VSB2-Net: Visual-Semantic Bi-Branch Network for Zero-Shot Hashing

Xin Li, Xiangfeng Wang, Bo Jin, Wenjie Zhang, Jun Wang and Hongyuan Zha
Outline

• Introduction
• Motivation
• Framework
• Experiments
Introduction

• The definition between traditional hashing and zero-shot hashing
Motivation

• The existing methods mainly focused on optimizing the mapping between hash codes and semantic space, but ignored the core of the hash problem that generates discriminative hash codes.

• The hash codes are binary-like, and have lower representative ability compared with the high-dimensional visual feature vectors, which makes it difficulty to distinguish similar classes in hamming space.
The architecture is a bi-branch network, which includes the semantic similarity branch and the visual feature transfer branch.

The reconstruction module and classification module are directly employed to enhance the generalization and transfer abilities on unseen classes.
Framework

• Semantic Similarity Branch Network
  • Computing cosine similarity between semantic word vectors
  • Applying $K$ nearest neighbors technique to separate all $L$ training word vectors into $T$ groups
  • utilizing a barycenter-based fisher criteria to maximize the distance between two word vector groups and maintain the similarity relationships within each group
• Loss function is designed as follows

$$
\mathcal{L}_s = \alpha \sum_{i=1}^{T} \sum_{w_{i}, w_{j} \in N_i} \left\| \text{Sim}(w_{i}, w_{j}) - \text{Sim}(s_{i}, s_{j}) \right\|^2 
+ \beta \sum_{t_1=1}^{T} \sum_{t_2=1, t_2 \neq t_1}^{T} \max \left( 0, \lambda - \left\| \text{BC}(N_{t_1}) - \text{BC}(N_{t_2}) \right\|^2 \right) 
+ \gamma \sum_{i=1}^{T} \left\| s_{i} - e \right\|_1,
$$
Framework

- Visual Feature Transfer Branch Network
  - employing dot product metric between hash code and the target semantic vector to characterize the similarity relationships
- Loss function is designed as follows

\[
\mathcal{L}_t = \sum_{i=1}^{N} \max \left( 0, m - b_i^T \cdot s^{\ell_i} + \max_{\ell_j \neq \ell_i} b_i^T \cdot s^{\ell_j} \right)
\]
Framework

• Task-driven Regularization
  • reconstruction module: push hash codes reconstructing the visual features
  • classification module: drive the error hash codes to near the target semantic vector in order to reduce ambiguity
• Loss function is designed as follows

\[ \mathcal{L}_t = \sum_{i=1}^{N} \max \left( 0, m - b_i^T \cdot s_i + \max_{\ell_j \neq \ell_i} b_i^T \cdot s_{\ell_j} \right) \]
Experiments

(a) AwA

(b) APY

(c) CUB
Experiments

Precision-recall curves (64bits) of VSB²-Net on AwA dataset

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

- MAP of two mapping modes w.r.t different numbers of bits on AwA.

<table>
<thead>
<tr>
<th>Method</th>
<th>16 bits</th>
<th>32 bits</th>
<th>48 bits</th>
<th>64 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mini-SitNet</td>
<td>0.128</td>
<td>0.150</td>
<td>0.175</td>
<td>0.192</td>
</tr>
<tr>
<td>Mini-VSB²-Net</td>
<td>0.151</td>
<td>0.178</td>
<td>0.220</td>
<td>0.241</td>
</tr>
</tbody>
</table>

- MAP of two reconstructive space methods w.r.t different numbers of bits on AwA.

<table>
<thead>
<tr>
<th>Method</th>
<th>16 bits</th>
<th>32 bits</th>
<th>48 bits</th>
<th>64 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>VSB²-G</td>
<td>0.155</td>
<td>0.198</td>
<td>0.217</td>
<td>0.270</td>
</tr>
<tr>
<td>VSB²-Net</td>
<td>0.175</td>
<td>0.230</td>
<td>0.272</td>
<td>0.293</td>
</tr>
</tbody>
</table>

- Parameter verification with 64-bit hash codes on AwA.

<table>
<thead>
<tr>
<th>MAP \ β</th>
<th>α \ 1</th>
<th>0.1</th>
<th>0.01</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.280</td>
<td>0.285</td>
<td>0.274</td>
</tr>
<tr>
<td>0.1</td>
<td>0.286</td>
<td><strong>0.292</strong></td>
<td>0.264</td>
</tr>
<tr>
<td>0.01</td>
<td>0.274</td>
<td>0.281</td>
<td>0.266</td>
</tr>
</tbody>
</table>
Thank you!