Hugs Are Better Than Handshakes: Unsupervised Cross-Modal Transformer Hashing with Multi-granularity Alignment

Jinpeng Wang[1,3],Ziyun Zeng[1,3]
{wjp20,zengzy21}@mails.tsinghua.edu.cn
Bin Chen[2,3†]
chenbin2021@hit.edu.cn
Yuting Wang[1,3]
huangmozhi9527@gmail.com
Dongliang Liao[4†],Gongfu Li[4],Yiru Wang[4]
{brightliao,gongfuli,dorisyrwang}@tencent.com
Shu-Tao Xia[1,3]
xiast@sz.tsinghua.edu.cn

[1] Tsinghua Shenzhen International Graduate School, Tsinghua University, China
[2] Harbin Institute of Technology, Shenzhen, China
[3] Research Center of Artificial Intelligence, Peng Cheng Laboratory, China
[4] Wechat Group, Tencent Inc., China

Abstract

The goal of unsupervised cross-modal hashing (UCMH) is to map different modalities into a semantic-preserving hamming space without requiring label supervision. Existing deep approaches mainly took classic CNNs and multilayer perceptrons to encode images and texts, which are inadequate for semantic extraction and hard to generate high-quality hash codes. Motivated by recent advances in transformers, we take the first investigation of transformer-based UCMH that learns to generate hash codes via global representation (*i.e.*, "[CLS]") tokens. We propose *hugging*, a multi-granularity aligning framework for transformer-based UCMH learning. In particular during training, apart from directly aligning hash codes from global tokens, *hugging* further develops fine-grained alignment based on content token sequences, which fully exploits the structural semantics contained in transformer architectures. Unifying global and fine-grained alignment enables complete cross-modal learning, helping to bridge heterogeneous modality gaps and providing solid self-supervision. As an instantiation of the proposed *hugging* framework, we build a simple HUGGINGHASH model with a contrastive hashing learning objective and demonstrate its comprehensive merits on three benchmark datasets. Moreover, we also adapt several state-of-the-art hashing methods using the *hugging* framework, verifying that it can be general and practical to benefit transformer-based UCMH.

1 Introduction

Unsupervised cross-modal hashing (UCMH) is a practical task that learns to generate binary representations for different modalities (*e.g.* images and texts) without requiring label

†: Corresponding authors.

information. It has been a popular indexing strategy for large-scale multimedia data because the Hamming descriptors can accelerate cross-modal retrieval with fast XOR operators [3, 53, 55, 64]. The quality of hash representations is inherently subject to multimedia understanding. Although neural networks have made remarkable progress in hashing, the advances of deep learning have yet to be fully exploited. State-of-the-art approaches [18, 26, 50, 56] mainly used classic convolutional neural networks (CNNs), *e.g*. VGGNet [48] and AlexNet [23], to extract visual features and used multilayer perceptrons (MLPs) to encode text information. These designs are sub-optimal to capture semantics from visual and linguistic data, and they also suffer from limited transferability. To improve UCMH and keep pace with the development of deep learning, one promising direction is to explore transformer-based methodology.

In the past few years, transformers [52] have shown excellent talents in computer vision [4, 9, 37] and natural language processing [7, 36, 62] tasks, sparking explorations toward better machine understanding. Pretrained on large-scale corpora [23, 72], transformers can serve as versatile experts that are effective and generalizable for various downstream tasks. Lately, there have been some attempts in transformer-based image hashing [6, 10, 13, 29, 39] and video hashing [15, 28], which have shown impressive results. However, transformer-based cross-modal hashing remains under-explored. *How can UCMH benefit from transformers? How can we make better use of transformers for cross-modal hashing?* These questions motivate our study.

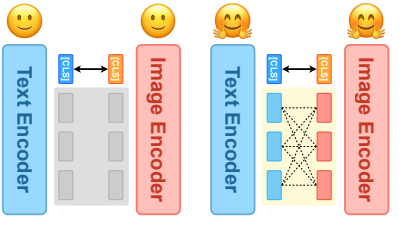


Figure 1: Alignment paradigms for transformer-based UCMH. Compared with *Handshaking*, *Hugging* further exploits fine-grained alignment based on the content tokens. Unifying multi-granularity alignment provides solid self-supervision for cross-modal hashing. Note that fine-grained alignment is an auxiliary task and is removed after finishing training. Thus, it will *not* increase extra inference overhead.

Although pre-trained transformers provide solid semantic extraction for each modality, UCMH is still non-trivial. The main challenge is to bridge heterogeneous modalities so that the hash codes can be well aligned. Analogous to existing CNN-MLP-based UCMH models, we can train transformer-based UCMH models to produce hash codes via the global representation (*i.e.*, "[CLS]") tokens. A simple way for learning is to align the global tokens using the objectives in existing UCMH methods. We liken this global alignment strategy to "*handshaking*", as shown in Figure 1(a). In practice, handshaking is effective as expected but can be improved to reduce the modality gap. Note that transformer is a sequential architecture that arranges inputs as sequences. It naturally provides a set of content tokens (*e.g*. words of a text or patches of an image) with fine-grained and structural semantics, which can capture heterogeneous modality knowledge but was usually overlooked.

To enhance transformer-based UCMH learning, we present a multi-granularity alignment framework dubbed *hugging*, as illustrated in Figure 1(b). Besides aligning the hashing representations from the "[CLS]" tokens, we further develop a fine-grained alignment mechanism based on the content tokens. In particular, we construct another shared latent space with semantic structure via a GhostVLAD [69] module. Each content token in this space is softly assigned to a series of parameterized clusters, each representing a latent topic or semantic concept. Cluster-wise contrastive [5, 14] alignment serves as an auxiliary objective for model training, which enhances the cross-modal alignment and effectively im-

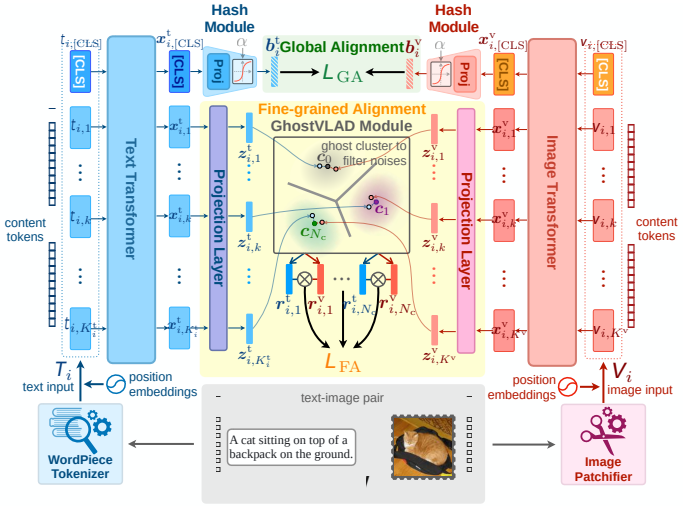


Figure 2: The HUGGINGHASH model. During training, it unifies global alignment based on hash codes and fine-grained alignment using GhostVLAD [69] based on content tokens. In inference, we deactivate the fine-grained alignment so that it will not impose an extra cost.

proves the cross-modal consistency of the learned hash codes. After training, we remove the GhostVLAD module and the fine-grained alignment such that there is *no* extra inference overhead. We instantiate the hugging framework by building a simple HUGGINGHASH model with contrastive learning for global alignment. Experiments show that HUGGINGHASH outperforms state-of-the-art UCMH methods integrating transformers in *handshaking* style. Moreover, we also adapt several state-of-the-art UCMH methods with the proposed *hugging* framework, demonstrating its general effectiveness for transformer-based UCMH.

Contribution summary: (**i**) To our knowledge, we are the first to study transformer-based UCMH, providing a research basis for this promising direction. (**ii**) We propose *hugging* that unifies multi-granularity alignment for transformer-based UCMH. (**iii**) Extensive experiments verify the effectiveness of *hugging*, and it is also compatible with state-of-the-arts.

2 The Proposed Method

2.1 Problem Formulation and Method Overview

Without loss of generality, we study text-image hashing as an example of cross-modal hashing. Given an unlabeled training set D of N_D naturally coexisted text-image pairs, our goal is to learn a pair of modality-specific hash encoders that encode texts and images as L-bit semantic-preserving binary codes for efficient cross-modal retrieval.

To this end, we construct a HUGGINGHASH model using the *hugging* framework, as illustrated in Figure 2. Specifically, given a training pair, we first preprocess the text and the image as the input tokens for transformers. Then, we extract features with transformers and get the output embeddings from the [CLS] and content tokens (§2.2). Next, we forward the [CLS] tokens to the hash modules and produce text and image hash code vectors. Meanwhile, we project the embeddings of content tokens to a cross-modal latent space. Finally, we

conduct multi-granularity alignment, including the global alignment based on the hash codes and the structural alignment based on the latent content embeddings, to bridge text and image modalities (§2.3). In inference, we deactivate the components for structural alignment and directly produce the hash codes through the global branch (§2.4).

2.2 Feature Extraction with Transformers

For each text sample, we tokenize it into word pieces and construct content tokens. Then we append a [CLS] token and form the text input. Denote the token sequence of the i-th text in a mini-batch B by $T_i = \{t_{i,[\mathrm{CLS}]}, t_{i,1}, t_{i,2}, \ldots, t_{i,K_i^{\mathrm{t}}}\}$, where K_i^{t} is the number of content tokens for the i-th text. We pad the sequence to a fixed length, add position embeddings and forward it to the BERT [7] encoder $f^{\mathrm{t}}(\cdot; \boldsymbol{\theta}_f^{\mathrm{t}})$ to compute the token embeddings, namely $\boldsymbol{x}_{i,[\mathrm{CLS}]}^{\mathrm{t}} \in \mathbb{R}^{D^{\mathrm{t}}}$ and $\{\boldsymbol{x}_{i,k}^{\mathrm{t}}\}_{k=1}^{K_i^{\mathrm{t}}} \in \mathbb{R}^{D^{\mathrm{t}}}$. We can formulate the whole process by

$$\boldsymbol{x}_{i,k}^{\mathrm{t}} = f^{\mathrm{t}}(T_i; \boldsymbol{\theta}_f^{\mathrm{t}})_k \in \mathbb{R}^{D^{\mathrm{t}}}, k = [\mathrm{CLS}], 1, 2, \ldots, K_i^{\mathrm{t}}. \tag{1}$$

For each image sample, we use the ViT [9] pre-processor to patchify it into a fixed number (*e.g.* 256) of content tokens and add a [CLS] token to form the image input. We denote the token sequence of the i-th image in a mini-batch B by $V_i = \{v_{i,[\mathrm{CLS}]}, v_{i,1}, v_{i,2}, \ldots, v_{i,K^{\mathrm{v}}}\}$, where K^{v} is the number of content tokens. We then add position embeddings and forward them to the ViT $f^{\mathrm{v}}(\cdot; \boldsymbol{\theta}_f^{\mathrm{v}})$ to compute embeddings, namely $\boldsymbol{x}_{i,[\mathrm{CLS}]}^{\mathrm{v}} \in \mathbb{R}^{D^{\mathrm{v}}}$ and $\{\boldsymbol{x}_{i,k}^{\mathrm{v}}\}_{k=1}^{K^{\mathrm{v}}} \in \mathbb{R}^{D^{\mathrm{v}}}$. Analogous to the text side, we summarize the image feature extraction process by

$$\boldsymbol{x}_{i,k}^{\mathrm{v}} = f^{\mathrm{v}}(V_i; \boldsymbol{\theta}_f^{\mathrm{v}})_k \in \mathbb{R}^{D^{\mathrm{v}}}, k = [\mathrm{CLS}], 1, 2, \ldots, K^{\mathrm{v}}. \tag{2}$$

2.3 *Hugging*: Multi-granularity Alignment for Training

Though transformers provide better understanding, aligning heterogeneous knowledge between texts and images is still challenging for hashing learning. We present *hugging*, a multi-granularity alignment framework to tackle it. In addition to the global alignment based on the hash codes, we design a fine-grained and structural alignment using GhostVLAD [69] based on the content tokens. The global alignment provides direct guidance on hash codes, while the structural alignment supplies fine-grained supervision to reduce the modality gap.

2.3.1 Global Alignment

We apply global alignment to the hash codes. First, we project and convert the output embeddings of the aggregation tokens (*i.e.*, the [CLS] tokens) into binary hash codes:

$$\boldsymbol{h}_i^{\mathrm{t}} = \tanh\left(\alpha \left(f^{\mathrm{t}}(\boldsymbol{x}_{i,[\mathrm{CLS}]}^{\mathrm{t}}) \in [-1, +1]^L\right); \quad f^{\mathrm{t}}: \mathbb{R}^{D^{\mathrm{t}}} \overset{\boldsymbol{\theta}_f^{\mathrm{t}}}{\mapsto} \mathbb{R}^L\right), \tag{3}$$

$$\boldsymbol{h}_i^{\mathrm{v}} = \tanh\left(\alpha \left(f^{\mathrm{v}}(\boldsymbol{x}_{i,[\mathrm{CLS}]}^{\mathrm{v}}) \in [-1, +1]^L\right); \quad f^{\mathrm{v}}: \mathbb{R}^{D^{\mathrm{v}}} \overset{\boldsymbol{\theta}_f^{\mathrm{v}}}{\mapsto} \mathbb{R}^L\right), \tag{4}$$

$$\boldsymbol{b}_i^{\mathrm{t}} = \boldsymbol{h}_i^{\mathrm{t}} - \mathrm{sg}\left(\boldsymbol{h}_i^{\mathrm{t}} - \mathrm{sgn}(\boldsymbol{h}_i^{\mathrm{t}})\right) \in \{-1, +1\}^L, \tag{5}$$

$$\boldsymbol{b}_i^{\mathrm{v}} = \boldsymbol{h}_i^{\mathrm{v}} - \mathrm{sg}\left(\boldsymbol{h}_i^{\mathrm{v}} - \mathrm{sgn}(\boldsymbol{h}_i^{\mathrm{v}})\right) \in \{-1, +1\}^L, \tag{6}$$

where f^{t} and f^{v} are modality-specific projection layers. $\alpha > 0$ is a scaling factor in controlling the smoothness of the tanh outputs. $\boldsymbol{h}_i^{\mathrm{t}}$ and $\boldsymbol{h}_i^{\mathrm{v}}$ are the smoothed hash codes. $\mathrm{sgn}(\cdot)$ is a

sign function that outputs +1 for positive input and 1 otherwise on each element. sg( ) is the *stop gradient* operator that is the identity function in the forward pass but drops gradient for variables inside it during the backward pass. Eq.(5) and eq.(6) allow us to directly pass the gradient straight through [2] the binary hash codes, *i.e.*, $\boldsymbol{b}_i^{\mathrm{t}}$ and $\boldsymbol{b}_i^{\mathrm{v}}$. In HUGGINGHASH, we adopt the contrastive learning loss [5, 14, 41] for global alignment, namely

$$L_{\mathrm{GA}} = \frac{1}{2jBj}\sum_{i=1}^{jBj}\left(-\log\frac{\exp(M_{ii}=t)}{\sum_{j=1}^{jBj}\exp(M_{ij}=t)} - \log\frac{\exp(M_{ii}=t)}{\sum_{j=1}^{jBj}\exp(M_{ji}=t)}\right); \tag{7}$$

where B denotes a mini-batch. $M_{ij} = \cos(\boldsymbol{b}_i^{\mathrm{t}}; \boldsymbol{b}_j^{\mathrm{v}})$. $t > 0$ is the temperature hyper-parameter.

2.3.2 Fine-grained, Structural Alignment

We present a clustering-based strategy with GhostVLAD [69] for fine-grained alignment. Our basic idea is to exploit concept-aware semantics and enable concept-aware alignment in the latent space. Specifically, we first project the output embeddings of the content tokens[1] into a shared latent space, *i.e.*,

$$\boldsymbol{z}_{i;k}^{\mathrm{t}} = \gamma^{\mathrm{t}}(\boldsymbol{x}_{i;k}^{\mathrm{t}}); k = 1;2; \quad ;K_i^{\mathrm{t}}; \quad \gamma^{\mathrm{t}} : \mathrm{R}^{D^{\mathrm{t}}} \mapsto \mathrm{R}^{D} \tag{8}$$

$$\boldsymbol{z}_{i;k}^{\mathrm{v}} = \gamma^{\mathrm{v}}(\boldsymbol{x}_{i;k}^{\mathrm{v}}); k = 1;2; \quad ;K^{\mathrm{v}}; \quad \gamma^{\mathrm{v}} : \mathrm{R}^{D^{\mathrm{v}}} \mapsto \mathrm{R}^{D}; \tag{9}$$

where γ^{t} and γ^{v} are the projection layers for texts and images, respectively. D is the dimension of the shared latent space. We denote the collections of latent embeddings for text and image content tokens as $Z_i^{\mathrm{t}} = \{\boldsymbol{z}_{i;k}^{\mathrm{t}}\}_{k=1}^{K_i^{\mathrm{t}}}$ and $Z_i^{\mathrm{v}} = \{\boldsymbol{z}_{i;k}^{\mathrm{v}}\}_{k=1}^{K^{\mathrm{v}}}$, respectively.

After that, we use a GhostVLAD module to learn $N_{\mathrm{c}} + 1$ D-dimensional cluster centroids, $\{\boldsymbol{c}_0; \boldsymbol{c}_1; \boldsymbol{c}_2; \quad ; \boldsymbol{c}_{N_{\mathrm{c}}}\}$. Specially, we designate $\boldsymbol{c}_0$ as the "ghost" centroid to filter noise, *e.g.* uninformative words in a sentence and background features for an image. We forward Z_i^{t} and Z_j^{v} to the GhostVLAD, where each embedding will be softly assigned to all clusters. For instance, the assignment score of the text embedding $\boldsymbol{z}_{i;k}^{\mathrm{t}}$ *w.r.t.* the n-th cluster is

$$a_{i;k;n}^{\mathrm{t}} = \frac{\exp(\mathrm{BatchNorm}(\boldsymbol{w}_n^{\top}\boldsymbol{z}_{i;k}^{\mathrm{t}}))}{\sum_{n'=0}^{N_{\mathrm{c}}}\exp(\mathrm{BatchNorm}(\boldsymbol{w}_{n'}^{\top}\boldsymbol{z}_{i;k}^{\mathrm{t}}))}; \tag{10}$$

where BatchNorm( ) is the batch normalization [21] and $\boldsymbol{W} = [\boldsymbol{w}_0; \boldsymbol{w}_1; \quad ; \boldsymbol{w}_{N_{\mathrm{c}}}]$ is the trainable parameter matrix. After clustering, we aggregate modality-wise residual embeddings at each cluster except the "ghost" cluster. Take the n-th cluster as an example. We aggregate residual embeddings of Z_i^{t} and Z_j^{v} respectively by

$$\boldsymbol{r}_{i;n}^{\mathrm{t}} = \sum_{k=1}^{K_i^{\mathrm{t}}} a_{i;k;n}^{\mathrm{t}}\,(\boldsymbol{z}_{i;k}^{\mathrm{t}} \quad \boldsymbol{c}_n); \quad \boldsymbol{r}_{j;n}^{\mathrm{v}} = \sum_{k=1}^{K^{\mathrm{v}}} a_{j;k;n}^{\mathrm{v}}\,(\boldsymbol{z}_{j;k}^{\mathrm{v}} \quad \boldsymbol{c}_n): \tag{11}$$

Finally, we define the cluster-wise contrastive learning loss for fine-grained alignment as

$$L_{\mathrm{FA}} = \frac{1}{2N_{\mathrm{c}}jBj}\sum_{n=1}^{N_{\mathrm{c}}}\sum_{i=1}^{jBj}\left(-\log\frac{\exp(m_{ii}^{n}=t)}{\sum_{j=1}^{jBj}\exp(m_{ij}^{n}=t)} - \log\frac{\exp(m_{ii}^{n}=t)}{\sum_{j=1}^{jBj}\exp(m_{ji}^{n}=t)}\right); \tag{12}$$

[1] The text padding tokens have been removed from the outputs.

where $m_{ij}^n = \cos(\boldsymbol{r}_{i;n}^{\mathrm{t}}; \boldsymbol{r}_{j;n}^{\mathrm{v}})$ is the fine-grained similarity of Z_i^{t} and Z_j^{v} *w.r.t.* the n-th cluster. B denotes a training mini-batch and t denotes the temperature hyper-parameter.

2.3.3 Learning Objectives

Here we summarize the learning objectives of HUGGINGHASH as follows:

$$L_{\text{HUGGINGHASH}} = L_{\text{GA}} + l\, L_{\text{FA}} + g R_{\text{quant}}; \tag{13}$$

$$R_{\text{quant}} = \frac{1}{2LjBj} \sum_{i=1}^{jBj} \boldsymbol{b}_i^{\mathrm{t}} \quad \boldsymbol{h}_i^{\mathrm{t}} \quad {}_2^2 + k\boldsymbol{b}_i^{\mathrm{v}} \quad \boldsymbol{h}_i^{\mathrm{v}} k_2^2 \; : \tag{14}$$

R_{quant} is the quantization loss. $l\,; g > 0$ are the hyper-parameters to balance different loss terms. The proposed *hugging* framework is highly flexible and compatible. By replacing L_{GA} with other hashing objectives, we can easily extend *hugging* to existing methods.

2.4 Indexing and Retrieval

We take the text-to-image retrieval as an example to describe how HUGGINGHASH inferences. First, we encode the database images with the image hash encoder, which comprises the patchifier, the image transformer and the image hash module. We denote the hash codes of the i-th image by $\boldsymbol{b}_i^{\mathrm{v}}$ 2 f 1; + 1g^L. Given a text query, we forward it with the text hash encoder, which comprises the tokenizer, the text transformer and the text hash module. We denote the query hash codes as $\boldsymbol{b}_{\mathrm{q}}^{\mathrm{t}}$ 2 f 1; + 1g^L. We use it to retrieve the nearest database images in the Hamming space. The Hamming distance between $\boldsymbol{b}_{\mathrm{q}}^{\mathrm{t}}$ and $\boldsymbol{b}_i^{\mathrm{v}}$ is

$$d_{\mathbb{H}}(\boldsymbol{b}_{\mathrm{q}}^{\mathrm{t}}; \boldsymbol{b}_i^{\mathrm{v}}) = \frac{1}{2} \quad L \quad \boldsymbol{b}_{\mathrm{q}}^{\mathrm{t}>} \boldsymbol{b}_i^{\mathrm{v}} \; : \tag{15}$$

3 Experiments

3.1 Experimental Setup

Datasets. We conduct experiments on three benchmark datasets in cross-modal hashing: (**i**) **Flickr25K** [20] contains 25,000 image-text pairs with 24 annotated labels. We filtering out data without labels and use 20,015 pairs in our experiment. The tag information for each image is represented as a 1,386-dimensional bag-of-words vector. (**ii**) **MSCOCO** [31] consists of 123,558 image-sentence pairs from 80 object categories. Each image is associated with 4 short sentences describing its content. The text information is represented as a 2,000-dimensional bag-of-words vector. (**iii**) **Wiki** [44] composes of 2,866 documents from 10 categories. Each document contains an image and a text with at least 70 words. An 128-dimensional SIFT feature vector is provided for each image, and each text is represented as a 10-dimensional topic vector. The data split information is described in Table 1.

Implementation Details. We implement HUGGINGHASH with PyTorch [42]. We adopt the standard metric, mean average precision (**MAP@N**), for evaluation. For comparison, shallow methods

Table 1: Dataset information and settings.

Dataset	Text Style	Setting Reference	#Train	#Query	#Database	Metric
Flickr25K [20]	Hashtags	Li et al. [44]	5,000	2,000	18,015	MAP@All
MSCOCO [31]	Sentence	Wang et al. [51]	122,558	1,000	122,558	MAP@All
Wiki [44]	Article	Wang et al. [51]	2,173	693	2,173	MAP@50

Table 2: Cross-modal retrieval mean average precision (MAP) results for different numbers of bits on the three datasets. 'I2T' and 'T2I' stand for 'Image-to-Text' and 'Text-to-Image' for short, respectively. The best value of each column is shown in boldface.

Method #	Dataset / Venue #	Backbone #	Flickr25K						MSCOCO						Wiki					
			T2I Retrieval			I2T Retrieval			T2I Retrieval			I2T Retrieval			T2I Retrieval			I2T Retrieval		
			16 bits	32 bits	64 bits	16 bits	32 bits	64 bits	16 bits	32 bits	64 bits	16 bits	32 bits	64 bits	16 bits	32 bits	64 bits	16 bits	32 bits	64 bits
CVH [24]	IJCAI'11	Non-Deep	0.607	0.591	0.581	0.602	0.587	0.578	0.507	0.479	0.446	0.499	0.471	0.443	0.252	0.235	0.171	0.179	0.162	0.153
IMH [45]	SIGMOD'13		0.586	0.593	0.589	0.588	0.581	0.585	0.413	0.435	0.443	0.416	0.435	0.443	0.467	0.478	0.453	0.201	0.203	0.204
CMFH [8]	TIP'16		0.611	0.606	0.575	0.659	0.660	0.663	0.453	0.435	0.499	0.442	0.423	0.492	0.595	0.601	0.616	0.251	0.253	0.259
FSH [32]	CVPR'17		0.589	0.595	0.595	0.590	0.597	0.597	-	-	-	-	-	-	-	-	-	-	-	-
ACQ [22]	ICCV'17		-	-	-	-	-	-	0.565	0.561	0.520	0.559	0.553	0.515	-	-	-	-	-	-
DBRC [30]	MM'17	CNN + MLP	0.591	0.596	0.598	0.596	0.600	0.602	0.562	0.566	0.573	0.555	0.561	0.564	0.574	0.588	0.598	0.253	0.265	0.269
UGACH [67]	AAAI'18		0.676	0.692	0.703	0.676	0.693	0.702	0.566	0.595	0.607	0.550	0.584	0.599	0.544	0.555	0.572	0.388	0.392	0.403
UCH [26]	AAAI'19		0.661	0.667	0.668	0.654	0.669	0.679	0.446	0.469	0.488	0.447	0.471	0.485	-	-	-	-	-	-
DJSRH [60]	ICCV'19		0.683	0.694	0.717	0.666	0.678	0.699	0.573	0.578	0.584	0.572	0.575	0.579	0.611	0.635	0.646	0.388	0.403	0.412
UKD-SS [18]	CVPR'20		0.704	0.705	0.714	0.700	0.706	0.709	0.580	0.594	0.603	0.564	0.592	0.601	0.556	0.565	0.578	0.403	0.411	0.416
SRCH [56]	IJCAI'20		-	-	-	-	-	-	0.600	0.606	0.623	0.598	0.605	0.623	-	-	-	-	-	-
DSAH [60]	MM'20		0.707	0.713	0.728	0.701	0.712	0.722	0.606	0.599	0.620	0.598	0.589	0.609	0.644	0.650	0.660	0.416	0.430	0.438
DGCPN [63]	AAAI'21		0.729	0.741	0.749	0.732	0.742	0.751	0.594	0.603	0.616	0.587	0.594	0.612	0.629	0.638	0.641	0.422	0.440	0.446
DBRC [30]	MM'17	Transformers	0.628	0.633	0.639	0.627	0.637	0.642	0.592	0.605	0.602	0.594	0.603	0.611	0.591	0.594	0.599	0.449	0.460	0.466
DJSRH [60]	ICCV'19		0.716	0.724	0.725	0.712	0.718	0.723	0.623	0.621	0.627	0.619	0.624	0.627	0.640	0.649	0.652	0.496	0.502	0.511
DSAH [60]	MM'20		0.726	0.729	0.729	0.723	0.727	0.734	0.641	0.636	0.642	0.637	0.639	0.648	0.655	0.661	0.667	0.491	0.489	0.501
DGCPN [63]	AAAI'21		0.743	0.756	0.751	0.745	0.750	0.755	0.638	0.649	0.650	0.633	0.641	0.643	0.651	0.658	0.662	0.489	0.498	0.503
HUGGINGHASH	BMVC'22		**0.745**	**0.760**	**0.766**	**0.752**	**0.758**	**0.764**	**0.652**	**0.661**	**0.663**	**0.646**	**0.653**	**0.662**	**0.659**	**0.669**	**0.675**	**0.522**	**0.520**	**0.526**

take bag-of-words features and hand-crafted visual descriptors (*e.g.* SIFT [38]) as text and image inputs, respectively. Deep methods use CNN (*e.g.* AlexNet [23]) features as image inputs. For transformer-based methods, we use pretrained BERT [7] ('bert-base-uncased') and ViT [9] ('vit-base-patch16-224') as default transformers, where the token embedding dimensions are $D^{\mathrm{t}} = D^{\mathrm{v}} = 768$. The maximum number of text tokens is 128. The the dimension of fine-grained alignment space is $D = 128$. Other default settings are as follow: (**i**) The loss weights in eq.(13) are $l = 0.2$ and $g = 1$. (**ii**) The scaling factor in eq.(3) and eq.(4) is $a = 0.5$. (**iii**) The temperature factor in eq.(7) and eq.(12) is $t = 0.2$. (**iv**) The number of active cluster in GhostVLAD is $N_{\mathrm{c}} = 7$.

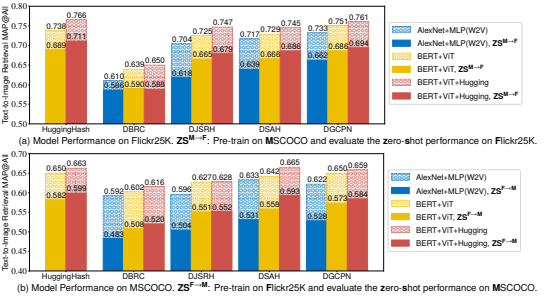
3.2 Comparison with Existing Methods

Performance. Table 2 reports the MAP results under different numbers of hash bits. The comparison is with 13 UCMH baselines: (**i**) 5 shallow methods: CVH [24], IMH [45], CMFH [8], FSH [32], ACQ [22]. (**ii**) 8 SOTA deep methods: DBRC [30], UGACH [67], UCH [26], DJSRH [60], UKD-SS [18], SRCH [56], DSAH [60], DGCPN [63]. To explore the impact of transformers on UCMH, we adapt open-sourced implementations of 4 representative baselines, *i.e.*, DBRC, DJSRH, DSAH, and DGCPN, by using the same backbones as HUGGINGHASH. The same methods in the 'Transformers' block outperform those in the 'CNN + MLP' block by considerable margins. It verifies that pre-trained transformers provide better modality understandings than CNNs and MLPs, thus contributing to high-quality hash codes. Besides, on all settings, HUGGINGHASH outperforms transformer-based baselines that only consider global alignment. In terms of global alignment for hash codes, although HUGGINGHASH adopts a simple contrastive learning objective (*i.e.*, eq.(7)) that is much simpler than the baselines, the superior results it shows indicate the effectiveness of exploring multi-granularity alignment with transformers beyond global alignment itself.

Transferability. Transferability is an important but often ignored target in practice, reflecting the domain generalizability from offline training to online serving. Here we conduct a multi-dataset (Flickr25K-MSCOCO) evaluation with DBRC, DJSRH, DSAH, DGCPN, and the proposed HUGGINGHASH. We compare *standard* (*i.e.*, train and test on the same dataset) and *zero-shot* performance (*i.e.*, train and test on different datasets). We also investigate different backbones. To deal with different vocabularies between datasets, we replace the bag-of-words features with the word2vec [25] features as the text inputs for vanilla variants

("AlexNet+MLP(W2V)"). "BERT+ViT" indicates transformer-based variants using global alignment. "BERT+ViT +Hugging" indicates the variants with *hugging*.

The results are shown in Figure 3. We can learn that transformers not only boost in-domain performance but also improve generalizability. Using the proposed *hugging* framework can further improve the zero-shot performance in most cases. Besides, although combining *hugging* with SOTA baselines yield competitive results with HUGGINGHASH, *e.g.* DASH on MSCOCO, we notice that HUGGINGHASH shows the best zero-shot performance on both datasets. The reason is that the contrastive learning objectives help to produce more transferable hash codes.



(a) Model Performance on Flickr25K. $\mathbf{ZS}^{\mathbf{M}\to\mathbf{F}}$: Pre-train on **M**SCOCO and evaluate the zero-**s**hot performance on **F**lickr25K.

(b) Model Performance on MSCOCO. $\mathbf{ZS}^{\mathbf{F}\to\mathbf{M}}$: Pre-train on Flickr25K and evaluate the zero-**s**hot performance on **M**SCOCO.

Figure 3: Multi-dataset (Flickr25K-MSCOCO) evaluation with different 64-bit UCMH methods. Transformers and the *hugging* improve generalizability and robustness. Besides, HUGGINGHASH shows the best zero-shot results.

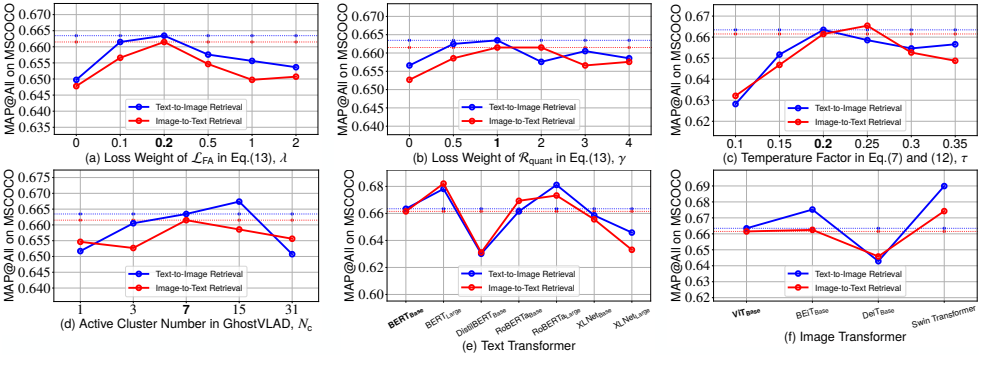


Figure 4: Parameter Sensitivities of a 64-bit HUGGINGHASH model on MSCOCO. Default settings are marked in bold. The dotted lines mark the MAP results under default settings.

3.3 Model Analyses

Effects of loss terms. We analyze HUGGINGHASH on MSCOCO since it is a benchmark dataset for vision-language tasks. λ is the weight of fine-grained alignment loss L_{FA} that essentially controls its task gradient strength in the learning process. Adjusting λ from 0 to 0.2 can boost the performance, verifying that L_{FA} is beneficial. However, we can see the gain drops as we increase λ beyond 0.2. This is because the auxiliary task of fine-grained alignment dominates the learning and even adversely constricts the main task of aligning hash codes. γ controls the strength of regularization R_{quant}. A proper range for γ is 0.5 1.

Effects of τ and N_c. Figure 4(c) illustrates the effect of the temperature τ factor in contrastive learning. A suitable range for τ is 0.2 0.25. Figure 4(d) shows the effect of active cluster (*i.e.*, latent concept) number N_c in GhostVLAD. 15 and 7 are reasonable choices for N_c. We set $N_c = 7$ by default for the higher training efficiency.

Effects of used transformers. We equip HUGGINGHASH with different transformers [1, 36, 37, 46, 51, 62]. Figure 4(e) and (f) show that large BERT [7] and RoBERTa [36] variants are good choices for the text transformer. Swin Transformer [37] is a good choice for image encoder.

***Hugging* with existing methods.** Apart from Figure 3, we demonstrate the effectiveness of transformers and the *hugging* framework on SOTA methods in Table 3. We can see consistent improvements. Besides, SOTA methods surpass HUGGINGHASH when equipped with *hugging*. This is reasonable because they use more complex global alignment mechanisms.

Table 3: Retrieval mean average precision (MAP@All) results on MSCOCO. 'Use Tr' means whether the variant uses transformers.

Method	Use Tr	*Hugging*	T2I Retrieval			I2T Retrieval		
			16 bits	32 bits	64 bits	16 bits	32 bits	64 bits
DBRC [30]			0.562	0.566	0.573	0.555	0.561	0.564
	✓		0.592	0.605	0.602	0.594	0.603	0.611
	✓	✓	0.611	0.613	0.616	0.610	0.619	0.630
DJSRH [50]			0.573	0.578	0.584	0.572	0.575	0.579
	✓		0.623	0.621	0.627	0.619	0.624	0.627
	✓	✓	0.637	0.626	0.628	0.630	0.634	0.637
DSAH [60]			0.606	0.599	0.620	0.598	0.589	0.609
	✓		0.641	0.636	0.642	0.637	0.639	0.648
	✓	✓	0.656	0.659	0.665	0.669	0.663	0.670
DGCPN [65]			0.594	0.603	0.616	0.587	0.594	0.612
	✓		0.638	0.649	0.650	0.633	0.641	0.643
	✓	✓	0.645	0.662	0.659	0.644	0.650	0.658
HUGGINGHASH	✓		0.633	0.644	0.650	0.631	0.636	0.644
	✓	✓	0.652	0.661	0.663	0.646	0.653	0.662

4 Related Work

Unsupervised Cross-modal Hashing (UCMH). Traditional UCMH methods learned to transform hard-craft features [38] into binary codes by solving linear problems, *e.g.* matrix factorization [8, 19, 70] and spectral decomposition [24, 49]. The shallow features and linear solutions limited the performance. In contrast, by leveraging deep neural networks (DNNs), deep UCMH methods can capture richer semantic information and generate better hash codes. Early deep methods [16, 17, 59] replaced the hand-crafted features with the deep features and applied linear solutions as in some shallow methods [19, 24, 49, 70]. To better estimate pairwise similarity to guide hashing learning, later deep methods [33, 50, 56, 60, 63, 68] fused similarity matrices from different modalities during training. Besides, some recent deep methods tried to narrow the modality gap by using adversarial learning [26, 66, 67] or knowledge distillation [18, 27], showing promising results. Note that existing deep methods mainly used classic VGGNets [48] or AlexNet [23] to extract visual features and used MLPs to encode text information, which suffered from inadequate semantic extraction and limited the representation quality. Instead, we take the first step to study the effectiveness of transformers on UCMH, which is a good practice learned from the recent successes in the deep learning community. We also explore how to use transformers for UCMH better.

Transformers in Multimedia Retrieval. Recently, transformers [52] have made remarkable progress in CV [4, 9, 37] and NLP [7, 36, 62] tasks, triggering the surge toward better multimedia understanding. In cross-modal retrieval, the great potential of transformers has also been explored [47]. Most dual-stream methods [11, 12, 34, 43, 45, 58, 61] aligned the global representation tokens (*e.g.* "[CLS]") with metric learning [35] or contrastive learning [5]. To reduce the modality gap further, several works [40, 54, 57] proposed to align fine-grained representations extracted from the content tokens, showing better performance but lower efficiency due to the complex similarity computations in inference. Different from them, we design fine-grained alignment as an *auxiliary task* to improve hashing learning and *remove it in inference*. It demonstrates effectiveness while maintaining efficiency.

In the specific field of hashing-based retrieval, recent advances in image [6, 10, 13, 29, 39] and video hashing [15, 28, 65] have also revealed the effectiveness of uni-modal transformer hashing. In contrast, the cross-modal scenario is still under-explored. We believe our work can fill the blank and serve as a research basis for this promising direction.

5 Conclusions

This paper studies the novel and practical problem of transformer-based unsupervised cross-modal hashing (UCMH). We propose a *hugging* framework that unifies multi-granularity cross-modal alignment as solid self-supervision for hashing learning. We present HUGGINGHASH as an instantiation and show its advantages on three benchmark datasets. We also show that *hugging* is compatible and beneficial with existing UCMH methods when choosing transformers as the backbones. Our work provides a research basis for the promising direction of transformer-based UCMH. Future work includes integrating multi-granularity representations to generate high-quality hash codes.

Acknowledgement

This work is supported by the National Natural Science Foundation of China under grant 62171248, the PCNL KEY project (PCL2021A07), the Guangdong Basic and Applied Basic Research Foundation under grant 2021A1515110066, and the GXWD 20220811172936001.

References

[1] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.

[2] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

[3] Yue Cao, Bin Liu, Mingsheng Long, and Jianmin Wang. Cross-modal hamming hashing. In *Proceedings of the European conference on computer vision*, pages 202–218, 2018.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[6] Yongbiao Chen, Sheng Zhang, Fangxin Liu, Zhigang Chang, Mang Ye, and Zhengwei Qi. Transhash: Transformer-based hamming hashing for efficient image retrieval. In *Proceedings of the 2022 International Conference on Multimedia Retrieval*, pages 127–136, 2022.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, 2019.

[8] Guiguang Ding, Yuchen Guo, Jile Zhou, and Yue Gao. Large-scale cross-modality search via collective matrix factorization hashing. *IEEE Transactions on Image Processing*, 25(11):5427–5440, 2016.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

[10] Shiv Ram Dubey, Satish Kumar Singh, and Wei-Ta Chu. Vision transformer hashing for image retrieval. In *2022 IEEE International Conference on Multimedia and Expo*, 2022.

[11] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *European Conference on Computer Vision (ECCV)*, 2020.

[12] Simon Ging, Mohammadreza Zolfaghari, Hamed Pirsiavash, and Thomas Brox. Coot: Cooperative hierarchical transformer for video-text representation learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 22605–22618, 2020.

[13] Qinkang Gong, Liangdao Wang, Hanjiang Lai, Yan Pan, and Jian Yin. Vit2hash: Unsupervised information-preserving hashing. *arXiv preprint arXiv:2201.05541*, 2022.

[14] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[15] Xiangteng He, Yulin Pan, Mingqian Tang, and Yiliang Lv. Self-supervised video retrieval transformer network. *arXiv preprint arXiv:2104.07993*, 2021.

[16] Tuan Hoang, Thanh-Toan Do, Tam V Nguyen, and Ngai-Man Cheung. Unsupervised deep cross-modality spectral hashing. *IEEE Transactions on Image Processing*, 29: 8391–8406, 2020.

[17] Di Hu, Feiping Nie, and Xuelong Li. Deep binary reconstruction for cross-modal hashing. *IEEE Transactions on Multimedia*, 21(4):973–985, 2018.

[18] Hengtong Hu, Lingxi Xie, Richang Hong, and Qi Tian. Creating something from nothing: Unsupervised knowledge distillation for cross-modal hashing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3123–3132, 2020.

[19] Mengqiu Hu, Yang Yang, Fumin Shen, Ning Xie, Richang Hong, and Heng Tao Shen. Collective reconstructive embeddings for cross-modal hashing. *IEEE Transactions on Image Processing*, 28(6):2770–2784, 2018.

[20] Mark J Huiskes and Michael S Lew. The mir flickr retrieval evaluation. In *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, pages 39–43, 2008.

[21] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[22] Go Irie, Hiroyuki Arai, and Yukinobu Taniguchi. Alternating co-quantization for cross-modal hashing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1886–1894, 2015.

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012.

[24] Shaishav Kumar and Raghavendra Udupa. Learning hash functions for cross-view similarity search. In *Twenty-second international joint conference on artificial intelligence*, 2011.

[25] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196. PMLR, 2014.

[26] Chao Li, Cheng Deng, Lei Wang, De Xie, and Xianglong Liu. Coupled cyclegan: Unsupervised hashing network for cross-modal retrieval. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 176–183, 2019.

[27] Mingyong Li and Hongya Wang. Unsupervised deep cross-modal hashing by knowledge distillation for large-scale cross-modal retrieval. In *Proceedings of the 2021 International Conference on Multimedia Retrieval*, pages 183–191, 2021.

[28] Shuyan Li, Xiu Li, Jiwen Lu, and Jie Zhou. Self-supervised video hashing via bidirectional transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13549–13558, 2021.

[29] Tao Li, Zheng Zhang, Lishen Pei, and Yan Gan. Hashformer: Vision transformer based deep hashing for image retrieval. *IEEE Signal Processing Letters*, 2022.

[30] Xuelong Li, Di Hu, and Feiping Nie. Deep binary reconstruction for cross-modal hashing. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1398–1406, 2017.

[31] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[32] Hong Liu, Rongrong Ji, Yongjian Wu, Feiyue Huang, and Baochang Zhang. Cross-modality binary code learning via fusion similarity hashing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7380–7388, 2017.

[33] Song Liu, Shengsheng Qian, Yang Guan, Jiawei Zhan, and Long Ying. Joint-modal distribution-based similarity hashing for large-scale unsupervised deep cross-modal retrieval. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pages 1379–1388, 2020.

[34] Song Liu, Haoqi Fan, Shengsheng Qian, Yiru Chen, Wenkui Ding, and Zhongyuan Wang. Hit: Hierarchical transformer with momentum contrast for video-text retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11915–11925, October 2021.

[35] Tie-Yan Liu et al. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331, 2009.

[36] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[37] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[38] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[39] Di Lu, Jinpeng Wang, Ziyun Zeng, Bin Chen, Shudeng Wu, and Shu-Tao Xia. Swinfghash: Fine-grained image retrieval via transformer-based hashing network. In *32nd British Machine Vision Conference*, 2021.

[40] Nicola Messina, Giuseppe Amato, Andrea Esuli, Fabrizio Falchi, Claudio Gennaro, and Stéphane Marchand-Maillet. Fine-grained visual textual alignment for cross-modal retrieval using transformer encoders. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 17(4):1–23, 2021.

[41] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[42] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32:8026–8037, 2019.

[43] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[44] Nikhil Rasiwasia, Jose Costa Pereira, Emanuele Coviello, Gabriel Doyle, Gert RG Lanckriet, Roger Levy, and Nuno Vasconcelos. A new approach to cross-modal multimedia retrieval. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 251–260, 2010.

[45] Amaia Salvador, Erhan Gundogdu, Loris Bazzani, and Michael Donoser. Revamping cross-modal recipe retrieval with hierarchical transformers and self-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15475–15484, 2021.

[46] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[47] Andrew Shin, Masato Ishii, and Takuya Narihira. Perspectives and prospects on transformer architecture for cross-modal tasks with language and vision. *International Journal of Computer Vision*, pages 1–20, 2022.

[48] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[49] Jingkuan Song, Yang Yang, Yi Yang, Zi Huang, and Heng Tao Shen. Inter-media hashing for large-scale retrieval from heterogeneous data sources. In *Proceedings of the 2013 ACM SIGMOD international conference on management of data*, pages 785–796, 2013.

[50] Shupeng Su, Zhisheng Zhong, and Chao Zhang. Deep joint-semantics reconstructing hashing for large-scale unsupervised cross-modal retrieval. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3027–3035, 2019.

[51] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021.

[52] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[53] Jinpeng Wang, Bin Chen, Qiang Zhang, Zaiqiao Meng, Shangsong Liang, and Shutao Xia. Weakly supervised deep hyperspherical quantization for image retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2755–2763, 2021.

[54] Jinpeng Wang, Bin Chen, Dongliang Liao, Ziyun Zeng, Gongfu Li, Xia Shu-Tao, and Jin Xu. Hybrid contrastive quantization for efficient cross-view video retrieval. In *Proceedings of the Web Conference 2022*, 2022.

[55] Jinpeng Wang, Ziyun Zeng, Bin Chen, Tao Dai, and Shu-Tao Xia. Contrastive quantization with code memory for unsupervised image retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2468–2476, 2022.

[56] Weiwei Wang, Yuming Shen, Haofeng Zhang, Yazhou Yao, and Li Liu. Set and rebase: determining the semantic graph connectivity for unsupervised cross-modal hashing. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 853–859, 2020.

[57] Xiaohan Wang, Linchao Zhu, and Yi Yang. T2vlad: global-local sequence alignment for text-video retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5079–5088, 2021.

[58] Keyu Wen, Jin Xia, Yuanyuan Huang, Linyang Li, Jiayan Xu, and Jie Shao. Cookie: Contrastive cross-modal knowledge sharing pre-training for vision-language representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2208–2217, 2021.

[59] Gengshen Wu, Zijia Lin, Jungong Han, Li Liu, Guiguang Ding, Baochang Zhang, and Jialie Shen. Unsupervised deep hashing via binary latent factor models for large-scale cross-modal retrieval. In *IJCAI*, volume 1, page 5, 2018.

[60] Dejie Yang, Dayan Wu, Wanqian Zhang, Haisu Zhang, Bo Li, and Weiping Wang. Deep semantic-alignment hashing for unsupervised cross-modal retrieval. In *Proceedings of the 2020 international conference on multimedia retrieval*, pages 44–52, 2020.

[61] Jianwei Yang, Yonatan Bisk, and Jianfeng Gao. Taco: Token-aware cascade contrastive learning for video-text alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11562–11572, 2021.

[62] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.

[63] Jun Yu, Hao Zhou, Yibing Zhan, and Dacheng Tao. Deep graph-neighbor coherence preserving network for unsupervised cross-modal hashing. In *Proceedings of the AAAI Conference on Artificial Intelligence. AAAI*, pages 4626–4634, 2021.

[64] Ziyun Zeng, Jinpeng Wang, Bin Chen, Tao Dai, and Shu-Tao Xia. Pyramid hybrid pooling quantization for efficient fine-grained image retrieval. *arXiv preprint arXiv:2109.05206*, 2021.

[65] Ziyun Zeng, Jinpeng Wang, Bin Chen, Yuting Wang, and Shu-Tao Xia. Motion-aware graph reasoning hashing for self-supervised video retrieval. In *33nd British Machine Vision Conference*, 2022.

[66] Jian Zhang and Yuxin Peng. Multi-pathway generative adversarial hashing for unsupervised cross-modal retrieval. *IEEE Transactions on Multimedia*, 22(1):174–187, 2019.

[67] Jian Zhang, Yuxin Peng, and Mingkuan Yuan. Unsupervised generative adversarial cross-modal hashing. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[68] Peng-Fei Zhang, Yang Li, Zi Huang, and Xin-Shun Xu. Aggregation-based graph convolutional hashing for unsupervised cross-modal retrieval. *IEEE Transactions on Multimedia*, 24:466–479, 2021.

[69] Yujie Zhong, Relja Arandjelović, and Andrew Zisserman. Ghostvlad for set-based face recognition. In *Asian conference on computer vision*, pages 35–50. Springer, 2018.

[70] Jile Zhou, Guiguang Ding, and Yuchen Guo. Latent semantic sparse hashing for cross-modal similarity search. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 415–424, 2014.

[71] Xiaofeng Zhu, Zi Huang, Heng Tao Shen, and Xin Zhao. Linear cross-modal hashing for efficient multimedia search. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 143–152, 2013.

[72] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27, 2015.