash codes. Due to lack supervised knowledge and intrinsic semantic property, unsupervised hashing methods usually have lower performance than supervised hashing methods. Additionally, with the rapid development of deep neural network, deep hashing approaches [10], [18]–[20] have been proposed with significant improvements. However, these supervised hashing methods require manual labels. It has a potential problem that hashing model is unable to produce accurate hash codes when a new category appears without supervised label.

Corresponding author: Bo Jin, Xiangfeng Wang

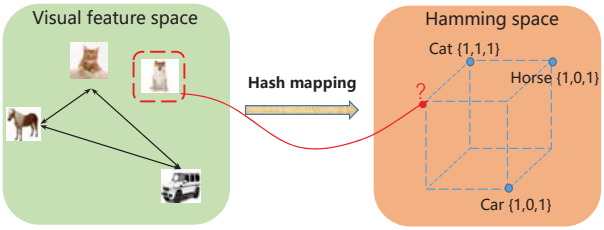


Fig. 1. Illustration of zero-shot hashing. The key of hashing is to learn the mapping between visual feature space and hamming space. The red rectangle in visual space represent unseen class("dog"). For zero-shot hashing, the hash function learned from seen classes should generalize well for unseen classes.

Moreover, labelling new samples and retraining the hashing model can be time-consuming, expensive, and sometimes impractical. Therefore, this is a demand to seek a training mechanism, which makes the learned hash function can be effectively generalizable to unseen classes. The detailed description is shown in the figure 1. The hash function, mapping the data samples into the hamming space, is learned from the seen classes (e.g. horse, cat and car). For a new unseen class (never appear in the training phase), zero shot hashing needs to generate codes, that not only are closer to the cat than to the car, but also maintain differentiation among all classes.

The main reason why existing methods can't work well is that these model only learn the hash function in train set and has no access to new concepts or related information. To solve above problems, zero-shot learning (ZSL) [25] are employed for hash function learning. The aim of zero-shot learning is to learn a general mapping from the feature space to a high-level semantic space, where the relationships of seen classes and unseen classes are well characterized and thus seen supervised knowledge can be transferred to unseen classes. Two semantic spaces are widely used in the literature, i.e. class-attributes labeled by experts and word vectors extracted from natural

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

language processing models. The semantic space bridges the semantic gaps between low-level visual features and high-level semantics. Inspired by the success of zero-shot learning, zero-shot hashing is firstly introduced by [13]. Transferring Supervised Knowledge (TSK) is the pioneer method, which projects independent data labels ($0/1$-form label vectors) into semantic embedding space. Similar to zero-shot learning, the main framework of TSK is to project the hash codes into a semantic space. The obtained hash function are able to transfer supervised knowledge from seen classes to unseen classes. Guo et al. [14] propose a novel deep hashing method, named Discrete Similarity Transfer Network (SitNet). SitNet introduces the semantic embedding scheme via enforcing the hash codes to capture the semantic similarity relationship among different categories. Both the core architecture of two methods above is to learn a mapping from hash codes to semantic space. However, they ignore the control of hash code distance explicitly. In other words, if two classes are close in the semantic space, they may even be closer in the hamming space, which may makes it indistinguishable for hash codes of unseen classes and their similar training classes. Recently, Attribute-Guided Network (AgNet) [15] is proposed for cross-modal zero-shot hashing. Different with above methods, AgNet consists of two stages. First, visual features and word vectors are embedded into a common attribute space respectively. Then, visual hash codes and textual hash codes are generated by attribute similarities and inter-modal similarities loss. Although AgNet pays more attention to maximize the distance between classes, it views semantic mapping as a classification problem and adopt binary cross entropy for attribute transfer learning in first stage, which is proved to have a poor generalization performance for zero-shot learning [12]. The attribute learning will greatly affect the generation of subsequent hash codes, especially for unseen classes.

To address the issues above, we proposed a novel zero-shot hashing method, named **visual-to-semantic hashing** (**VSH**). In order to enhance the measurement of hash codes in hamming distance, we project visual features into semantic space. Different with AgNet, margin-based ranking loss is employed to learn semantic structure among seen classes. For the image retrieval task, we aim to search the closest samples around query samples rather than recognize the specific category. So as to enhance the differentiation of vectors, classification module is applied to make the vectors of same categories close and make the vectors of different categories far away in the semantic space. Additionally, we map the learned semantic vector to the hamming space to learn similarity-preserving binary-like image representations. Hash codes are achieved by measuring the network outputs to pull the codes of similar images together and push the codes of dissimilar images away from each other. Our contributions are summarized as below:

- A modified two-stage zero-shot hashing framework is proposed, which firstly maps visual feature to semantic space and then output hash codes according to learned semantic vectors. The hash function learned with semantic information has significant performance on unseen classes.
- To fully exploit the semantic information, a margin-based ranking loss is employed to learn the semantic structure, which is able to transfer supervised knowledge from seen classes to unseen classes. To boost the discriminability of semantic mapping, a classification module is introduced to distinguish between different semantic mapping vectors.
- A learning scheme is adopted to pull the codes of similar images together and maximize the hamming distance between the codes of different categories so as to enlarge the hamming distance explicitly.

The paper is organized as follows. A brief review of zero-shot hashing is presented in Section 2, together with related works. Section 3 reports our algorithm framework. Experiments and discussion are shown in Section 4 and Section 5 concludes this paper.

II. Related Work

A. Traditional Hashing

Recently, lots of hashing methods have been proposed to improve the performance on image retrieval, because of their fast retrieval efficiency and low storage on large-scale multimedia data. At the beginning, researchers hashed the data into hamming space by several random function, such as a family of methods called Locality Sensitive Hashing [17]. However, because of lack prior knowledge, LSH methods usually require hundred of bits to achieve satisfied performance. Therefore, data-dependent methods have become more and more popular. Spectral Hashing [3] explores the data distribution and view hash learning as a graph partitioning question. Iterative Quantization Hashing (IQH) [5] finds a rotation of zero-centered data of samples, so as to minimize the quantization error from mapping the data to hamming space. [6] proposed a Semi-Supervised Hashing (SSH) framework that minimizes empirical error over the labeled set and an information theoretic regularizer over both labeled and unlabeled sets. Recently, deep neural network has also been introduced to deep hashing methods to improve the representational ability. CNN Hashing (CNNH) [18] firstly replaced hand-crafted feature vectors with deep CNN extracted feature. The hash codes are learned from similarity matrix decomposition. Then, deep deep network is employed to learn target hash codes for input images. Deep Supervised Hashing (DSH) [9] used an end-to-end training way, which directly quantize the outputs of image based on discriminability terms.

B. Zero-Shot Hashing

Inspired by zero-shot learning on image classification, Yang et al. firstly introduced the zero-shot hashing question [13], which learnt the hash function only from seen classes and can generalize well for unseen classes. They used a NLP model to transform seen labels into a semantic-rich embedding space, where each label is represented by a real-valued vector.

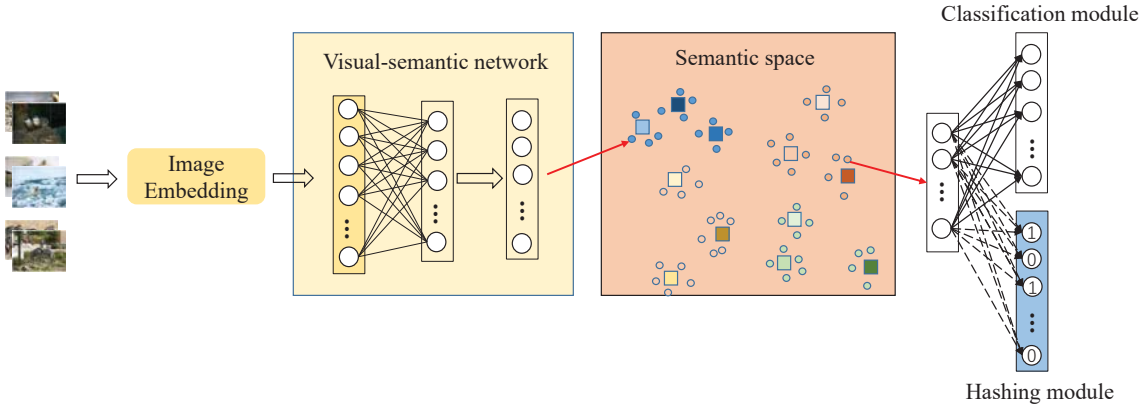


Fig. 2. Overview of the proposed hashing framework VSH. Visual features are extracted from image embedding network, which is pre-trained on ImageNet-1K. Visual-semantic network embeds the inputs of visual features into semantic space. Two Additional module are employed behind visual-semantic network. The hashing module aims to output binary hash codes, and the classification module improves the discrimination of vector in semantic space so as to generate high quality hash codes. The squares in semantic space represent target semantic vectors extracted from word2vec model and the circles represent mapping vectors of visual features.

Through this semantic space, the structure of all classes can be well captured. Then, low-level visual features are projected into the semantic space, which makes the supervised knowledge transferred to unseen classes. Furthermore, a semantic alignment strategy is proposed to aligns the initial embedding space with the distributional properties of low-level visual feature. Discrete Similarity Transfer Network (SitNet) [14] was proposed to learn the semantic similarity for zero shot learning. Similar with the TSK, they also map the learned hash codes to semantic space, and adopt a multi-task architecture to exploit the supervised information for seen concepts and learn discriminative hash codes simultaneously. Ji et al. consider investigating ZSH in a cross-modal retrieval setting [15]. They propose a two-stage learning scheme, which embed the visual features and textual features into a shared binary attribute space and then encodes the visual and textual vectors in attribute space into hamming space, respectively. Liu et al. proposes a general Cross-modal Zero-shot Hashing (CZHash) solution [22]. CZHash first quantifies the composite similarity between instances using label and feature information. Then, binary hash codes are obtained based on the category attribute spaces and quantification hash functions. Shen et al. projects both visual features and textual features into an intermediate unified hamming space [21]. However, the proposed method mainly aims to solve zero-shot classification task and the retrieval settings is different with above methods. Lai et al. take a two-streams network for zero-shot hashing [16], which the first stream operates on the labelled images of the seen classes while the second stream operates on the unlabelled images. The main difference between TZSH and other methods mentioned is that unlabelled images from the target classes are available with transductive setting.

III. THE PROPOSED APPROACH

A. Problem Definition

In this section, we first introduce the definition of zero-shot hashing similar as [13], [14]. Given a training set $D_{tr} = \{(x_i, y_i, c_i)\}_{i=1}^{N}$, $x_i \in R^d$ and $y_i \in Y^{tr}$ denote the visual feature and the label of the i-th image respectively, and Y^{tr} denotes the one-hot code ($\{0,1\}^L$) label set of training class with L be the number of seen classes. Additionally, $c_i \in R^Q$ denotes the semantic vector of the i-th image, and $c_i = c^\ell$ is the i-th image belongs to class ℓ ($\ell \in \{1, \cdots, L\}$). Define $C^{tr} = \{c^1, \cdots, c^L\} \in R^{Q \times L}$ as the semantic vectors of training set. For instance, word vectors (extracted from natural language processing method) and attributes (labeled manually by experts) can be chosen as semantic vectors. In most cases, attributes yield better performance than word-vector embeddings. However, in our work, the word vectors which are pre-trained on GoogleNew300 is selected as semantic space because of its convenience and universality. For zero-shot hashing setting, testing instances are sampled from new categories, shareing no common label with training set. The hash function is defined as a map from the visual features to a H-bit hash code $f : R^d \rightarrow \{-1, +1\}^{\mathrm{H}}$. Based on supervised knowledge of the semantic space, the learned hash function f should not only preserve similarity among seen classes, but can transfer to unseen classes.

B. Network Architecture

The overall framework of the proposed VSH framework is illustrated in Figure 2. The visual features are extracted via a image embedding network like ResNet-101, which is pre-trained on ImageNet-1K. Following the setting [13], [15], the parameters of image embedding network are fixed and aren't updated in training phase. Then, the Visual-semantic Transfer Network (VTNet) consists of a full connection layer with Q output units, embeding the visual features into the semantic space S, where semantic similarity relationships between

different categories are characterized. Based on the semantic space, the supervised knowledge can be transferred from seen classes to unseen classes. Two additional fully-connected layer named hashing module and classification module, are employed behind visual-semantic network. The former aims to generate hash codes from the learned semantic structure, and the latter improve the separability between categories in semantic space, which is intrinsically equivalent to the target hash codes.

C. Objective Function

To sum up, a novel two stage network architecture is proposed to learn compact hash codes in zero-shot setting. In the first stage, we design the objective functions for VTNet, which aims to map the visual feature space into the semantic space. Consider that hashing is a ranking problem in itself, we study the ranking loss to learn similarity structure between seen classes and unseen classes. Additionally, due to the two stage architecture, the performance of VTNet will directly affect the quality of generated hash code. In order to obtain more discriminative vectors in semantic space, a margin-based ranking loss is employed to learn generalization model. Given a training set of instances and their corresponding semantic vectors, the VTNet is modeled with the margin-based objective function:

$$\mathcal{L}_t = \sum_{i=1}^{N} \max\left(0, m + \left\|s^{\ell_i} - c^{\ell_i}\right\|^2 - \min_{\ell_j \neq \ell_i} \left\|s^{\ell_i} - c^{\ell_j}\right\|^2\right), \tag{1}$$

where s^{ℓ_i} is the i-th predicted vector of visual feature in semantic space, c^{ℓ_i} is the target semantic vector, and m is a margin parameter to control the distance between two similar classes. Frome et al. [23] demonstrates that margin-based loss can obtain better results in zero-shot learning, while some other loss functions, such as mean square error (MSE), yield about half the accuracy of the rank loss model. In zero-shot hashing, MSE loss makes the model more focused on the learning of seen categories and hurts the generalization for unseen classes.

Furthermore, the margin-based ranking loss only aims to metric distance between different target semantic vectors. There is a potential issue that two predicted semantic vectors from different categories may be close to each other. We add the classification module to alleviate this phenomenon and make predicted vectors from same categories compact. The cross entropy loss is defined as:

$$\mathcal{L}_{ce} = \sum_{i=1}^{N} y_i \log p_i, \tag{2}$$

where p_i is the classification probability of i-th image.

According to the predicted semantic vectors of visual features, the core algorithm is to generate compact hash codes B. The similar image should be naturally encoded to similar binary codes. Meanwhile, the codes of dissimilar images should be as far as possible [9]. Based on the above considerations, the objective function is designed to make hash codes of same category close together and push the hash codes of different categories away from each other. Then, the loss is defined as:

$$\begin{aligned}\mathcal{L}_h = &\sum_{i,j}^{N} s_{ij} * \left\|b_i - b_j\right\|^2 + s_{ij} * \max\left(0, \lambda - \left\|b_i - b_j\right\|^2\right) \\ &+ \alpha\left(\left\||b_i| - e\right\|_1 + \left\||b_j| - e\right\|_1\right),\end{aligned} \tag{3}$$

where s_{ij} is equal to 1 if i-th image and j-th image are belong to same categories; s_{ij} is equal to 0 if i-th image and i-th image are belong to different categories; λ denotes a margin parameter; α is penalty parameter; $e \in \mathbb{R}^{\mathrm{H}}$ denotes the all one vector. The first and second terms control the distance relationships between similar images and dissimilar images. The third penalty term drives the algorithm to produce binary-like vectors.

Finally, super parameters m and λ influence the algorithm performance. The detailed procedure of our proposed **VSH** algorithm is proposed in Algorithm 1, and the overall loss function can be formulated as

$$\mathcal{L}_{all} = \mathcal{L}_t + \mathcal{L}_{ce} + \mathcal{L}_h. \tag{4}$$

Algorithm 1 Procedure of **VSH.**

Input: The training dataset $\mathcal{D}_{tr} = \{(x_i, y_i, c_i)\}_{i=1}^{N}$; The set of word vectors C^{tr}; The retrieval database X^P; The unseen query set X^Q;

Output: MAP and Precision@R;

1: Mapping visual feature x_i into semantic space, with distance between word vector c_i and C^{tr};
2: Maximizing the distance from predicted semantic vectors and learning hash function f;
3: Calculating the hash codes B^P for X^P with f;
4: Calculating the hash codes B^Q for X^Q with f;
5: Calculating MAP and precision@R with the hamming distance between B^P and B^Q
6: **return** MAP and Precision@R;

IV. Experiments

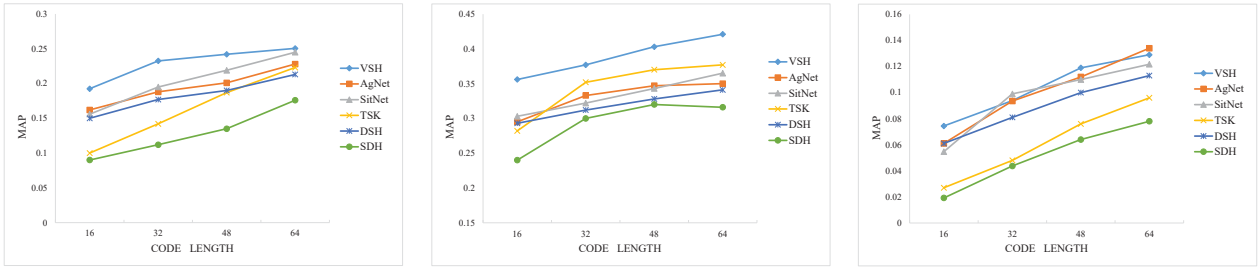
A. Platform

Our experiments are perfromed on a linux 64-bit system, with Intel(R) Xeon(R) E5-2680 v2 @2.80GHz×6 CPU, GeForce GTX1080Ti GPU and 128G RAM. The code is implemented in Python and Matlab. We utilize PyTorch to train all models.

B. Dataset

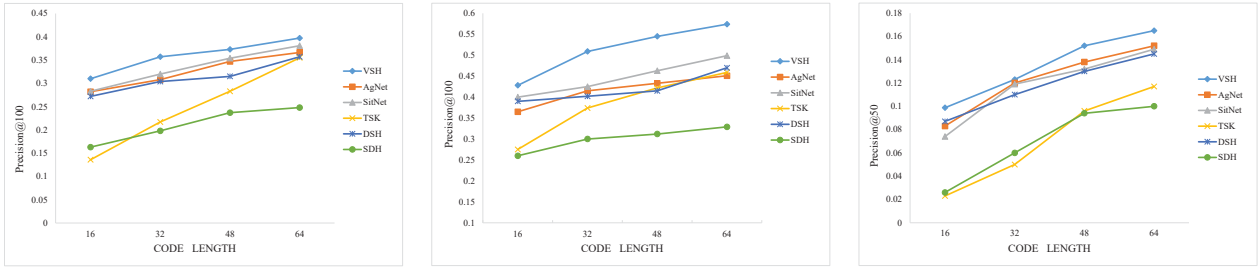
In our experiments, we employ four popular image datasets, i.e., AwA, aPY, CUB and ImageNet.

- Animals with Attributes (AwA) [25] consists of $30,475$ images from 50 animal categories, which each category is provided by 85 associated class-level attributes. This is the first dataset with attributes for zero-shot learning.
- Attribute Pascal and Yahoo (aPY) [26] has $15,339$ images coming from Yahoo and Pascal VOC 2008. It has 32 classes annotated with 64 attributes.



(a) AwA (b) APY (c) CUB

Fig. 3. Mean Average Precision on AwA, aPY and CUB datasets.



(a) AwA (b) APY (c) CUB

Fig. 4. Precision on AwA, aPY and CUB datasets.

- Caltech-UCSDBirds (CUB) [27] is a fine-grained dataset, which consists of $11,788$ images from 200 different types of birds annotated with 312 class attributes.
- ImageNet [28] is a large-scale image dataset organized according to the Word-Net [29] hierarchy. The subset of ImageNet for the Large Scale Visual Recognition Challenge 2012 (ILSVRC2012), which covers over 1.2 million images with $1,000$ object categories, is adopted in our experiments.

C. Experimental Setting and Comparison

Supervised Discrete Hashing (SDH) [8] and Deep Supervised Hashing (DSH) [9] are selected as two superior traditional hashing methods. Zero-shot Hashing with Transferring Supervised Knowledge (TSK) [13] and Discrete Similarity Transfer Network (SitNet) [14] are choosen as two single-modal zero-shot hashing methods. Besides, Attribute-Guided Network (AgNet) [15] is a typical cross-modal zero-shot hashing approach. These five state-of-the-art methods above are compared with our proposed VSH as the baseline.

For comparison, we adopt the ResNet101 [24], which is pre-trained on ImageNet-1K, to extract the fully connected layer as visual features. Each category is embedded into a 300-dimention word vector, extracted from word2vec model. For experimental normalization in zero-shot learning, [12] provides related data features and the proposed splits (PS), which guarantees that there is no intersection between the test categories and the pre-train categories.

It is noticed that we only use seen classes for training and all evaluations are taken on unseen classes. We employ two widely used evaluation metrics in our experiments. The first is mean average precision (MAP), reflecting the whole ranking results of retrieval. The second is precision at top-rank 100/50(precision@100/precision@50). For CUB dataset, the average number of each category is too small. So we only apply precision@50. In the actual application of image retrieval, top rank results embody the effectiveness of the model.

In order to train our model, xavier uniform [30] is used to initialize the network parameters. Adam [31] is adopted to optimize model. The initial learning rate is set to 0.001 and the weight decay parameter is 0.0001. The mini-batch size is set to 128. To ensure the same training data and test data for all approaches, we fix random seed to 1. The m, α and λ are set to 1, 0.001 and $2 * H$ respectively.

D. Results on AwA, aPY and CUB Datasets

AwA dataset: AwA dataset contains 50 animal categories and total $30,475$ images. For fair comparison, 40 classes are selected as seen classes while the rest 10 classes are used as unseen classes. We randomly choose $10,000$ samples from seen classes to train the model. 1000 images from the unseen classes are selected as query samples. All images expect query samples form retrieval database.

The comparison results are shown in Fig. 3(a) and Fig. 4(a). Our method yields the highest accuracy and beats all the baselines. With the increase of code length, the MAP performances of all methods keep improving, which is similar to the phenomenon in the precision@100. Our approach gains 23.2% in 32 bits, which has an improvement against AgNet by 18.1% in the same length.

Fig. 5. Effects of different number of class for training and testing on ImageNet dataset.

aPY dataset: This dataset consists of $15,339$ images from 32 classes, i.e., "aeroplane", "dog" and etc. Taking the limitation of dataset size into consideration, we only randomly select $5,000$ samples from 20 seen classes as training data and 500 images from 12 unseen classes as query data. The remaining unseen class images together with all seen class images regarded as the retrieval database.

Fig. 3(b) and Fig. 4(b) shows the performances of all comparing approaches. Because the data size is small and the semantic similarity of categories is distinguished widely, all the algorithms achieve good results. Further, the performances of zero-shot hashing are generally better than that of traditional hashing methods. In the worst case, our algorithm is two percentage points higher than the second best method.

CUB dataset: CUB comprises of $11,788$ images, which are collected from various categories of bird. More specially, we select 150 classes and $4,000$ images to train the model. In addition to, we select 50 classes and $4,00$ images for test. The remaining test images together with the images of seen categories are combined to form the retrieval database.

The performances of VSH and the comparative methods on CUB dataset are reported in Fig. 3(c) and Fig. 4(c). Compared with above two datasets, all the measures decline. It is mainly because that CUB is a fine-grained dataset and all vectors in semantic space are close to each other, which leads to the reduction in the discrimination of hash codes. Our approach has the best result in precision@50 and achieves a slightly inferior performance on 32 and 48 bits.

E. Results on ImageNet Dataset

ImageNet is a large-scale dataset and contains 1.2 million images from 1,000 categories. Following the setting of [12], 100 classes are selected to evaluate the model. In this section, we aim to explore the effect of the number of seen categories on zero-shot hashing. Precisely, the number of seen classes varies from 20 to 80. Relatively, the number of unseen classes varies from 80 to 20. 375 images per category are chosen to form training set. It is noticed that ImageNet don't include class attributes. Thus, we select the most similar work SitNet as our baseline.

We compare the proposed method with SitNet as shown in Fig. 5. As we can see, with the increase in number of seen classes, the MAP keep rising. In other words, it is convenient to transfer semantic knowledge from seen classes to unseen classes when the number of seen classes is large. Moreover, our algorithm is superior to SitNet in all different numbers of the seen classes.

V. Conclusion and Future Works

A modified two-stage zero-shot hashing framework, named visual-to-semantic hashing(VSH), is proposed in this paper. VSH can learn semantic structure knowledge and output the hash codes effectively. Compared to existing method, VSH utilize a margin-based ranking loss to transferable semantic structure from seen to unseen classes, and adopt a classification module to increase the discriminability of semantic mapping for different semantic mapping vectors. The experiment results show that our algorithm can learn discriminative hash codes and achieve better performance. In the future, we will explore the relationship between hash codes and semantic vectors to seek better mapping patterns, which can better transfer semantic knowledge among classes and learn semantic-preserving hash codes.

VI. Acknowledgements

This work is supported by NSFC (61702188), and STCSM (19ZR1414200).

References

[1] J. Song, Y. Yang, Y. Yang, et al. "Inter-media hashing for large-scale retrieval from heterogeneous data sources." Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data. 2013: 785-796.

[2] J. Wang, W. Liu, S. Kumar and S. F. Chang, "Learning to hash for indexing big data - A Survey," Proceedings of the IEEE, vol. 104, pp. 34–57, 2015.

[3] Y. Weiss, A. Torralba and R. Fergus, "Spectral hashing," Advances in Neural Information Processing Systems, pp. 1753–1760, 2008.

[4] W. Liu, C. Mu, S. Kumar and S. F. Chang, "Discrete graph hashing," Advances in Neural Information Processing Systems, pp. 3419–3427, 2014.

[5] Y.C. Gong, S. Lazebnik, A. Gordo and F. Perronnin, "Iterative Quantization: A procrustean approach to learning binary codes for large-scale image retrieval," IEEE Trans. Pattern Anal. Mach. Intell, vol. 35, pp. 2916–2929, 2013.

[6] J. Wang, S. Kumar, and S. F Chang, "Semi-supervised hashing for large-scale search," IEEE Trans. Pattern Anal. Mach. Intell, vol. 34, pp. 2393–2406, 2012.

[7] J. Wang, S. Kumar, and S. F Chang, "Sequential projection learning for hashing with compact codes," Proceedings of the 27th International Conference on Machine Learning, pp. 1127–1134, 2010.

[8] F. M. Shen, C. H. Shen, W. Liu and H. T. Shen, "Supervised discrete hashing," Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp. 37–45, 2015.

[9] H. M. Liu, R. P. Wang, S. G. Shan and X. L. Chen, "Deep supervised hashing for fast image retrieval," Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp. 2064–2072, 2016.

[10] Z. J. Cao, M. S. Long, J. M. Wang and P. S. Yu, "HashNet: Deep learning to hash by continuation," Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp. 5608–5617, 2017.

[11] C. H. Lampert, H. Nickisch and S. Harmeling "Learning to detect unseen object classes by between-class attribute transfer," 2019

[12] Y. Q. Xian, C. Lampert, B. Schiele and Z. Akata, "Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol 41, pp. 2251-2265, 2018.

[13] Y. Yang, Y. D. Luo, W. L. Chen, F. N. Shen, J. Shao and H. T. Shen, "Zero-shot hashing via transferring supervised knowledge," Proceedings of the 24th ACM International Conference on Multimedia, pp. 1286-1295, 2016.
[14] Y. C. Guo, G. G. Ding, J. G. Han and Y. Gao, "SitNet: Discrete similarity transfer network for zero-shot hashing," Proceedings of the 26th International Joint Conference on Artificial Intelligence, pp. 1767-1773, 2017.
[15] Z. Ji, Y. X. Sun, Y. L. Yu and Y. Gao, "Attribute-guided network for cross-modal zero-shot hashing," IEEE Transactions on Neural Networks and Learning Systems, vol 31, pp. 321-330, 2020.
[16] H. J. Lai, "Transductive zero-shot hashing via coarse-to-fine similarity mining," Proceedings of ACM on International Conference on Multimedia Retrieval, pp. 196–203, 2018.
[17] P. INDYK, "Approximate nearest neighbor : Towards removing the curse of dimensionality," Proceedings of the 30th Symposium on Theory of Computing, pp. 604–613, 1998.
[18] R. K. Xia, Y. Pan, H. J. Lai, C. Liu and S. C. Yan, "Supervised hashing for image retrieval via image representation learning," The 28th Association for the Advance of Artificial Intelligence, pp. 2156-2162, 2014.
[19] L. Liu. F. M. Shen, Y. M. Shen, X. L. Liu, and L. Shao, "Deep sketch hashing: Fast free-hand sketch-based image retrieval," Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp. 2862–2871, 2017.
[20] F. Zhao, Y. Z. Huang, L. Wang, T. N. Tan, "Deep semantic ranking based hashing for multi-label image retrieval," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1556-1564, 2015.
[21] F. M. Shen, X. Zhou, J. Yu, Y. Yang, L. Liu, and H. T. Shen, "Scalable zero-shot learning via binary visual-semantic embeddings," IEEE Transactions on Image Processing, vol 28, pp. 3662-3674, 2019.
[22] X. W. Liu, Z. Li, J. Wang, G. X. Yu, C. Domeniconi and X. L. Zhang, "Cross-modal zero-shot hashing," CoRR, vol abs/1908.07388, 2019.
[23] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, and M. Ranzato and T. Mikolov, "Devise: A deep visual-semantic embedding model," Proceedings of the International Conference on Neural Information Processing Systems, pp. 2121–2129, 2013.
[24] K. M. He, X. Y. Zhang, S. Q. Ren, S. Bengio, and J. Sun, "Deep Residual Learning for Image Recognition," IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778, 2016.
[25] C. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," IEEE Conference on Computer Vision and Pattern Recognition, pp. 951–958, 2009.
[26] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "escribing objects by their attributes," IEEE Conference on Computer Vision and Pattern Recognition, pp. 1778–1785, 2009.
[27] P. Welinder, S. Branson, T. Mita, C. Wah, F.Schroff, S. Belongie and P. Perona, "escribing objects by their attributes," California Institute of Technology, 2010.
[28] J. Deng, W. Dong, R. Socher, and L. J. Li, K. Li and F. F. Li "ImageNet: A large-scale hierarchical image database," IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255, 2009.
[29] G. A. Miller, "WordNet: a lexical database for English," Communications of the ACM, pp. 39–41, 1995.
[30] X. Glorot, Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," Proceedings of the 13th International Conference on Artificial Intelligence and Statistics, pp. 249–256, 2010.
[31] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," International Conference on Learning Representations, 2015.