Article

A Dual-Module System for Copyright-Free Image Recommendation and Infringement Detection in Educational Materials

Yeongha Kim , Soyeon Kim * , Seonghyun Min , Youngung Han , Ohyoung Lee , Wongyum Kim *

Posted Date: 30 September 2024

doi: 10.20944/preprints202409.2285.v1

Keywords: Copyright protection; Copyright-free; Image retrieval; Infringement detection

Article

A Dual-Module System for Copyright-Free Image Recommendation and Infringement Detection in Educational Materials

Yeongha Kim [1,†], Soyeon Kim [2,*,†], Seonghyun Min [3,†], Youngung Han [4,†], Ohyoung Lee [5,†] and Wongyum Kim [6,*,†]

[1] R&D Center, AIDEEP Co., Ltd., 25, 7na-gil Gwacheon-daero, Gwacheon-si, 13840, Gyeonggi-do, R.O.Korea; yhkim@aideep.ai
[2] sykim@aideep.ai
[3] shmin@aideep.ai
[4] yuhan@aideep.ai
[5] Tekville, 551, Eonju-ro, Gangnam-gu, 06138, Seoul, R.O.Korea.; 250.lee@tekville.com
[6] wgkim@aideep.ai
[*] Correspondence: sykim@aideep.ai (S.K.); wgkim@aideep.ai (W.K.)
[†] These authors contributed equally to this work

Abstract: Images are widely used in educational materials due to their ability to effectively convey complex concepts. However, unauthorized image use often leads to legal issues related to copyright infringement. To address this issue, we introduce a dual-module system designed specifically for educators. The first module, a copyright infringement detection system, leverages deep learning techniques to verify the copyright status of images. It employs a Convolutional Variational Autoencoder (CVAE) model to extract meaningful features from copyrighted images and compares them against user-provided images. If infringement is suspected, the second module, an image retrieval system, recommends alternative copyright-free images using a Vision Transformer (ViT)-based hashing model. Evaluation on benchmark datasets demonstrates the system's effectiveness, achieving a mean Average Precision (mAP) of 0.812 on the Flickr25k dataset. Furthermore, a user study with 65 teachers indicates high satisfaction levels, particularly in addressing copyright concerns and ease of use. Our system significantly aids educators in creating educational materials compliant with copyright regulations.

Keywords: copyright protection; copyright-free; image retrieval; infringement detection;

1. Introduction

In today's internet environment, images serve as a core means of communication, information dissemination, and expression. Their ability to convey complex information in a concise and clear manner makes them frequently utilized not only in simple online contexts but also as educational materials for teaching purposes. However, using such images without the permission of the copyright holder or using images with unverified copyright status can lead to legal issues related to copyright infringement.

In South Korea, according to copyright law, educational institutions are supported in publishing copyrighted works in textbooks necessary for educational purposes. Additionally, according to Article 25, Paragraph 7 of the Copyright Act, the Korea Literary and Artistic Copyright Association (KOLAA), a body designated by the Minister of Culture, Sports, and Tourism of Korea, receives an annual "Statement of Use of Copyrighted Works" and royalties for the use of copyrighted materials from institutions that use works for instructional purposes. KOLAA then distributes the royalties (compensation) to the respective copyright holders, thereby protecting the rights of the copyright owners and ensuring they receive fair compensation. Consequently, teachers can freely use published copyrighted images in their teaching materials without worrying about copyright issues, which greatly aids in improving students' understanding, maximizing learning outcomes, and enhancing the quality of lessons.

However, the internet contains countless images beyond those registered with KOLAA, many of which have unverified copyright status. Therefore, educators who create instructional materials for educational purposes must carefully consider potential legal issues related to copyright, especially if the unverified images are distributed beyond the school.

This paper proposes a copyright-free image recommendation system designed to alleviate these concerns. The system helps educators determine whether the images they choose for educational materials may result in copyright infringement. Furthermore, it retrieval similar images from a database of verified copyright-free images when there is a risk of copyright issues with the selected images.

The system is composed of two modules. The copyright infringement detection module determines whether the image provided by the user is a published copyrighted work registered with KOLAA. Using a deep learning-based model, the Convolutional Variational Auto-Encoder (CVAE) [1], meaningful feature vectors are extracted from the copyrighted images, which are then converted into two types of keys (1st, 2nd Key) for faster searching within the database and stored in the database. Subsequently, the image provided by the user is compared with the feature information in the database to determine whether it is a verified copyrighted image that exists in the copyright database.

If the inspected image is likely to result in copyright infringement, the image retrieval module suggests similar images from the copyright-free image database using an image hashing model. By extracting a hash code from the user-provided image and comparing it with vectors in the database, the system effectively recommends similar images. Through this process, users can easily find and actively utilize copyright-free images from the copyright database for educational material creation.

The main contributions of the proposed system are as follows:

- The system integrates copyright infringement detection and similar image retrieval functionalities to help educators create educational materials without infringing on copyrights. This approach prevents copyright issues in advance and enhances the efficiency and safety of educational material creation by recommending copyright-verified images.
- The network structure of the image retrieval model was improved by using the Vision Transformer (ViT) [2] as the backbone network, enhancing the system's ability to capture global feature information from images. Additionally, a new loss function was introduced to improve the model's training efficiency and the quality of retrieved images.
- The system's utility was verified through user satisfaction surveys and performance evaluations of the model. The image hashing model outperformed other hashing models on the Flickr25k [3], CIFAR10 [4] and NUS-WIDE [5] benchmark datasets. Surveys conducted among educators showed high satisfaction with the system and alleviated concerns about copyright infringement.

The structure of this paper is as follows: In Section 2, related work, we review studies related to copyright systems and image retrieval. Section 3, Copyright-free Image Recommendation System, provides an overview of the system, describing the infringement detection module and the image retrieval module. Section 4 analyzes the experimental results of the image retrieval model and the service satisfaction results. In Section 5, discussion, the research findings are discussed in-depth, and the final conclusion summarizes the main findings and implications of the study.

2. Related Work

The protection of digital image copyrights has become an increasingly important issue with the advancement of the information society. Particularly, with the proliferation of the internet and social media, there has been a significant increase in the indiscriminate distribution of images and cases of copyright infringement, leading to the proposal of various technical approaches to address this issue. Traditional methods such as digital watermarking, signatures, and hash functions have steadily evolved since their early research stages and have established themselves as fundamental technologies for copyright protection to this day. [6] proposed a system that uses watermarking technology to protect the copyrights of digital images. This system embeds a watermark containing copyright information into the image and detects it to verify whether copyright infringement has occurred. Subsequently, research was conducted to strengthen copyright protection by utilizing blockchain technology.

Khare et al. [7] proposed a copyright infringement detection system that combines artificial intelligence with blockchain. This system extracts image features and compares them with copyright information stored in the blockchain to determine whether infringement has occurred. Recent studies have suggested methods to enhance copyright protection using the latest technologies such as artificial intelligence and deep learning. Sun and Zhou [8] proposed a system that compares image similarity using deep perceptual hashing based on hash-centered techniques to detect copyright infringement. Additionally, Kim et al. [9] proposed a framework to accurately handle the manipulation of copyrighted photos. This framework detects the Region of Interest (RoI) in the image, generates binary descriptors from the detected RoI, and compares them with a database to search for similar images.

Previous studies have primarily focused on preventing the replication and infringement of copyrighted images. In contrast, our system aims to verify the copyright status of an image to prevent copyright infringement issues and retrieval copyright-free images that pose no legal concerns. This distinction sets our system apart from existing research, as it emphasizes preemptively addressing copyright issues and providing safe images for educational material creation. To achieve this, we have implemented a deep learning model to effectively perform image infringement detection and retrieval.

One of the recent studies on image copyright verification technology is the autoencoder-based copyright image authentication model proposed by Yang et al. [10]. The CVAE used in this model effectively extracts essential spatial features of an image by utilizing convolutional filters and enables efficient copyright image authentication by generating and reconstructing latent vectors of the input image using a variational autoencoder structure. It handles various image formats and resolutions, converting high-dimensional data into a lower-dimensional latent space, thus maintaining critical features while accurately assessing image similarity. Zajic et al. [11] proposed an algorithm for content-based image retrieval (CBIR). This methodology involves extracting and processing image features based on color, texture, and shape. In the initial search phase, similar images to the query image are selected based on Euclidean distance. This process is refined iteratively using a Radial Basis Function (RBF) type neural network to improve the search results, allowing the user to refine and filter the results. TBH [12] introduced a dual bottleneck structure to address the chronic issue of information loss in hashing models. The model adds a binary-bottleneck and a continuous-bottleneck to the basic autoencoder and utilizes Graph Convolution [13] to learn the correlations between images within a batch. TBH avoids information loss by combining continuous features with the input feature vectors based on the similarity matrix calculated from the binary-bottleneck.

3. Copyright-Free Image Recommendation System

The system is designed to assist educators in creating materials that comply with copyright regulations. In the following sections, we provide a comprehensive overview of the system architecture and explain the functionality of each module in detail.

3.1. System Overview

Figure 1 presents an overview of the copyright-free image recommendation system which consists of two main components: an image infringement detection module and an image retrieval module.

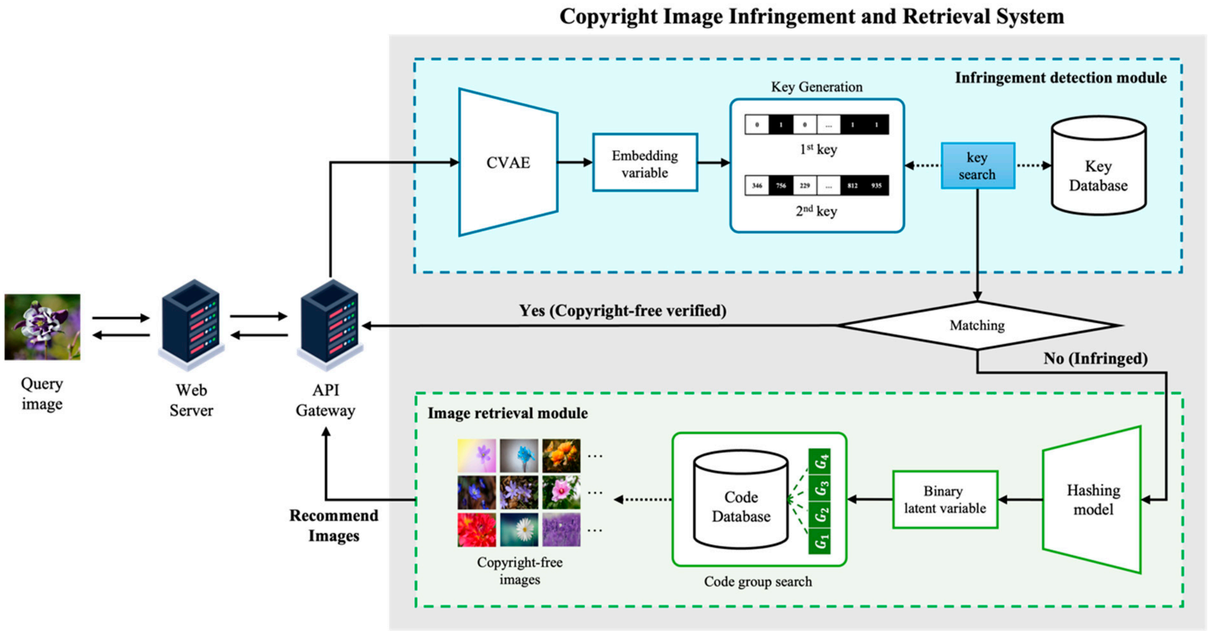


Figure 1. System architecture of the Copyright-free image recommendation system.

The image infringement detection module processes the query image uploaded by the user to determine whether it matches any copyrighted images in the database. This module employs a model based on the previously developed CVAE. The uploaded image is passed through the encoder of the pre-trained CVAE to extract its feature information. In a two-step process, unique feature vectors referred to as the first and second keys are generated. These keys act as distinctive identifiers that represent the image's characteristics.

The search is conducted in two stages by comparing these keys with those stored in the copyright database. In the first stage, the system calculates the Hamming distance between the first key of the query image and the key of the images in the database, selecting results that fall below the threshold $\mathcal{T}_{1st}$. If a single result is found, it is treated as the final result. If multiple results or no results are found, the system proceeds to the second stage where it compares the second key using cosine similarity. The result with the highest similarity exceeding the threshold $\mathcal{T}_{2nd}$ is selected as the final match. If the final result is determined, the system informs the user that the image is confirmed to be copyrighted. If the thresholds are not met, the system redirects the user to the image retrieval module which suggests similar images from a copyright-free database.

The image retrieval module uses an improved model of the Twin-Bottleneck Hashing (TBH), one of the Deep Hashing models, to extract feature information from the query image provided by the user and convert it into a binary vector. The search is then performed in the copyright-free image database by calculating the Hamming distance in a manner similar to the first step of the infringement detection process. The results are ranked in ascending order based on the Hamming distance with the top N images returned to the user. The proposed system is designed based on a web server and API server, allowing users to access and use its functionalities through a web interface.

3.2. Infringement Detection Module

The copyright image infringement detection module described in Figure 2 utilizes a deep learning model based on the CVAE to extract features from images and determine whether they are copyrighted by comparing these features with the information stored in the database. The CVAE predicts a probability distribution that accurately represents the given input image through the encoder, while the decoder generates a new image similar to the input image using the estimated probability distribution. In this paper, we adopt the CVAE model proposed by Yang et al. to take advantage of its ability to capture and compress key information from the input image and integrate it into the copyright infringement detection module.

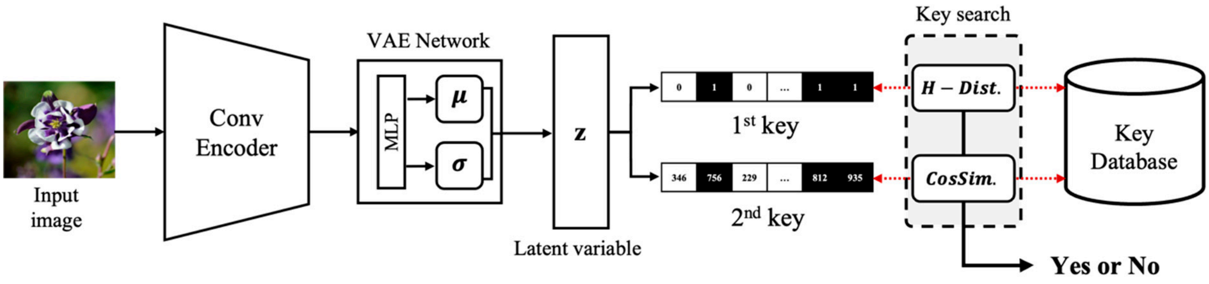


Figure 2. Copyright infringement detection process using CVAE.

When a query image is provided by the user, the encoder extracts local features through multiple convolutional filters, identifying key characteristics of the image and compressing them into a lower dimension. This compressed feature information is then used to predict the probability distributions μ and σ, from which an embedding vector z is sampled, effectively capturing the unique attributes of the image. The first and second keys generated from the embedding vector z are then compared with the key database to produce the final detection result. The method for generating and matching these keys is explained in detail below.

3.2.1. 1st Key Generation

The first key $K^{1st} \in \mathbb{R}^M$ is generated from the query image through the following process. First, the mean $A^{1st} \in \mathbb{R}^M$ of the embedding vectors $z \in \mathbb{R}^{N,M}$ for all the copyrighted images used during model training is pre-calculated. Here, N represents the total number of images, and M is the size of the embedding vector z, which also corresponds to the common key length for both the 1st key and 2nd key. Then, when a query image is provided by the user, the embedding vector is obtained through the CVAE, and the first key K^{1st} is generated by comparing it with the mean A^{1st} of the embedding vectors of the copyrighted images.

$$K^{1st} = \begin{cases} if\ z_i \geq \mathcal{A}_i^{1st},\ K_i^{1st} = 1 \\ if\ z_i < \mathcal{A}_i^{1st},\ K_i^{1st} = 0 \end{cases}. \tag{1}$$

3.2.2. 2nd Key Generation

The second key $K^{2nd} \in \mathbb{R}^M$ is generated from the embedding vector z by rounding each element of z to the fourth decimal place and then multiplying by 1000 to convert it into an integer, as shown in Eq. 2:

$$K^{2nd} = round(z, 4) \times 1000. \tag{2}$$

This process narrows down the values of the embedding vector to a certain number of digits, thereby reducing the data size and computational complexity during subsequent comparison and retrieval processes.

The reason for binarizing and integerizing K^{1st} and K^{2nd} during key generation is to improve system efficiency. If the feature vector composed of real values is used directly to perform searches within the database, it would require significant computational resources and time due to the need to find similar values through matrix multiplication of real vectors. Therefore, by adopting the above key generation method, more rapid and efficient searches can be achieved. Binarized and integerized keys simplify comparison operations, increasing processing speed and enhancing system performance.

3.2.3. Key Search

The generated K^{1st} and K^{2nd} undergo a comparison process with the copyrighted images in the DB to verify the copyright status of the query image. The first key K_Q^{1st} of the query image is compared with K_{DB}^{1st} of the copyrighted images in the DB by calculating the Hamming distance, searching for images in the DB that have a Hamming distance less than or equal to $\mathcal{T}_{1st}$.

$$\{\mathcal{R} \in DB |\ \mathcal{H}(K_{Q}^{1st}, K_{DB}^{1st}) \leq \mathcal{T}_{1st}\} \in \mathcal{R}^{1st}. \tag{3}$$

In Eq. 3, $\mathcal{R}$ represents each key entry in the DB, $\mathcal{R}^{1st}$ denotes the complete search results from the first key, $\mathcal{H}$ is the function that calculates the Hamming distance, and $\mathcal{T}_{1st}$ represents the threshold for the first key. If multiple results are returned from the first search, a secondary search is conducted using the second key. The second key K^{2nd} of the query image is compared with K^{2nd} of the filtered images from the first search using cosine similarity. Images with a cosine similarity greater than the second key threshold $\mathcal{T}_{2nd}$ are filtered, and the image with the highest cosine similarity is selected as the final search result. This process is expressed in Eq. 4 and 5:

$$\{\mathcal{R} \in \mathcal{R}^{1st} \mid \mathcal{C}(K_{Q}^{2nd}, K_{DB}^{2nd}) \geq \mathcal{T}_{2nd}\} \in \mathcal{R}^{2nd}. \tag{4}$$

$$\mathcal{R}^{final} = \mathcal{R}^{2nd}[argmax(\mathcal{C}^{2nd})]. \tag{5}$$

In Eq. 4, K_{Q}^{2nd} and K_{DB}^{2nd} represent the second key values of the query image and the copyrighted images in the DB, respectively, and is the function that calculates cosine similarity. In Eq. 5, $\mathcal{C}^{2nd}$ refers to the cosine similarity values calculated during the second key search process. $\mathcal{R}^{final}$ represents the final search result with the highest cosine similarity.

In the proposed system, the first threshold $\mathcal{T}_{1st}$ is set to 12, and the second threshold $\mathcal{T}_{2nd}$ is set to 0.9 to limit the search range. If the first threshold is exceeded or the second threshold is not met, resulting in no final search result, the copyright status of the image cannot be verified. In such cases, the system redirects to the image retrieval module, which suggests similar copyright-free images available in the copyright retrieval DB.

3.3. Image Retrieval Module

This chapter explains specific technical approaches to enhance the performance of the image retrieval module. It describes the integration of the Vision Transformer (ViT) backbone network, an image model based on transformers [14], into the existing TBH model and the introduction of a loss function to improve performance through efficient model training. Following this, methods to maximize search efficiency in large-scale databases are introduced, including a Hamming distance-based search method and search optimization using binary vector code groups.

3.3.1. Deep Hashing Model

Figure 3 illustrates the structure of the copyright image retrieval module, which performs image retrieval tasks using a Deep Hashing-based model. By hashing the feature information of images into binary vectors and storing them in a database, similar images can be quickly located among a vast number of images. The TBH we adopted employs an auto-encoding structure with two bottlenecks: a binary bottleneck and a continuous latent variable. The binary bottleneck builds an adaptive similarity graph based on Hamming distance, while the continuous bottleneck adjusts the data through GCN before feeding it into the decoder. The decoder reconstructs the input data, and the reconstruction loss compensates for the encoding quality of the encoder. This model is fully trainable via SGD and overcomes the limitations of static graph problems, generating more discriminative binary codes.

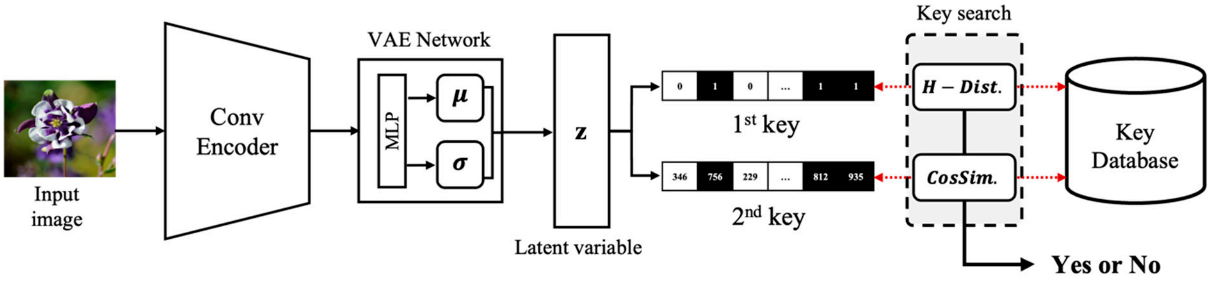


Figure 3. Image retrieval process.

In this study, we made the following improvements to enhance the model's representational capacity. First, we replaced the AlexNet model, previously used as the backbone of the TBH model, with the ViT model. While CNN-based backbones capture hierarchical features using small filters, they have limitations in considering the overall context of the image. In contrast, the ViT model divides an image into fixed-size patches, linearly embeds each patch, adds positional embeddings, and feeds them into a transformer encoder. The encoder, composed of multi-head self-attention and MLP blocks, models the relationships between patches and integrates information. Additionally, ViT exhibits less image-specific inductive bias than CNNs and shows generalized performance across various image types. These characteristics enable ViT to demonstrate superior performance in large-scale datasets and diverse image recognition tasks.

Next, we accelerated training by directly passing prior knowledge of inter-image relationships from the backbone to the bottleneck in the hashing model. The additional loss function was designed to minimize the mean squared error (MSE) between the adjacency matrices generated at the backbone network and bottleneck. The original TBH model consists of the sum of the autoencoder loss and the discriminator loss, calculated through Eq. 6 and Eq. 7.

$$\nabla L_{AE} \approx \frac{1}{B}\sum_{i=1}^{B} \mathbb{E}_{b_i}\left[\nabla(\frac{1}{2}\|x_i - \hat{x}_i\|^2 - \lambda \log D_1(b_i;\varphi_1) - \lambda \log D_2(z'_i;\varphi_2))\right], \tag{6}$$

$$\begin{aligned} L_D = \frac{\lambda}{B}\sum_{i=1}^{B}\Big(&\log d_1\left(y_i^{(b)};\varphi_1\right) + \log d_2\left(y_i^{(c)};\varphi_2\right) \\ &+ \log\left(1 - d_1(b_i;\varphi_1)\right) + \log\left(1 - d_2(z'_i;\varphi_2)\right)\Big). \end{aligned} \tag{7}$$

First, the autoencoder loss is a loss function designed to minimize the difference between the reconstructed features from the model and the original backbone features, while the discriminator loss serves to regularize the generated binary codes and continuous variables to resemble the target distribution. To update the autoencoder loss, the gradient is estimated using the Monte Carlo sampling method, which requires many iterations and results in prolonged training times.

To accelerate the convergence speed of the encoder's training, we propose a new similarity loss function $\mathcal{L}_{sim}$ that utilizes the backbone features. $\mathcal{L}_{sim}$ generates an adjacency matrix by considering the correlations between features extracted through the backbone from the input data within a batch, and then compares this with the binary bottleneck adjacency matrix of the model. By using the prior knowledge embedded in the backbone to create similarity labels for the adjacency matrix, which are then used in model training, we achieve faster convergence without significantly altering the existing model structure. Specifically, $\mathcal{L}_{sim}$ follows these steps: First, the input data within a batch is passed through the backbone to extract features, and based on the correlations between these features, the adjacency matrix A_B is computed.

$$A_B^{i,j} = 0.5\left(1 + \frac{x'_i \cdot x'_j}{\|x'_i\|_2 \cdot \|x'_j\|_2}\right),\ A_{bbn}^{i,j} = 0.5\left(1 + \frac{b'_i \cdot b'_j}{\|b'_i\|_2 \cdot \|b'_j\|_2}\right)\ for\ i,j = 1,2,\dots,B. \tag{8}$$

At this point, the adjacency matrix A_{bbn} between the binary codes in the binary bottleneck structure is also calculated in the same manner.

$$\mathcal{L}_{sim} = \frac{1}{B^2}\sum_{i=1}^{B}\sum_{J=1}^{B}\left(A_{bbn}^{i,j} - A_{B}^{i,j}\right)^2. \tag{9}$$

By making the adjacency matrix of the backbone features similar to the adjacency matrix of the binary bottleneck through $\mathcal{L}_{sim}$, the model inherits prior knowledge from the backbone, thereby improving the training speed and stability of the model. The total loss function of the proposed model, including the original TBH losses, is as follows.

$$\mathcal{L}_{TBH} = \mathcal{L}_{AE} + \mathcal{L}_{D} + \mathcal{L}_{sim}.. \tag{10}$$

The proposed system uses the improved model to pre-build the binary information of copyrighted images in the database. When a user inputs a query image, the system compares the binary feature vector of the query image with those stored in the database and recommends similar images. The comparison between vectors is based on Hamming distance, and the Top N images with the shortest Hamming distances are returned as the final retrieval results.

3.3.2. Binary Code Group-Based Search

Figure 4 illustrates the vector search method, which improves search processing speed by efficiently calculating Hamming distances within the database. The query image $\mathcal{I}_Q$ is converted into a binary feature vector $\mathcal{C}_Q$ through the deep learning model, and in this study, the length of the binary vector $\mathcal{C}_{bit}$ is set to 16 bits. The binary vector is then divided into four code groups, each of which is compared with the binary vectors in the database to maximize search efficiency. To search for images where the Hamming distance between binary codes is below the threshold $\mathcal{T}$, the number of matching groups $\mathcal{N}_M$is calculated through code group comparisons, and images that satisfy $\mathcal{N}_M \geq \max(\mathcal{N}_G - \mathcal{T}, 0)$ are filtered to enhance search speed. Finally, the Hamming distance is calculated only for the filtered images to generate the final list of similar images. Algorithm 1 shows the detailed operation process of the speed enhancement algorithm based on code groups.

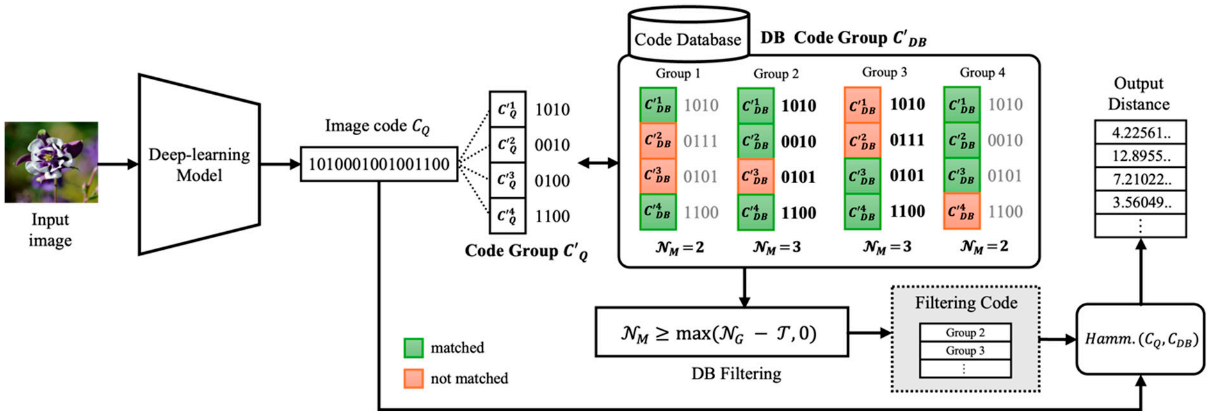


Figure 4. Code group-based image search process.

This approach was first introduced by [15] and allows efficient Hamming distance calculation in large-scale databases by using a simple method of reducing the candidate set through group matching before calculating the Hamming distance. This significantly improves the database search speed in the copyright image infringement detection and retrieval process.

Algorithm 1. Efficient hamming distance search algorithm from database pipeline.

```
Algorithm 1 Efficient hamming distance search from database pipeline
Input: Query image $\mathcal{I}_Q$
Output: Output Hamming distance list
Require: $\mathcal{N}_G$, $\mathcal{S}_G$, $\mathcal{T}$          // Code group number, size, threshold
```

```
  $\mathcal{T}' = \max(\mathcal{N}_G - \mathcal{T}, 0)$
// Get binary code using a deep learning model and slice binary code to make code group
  $\mathcal{C}_Q = Model(\mathcal{I}_Q)$
  $\mathcal{C}'^{1}_{Q}, \mathcal{C}'^{2}_{Q}, \ldots, \mathcal{C}'^{\mathcal{N}_G}_{Q} = Model(\mathcal{C}_Q, \mathcal{N}_G, \mathcal{S}_G)$
// $\mathcal{C}_Q$ = 1101001110101111 And $\mathcal{N}_G, \mathcal{S}_G$ = 4
// $\mathcal{C}'^{1}_{Q}$ = 1101, $\mathcal{C}'^{2}_{Q}$ = 0011, $\mathcal{C}'^{3}_{Q}$ = 1010, $\mathcal{C}'^{4}_{Q}$ = 1111
// Execute the query and save the query results in $\mathcal{R}$
SELECT * (
    CASE WHEN = THEN 1 ELSE 0 END +
    CASE WHEN = THEN 1 ELSE 0 END +
    ...
    CASE WHEN = THEN 1 ELSE 0 END
) AS
FROM DATABASE WHERE (
    $\mathcal{C}'^{1}_{DB} = \mathcal{C}'^{1}_{Q}$ OR
    $\mathcal{C}'^{2}_{DB} = \mathcal{C}'^{2}_{Q}$ OR
    ...
    $\mathcal{C}'^{\mathcal{N}_G}_{DB} = \mathcal{C}'^{\mathcal{N}_G}_{Q}$
) HAVING $\mathcal{N}_M \geq \mathcal{T}$
// Calculate the Hamming distance
for in do
    Calculate $HammingDist(\mathcal{C}_Q, \mathcal{C}_{DB})$
    And filtering $Hamming\ dist \leq \mathcal{T}$
end for
```

4. Experiments

4.1. Implementation Details

The proposed system was implemented using different deep learning frameworks for each module. The CVAE model in the infringement detection module was implemented with TensorFlow 2.8.0, and the TBH model for image retrieval was implemented with PyTorch 2.0.1. The entire system runs on an Ubuntu 20.04 environment equipped with an NVIDIA RTX A6000. For training the copyright infringement detection model, the batch size was set to 512, and a cosine scheduler was applied with a learning rate of 0.0001 for 1000 epochs. Each epoch took approximately 9 minutes and 26 seconds. The search thresholds for the infringement detection module were set to $\mathcal{T}_{1st}$ and $\mathcal{T}_{2nd}$, with values of 12 and 0.9, respectively. The image retrieval model utilized the TBH architecture, with a batch size of 128 set during the training process. The network was optimized using the Adam optimizer with a learning rate of 0.0001, and the learning rate was decreased using a cosine decay scheduler. The dimensionality of the feature size input to the model was set to 768, and training was conducted for up to 500 epochs, with early stopping applied to prevent overfitting.

4.2. Datasets

Flickr25k is a dataset consisting of 25,000 images, each with an average of 8.94 tags, and is one of the mainstream benchmarks used in image retrieval. For training the image retrieval model, we used 5,000 images for training, 2,000 images for testing, and the remaining images as the database dataset.

CIFAR10 consists of 60,000 images across 10 classes, with a relatively small resolution of 32x32. From the total of 60,000 images, we first split off 10,000 images as the test data for the image retrieval model and then evaluated the model using two different methods. In CIFAR10-I, the remaining 50,000 images, excluding the test data, were used for both training and the database. In CIFAR10-II, 5,000 images were allocated for training, and the remainder were used as the database.

NUS-WIDE is a dataset containing 269,648 images, with 81 concepts and 5,018 tags. We used 195,834 images belonging to 20 concepts for training the image retrieval model, and the remaining images, excluding 2,100 test data, were used for both training and the database.

KOLAA Copyright Image [16] is a private image dataset provided by the Korea Literary and Artistic Copyright Association, consisting of approximately 380,000 images with about 7 to 11 labels per image. We used 370,000 images for training and 1,000 images for testing the copyright infringement detection model. For the image retrieval model, 5,000 images were randomly selected for training, 2,000 images for testing, and 10,000 images as the database dataset.

4.3. Evaluation of Hashing Model

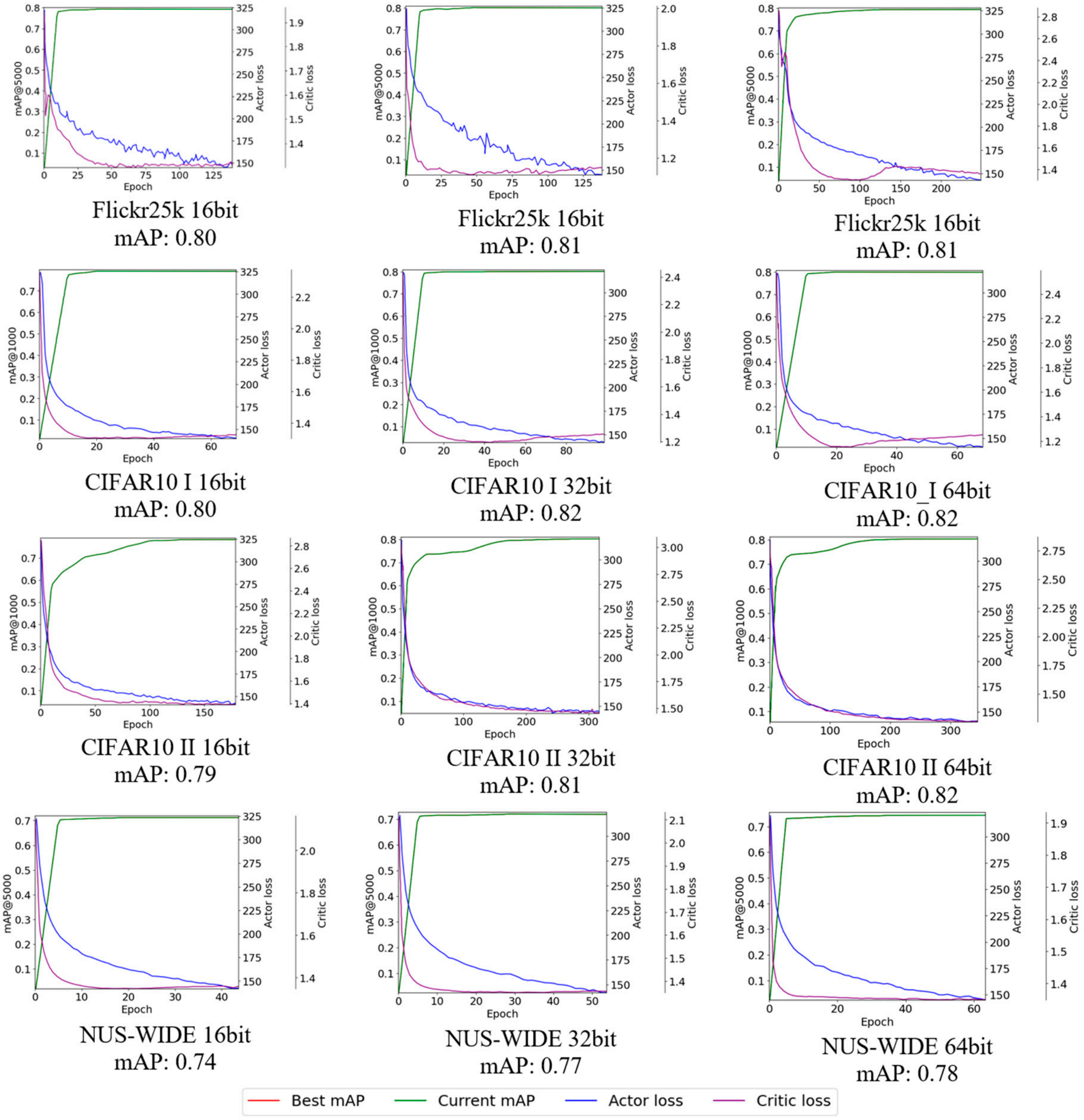
The performance of the proposed method was compared with that of existing hashing models on various benchmark sets (Flickr25k, CIFAR10, NUS-WIDE) and the KOLAA copyright image dataset. Performance evaluation was conducted using the mean Average Precision (mAP) metric. The mAP was calculated for the top k results with the highest similarity. The values of k were set to 5000 for Flickr25k and NUSWIDE, 1000 for CIFAR10, and 500 for KOLAA, with all datasets evaluated by generating hash codes of 16, 32, and 64-bit lengths. We compared the performance of Bi-halfNet [17], CIBHash [18], DSH [19], the existing TBH model, and the proposed model.

As shown in Table 1, the improved image retrieval model outperformed the existing TBH model across all datasets and achieved relatively higher results compared to other hashing models. On the CIFAR10-1 dataset, the proposed model demonstrated exceptional performance with a difference of up to 26%. Additionally, in the average model performance across all benchmarks, the proposed model achieved the highest mAP result of 78.9%, followed by CIBHash (70.8%), Bi-HalfNet (66.5%), TBH (65.8%), and DSH (64.1%). The experimental results demonstrate that the proposed model can generate unique hash codes that effectively represent images by leveraging the self-attention mechanism of the ViT backbone, which accurately identifies key features within images.

Table 1. Comparison with other hashing models.

Dataset	bits	Hashing Model				
		TBH [12]	**Bi-halfNet [17]**	**CIBHash [18]**	**DSH [19]**	**Ours**
Flickr25k [3] mAP@5000	16	0.743	0.760	0.772	0.677	**0.802**
	32	0.761	0.779	0.784	0.679	**0.809**
	64	0.778	0.786	0.795	0.712	**0.812**
CIFAR10 I [4] mAP@1000	16	0.546	0.561	0.593	0.651	**0.760**
	32	0.586	0.576	0.636	0.661	**0.774**
	64	0.624	0.595	0.651	0.676	**0.789**
CIFAR10 II [4] mAP@1000	16	0.532	0.499	0.590	0.640	**0.759**
	32	0.573	0.520	0.622	0.652	**0.766**
	64	0.578	0.553	0.641	0.669	**0.794**
NUS-WIDE [5] mAP@5000	16	0.717	0.768	0.790	0.552	**0.794**
	32	0.725	0.783	0.807	**0.558**	0.804
	64	0.735	0.799	0.815	**0.562**	0.810
KOLAA Copyright [16] mAP@500	16	0.544	0.614	0.633	0.598	**0.650**
	32	0.619	0.644	0.657	0.601	**0.711**
	64	0.629	0.667	0.673	0.623	**0.715**

Figure 5 illustrates the training process of the improved image retrieval model, showing the learning trends for 16, 32, and 64 bits on the Flickr25k, CIFAR10, and NUS-WIDE datasets. The best mAP, current mAP, actor loss, and critic loss are represented by red, green, blue, and purple lines, respectively. The graph shows that the current mAP curve closely aligns with the best mAP curve, indicating that the model is being trained appropriately. Notably, all graphs exhibit a sharp increase in performance during the early stages of training, suggesting that the model is effectively optimizing learning and accelerating convergence by leveraging diverse feature information. However, compared to the CIFAR10 learning trend, the loss curves for Flickr25k and NUS-WIDE show relatively slow and unstable declines. This could indicate that the model requires more time to learn diverse features or that the complexity of these datasets is relatively higher.



Flickr25k 16bit
mAP: 0.80

Flickr25k 16bit
mAP: 0.81

Flickr25k 16bit
mAP: 0.81

CIFAR10 I 16bit
mAP: 0.80

CIFAR10 I 32bit
mAP: 0.82

CIFAR10_I 64bit
mAP: 0.82

CIFAR10 II 16bit
mAP: 0.79

CIFAR10 II 32bit
mAP: 0.81

CIFAR10 II 64bit
mAP: 0.82

NUS-WIDE 16bit
mAP: 0.74

NUS-WIDE 32bit
mAP: 0.77

NUS-WIDE 64bit
mAP: 0.78

Figure 5. Training curves and mAP scores for different datasets.

Figure 6 presents the experimental retrieval results of the image retrieval module, visualized using T-SNE. The dataset used in the experiment is the two CIFAR10 datasets mentioned in Section 4.2, representing the distribution of 10 classes. In the 16-bit results, the clusters appear somewhat regionally localized, but as the bit length increases to 32 and 64, the distances between clusters decrease, leading to more cohesive groupings. Larger bit lengths reduce the likelihood of overlapping Hamming distances due to the longer vector length, thus improving accuracy. However, there is a trade-off with increased memory usage and computational costs. Additionally, when comparing the two datasets, CIFAR10 II, which was trained with fewer training data and with no overlap between the training and database sets, shows more compact clustering of each class compared to CIFAR10 I. This can be interpreted as an indication that the model improves performance by effectively handling diverse data without overfitting to specific data. Particularly, the good performance on untrained data suggests the potential for zero-shot learning. Additionally, the Appendix provides practical examples of the proposed approach through Figures A1–A5.

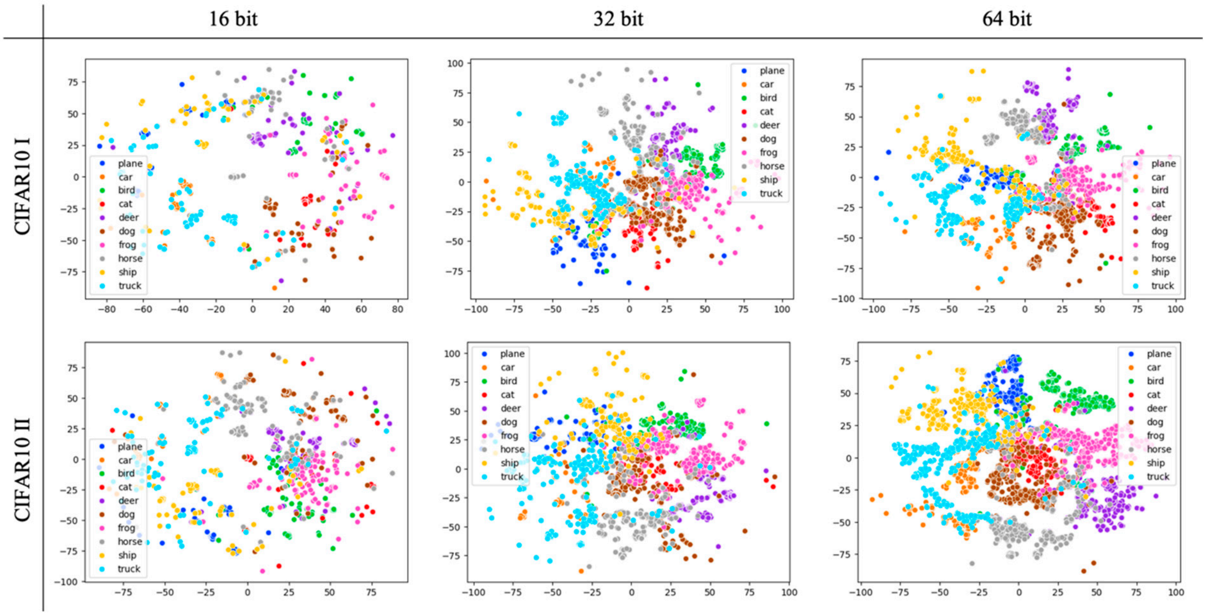


Figure 6. T-SNE visualization results for CIFAR10 I and CIFAR10 II.

4.4. User Satisfaction Analysis

To validate the effectiveness and practicality of the proposed copyright-free image recommendation system, a user satisfaction analysis was conducted. The focus was particularly on verifying whether teachers who create educational content could effectively prevent copyright infringement and receive efficient image retrieval. Among the 65 participating teachers, 47% were elementary school teachers, 31\% were middle school teachers, and 22% were high school teachers. Out of the 380,000 copyrighted images owned by KOLAA, 180,000 images were used to build the infringement detection DB and retrieval DB through the system's modules.

Teachers used the system to search for images related to their teaching materials and to verify copyright inspection results. During this process, a total of 1,757 retrieval results were generated, and reviews were written for each retrieval. Teachers evaluated their overall experience and satisfaction with the system using 13 evaluation items. The main evaluation metrics included Validity, Convenience, Accuracy, Speed, Completeness, retrieval Satisfaction, and Copyright Concern Relief, which are graphically represented in Figure 7.

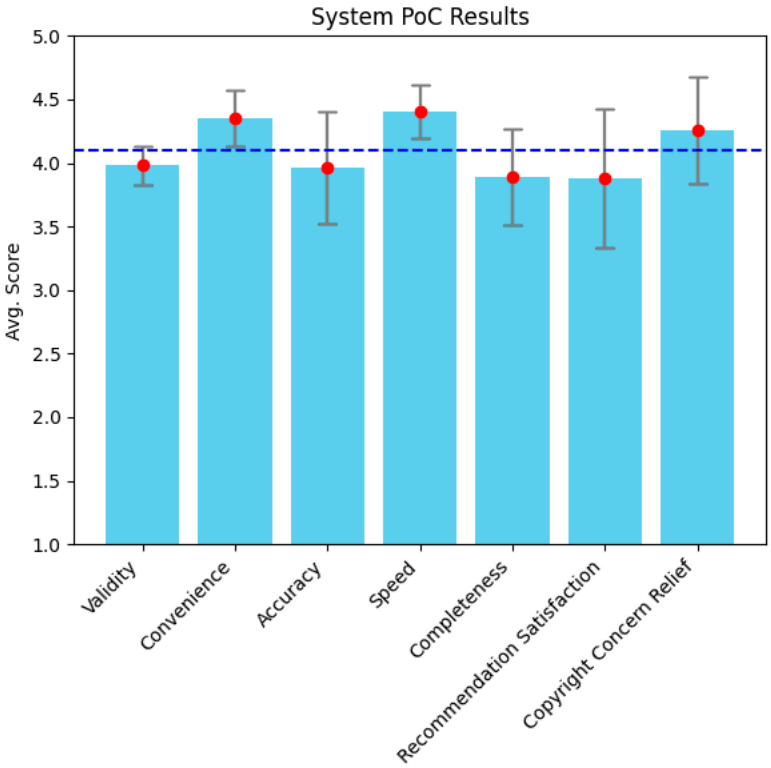


Figure 7. User satisfaction results of Copyright-free image recommendation system.

The analysis of the data collected using the Likert scale revealed that the overall satisfaction with the system was an average of 3.88 points (standard deviation of 0.55), indicating that users generally evaluated the system positively. The evaluation score for solving copyright image issues was particularly high, with an average of 4.26 points (standard deviation of 0.42). Notably, the Convenience (average of 4.35 points) and Speed (average of 4.40 points) metrics showed very high satisfaction, indicating that teachers experienced a high level of satisfaction when using the system.

5. Discussion

Considering the results of user satisfaction, the positive aspects of the system proposed in this study can be examined as follows. Teachers highly evaluated the system as a tool that effectively addresses copyright issues. In particular, the high score for alleviating concerns about copyright images indicates that the system greatly contributed to reducing legal issues in the process of creating educational materials. This allows teachers to use images with confidence when creating educational materials, enabling them to provide students with more diverse visual resources, thereby enhancing the quality of education. These results demonstrate that the proposed system exhibited significant performance in detecting copyright infringement and retrieving images. Additionally, the high evaluation of speed is attributed to the efficient design of the hash code within the system. The efficiently designed hash code improves data search and processing speed, optimizing system performance and enabling high evaluations of accuracy and speed. However, the large standard deviation in the accuracy and speed metrics suggests that further optimization is needed to provide consistent performance across diverse user environments and considering the flexibility and scalability of the hash code design, there is ample room for future improvement.

6. Conclusion

This study designed a copyright-free image recommendation system for educators, which checks copyright images and recommends similar images. The proposed system provides users with image infringement detection and image retrieval functions using two modules. Notably, the system's performance was improved over existing retrieval models by modifying the backbone network of the image retrieval model and introducing a new loss function, while also enhancing processing speed through group code matching. The performance evaluation of the proposed model and user satisfaction analysis of the overall system showed that the system can effectively solve copyright issues in educational settings and contribute to improving the quality of educational materials. In particular, as the potential of zero-shot learning was partially observed in image hashing model research, further studies are needed to examine its effectiveness in more diverse and practical datasets. Based on this, future research will move towards applying various deep learning models to further enhance system performance.

Author Contributions: Conceptualization, Yeongha Kim; methodology, Yeongha Kim; software, Yeongha Kim; validation, Yeongha Kim; investigation, Yeongha Kim, Soyeon Kim; writing—original draft preparation, Yeongha Kim, Soyeon Kim; writing—review and editing, Soyeon Kim, Seonghyun Min, Youngung Han, Wongyum Kim; visualization, Soyeon Kim; supervision, Wongyum Kim; project administration, Wongyum Kim, Seonghyun Min; resources, Ohyoung Lee; funding acquisition, Wongyum Kim. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by Ministry of Culture, Sports and Tourism and Korea Creative Content Agency (Project Number: 2021-EC-9500).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

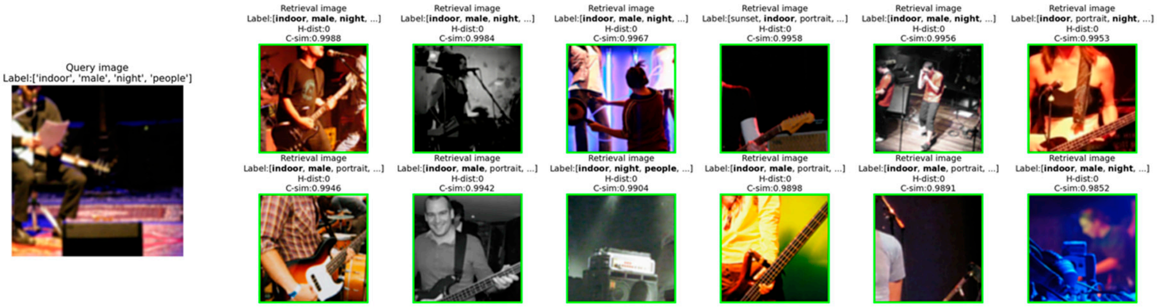
Conflicts of Interest: The authors declare no conflicts of interest.

Acknowledgments: This research is supported by Ministry of Culture, Sports and Tourism and Korea Creative Content Agency (Project Number: 2021-EC-9500).

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

This section provides additional visual results from the copyright-free image recommendation system discussed in the main paper. The leftmost large image is the input query image, while the smaller images on the right represent the search results. The similarity decreases from left to right in the first row, and from left to right in the second row. The first image of the second row follows the last image of the first row in terms of similarity ranking. Images that share the same tag as the query image are highlighted with a green border, while those that do not share the tag are outlined in red.


Query image
Label:['indoor', 'male', 'night', 'people']
Retrieval image
Label:[**indoor**, **male**, **night**, ...]
H-dist:0
C-sim:0.9988
Retrieval image
Label:[**indoor**, **male**, **night**, ...]
H-dist:0
C-sim:0.9984
Retrieval image
Label:[**indoor**, **male**, **night**, ...]
H-dist:0
C-sim:0.9967
Retrieval image
Label:[sunset, **indoor**, portrait, ...]
H-dist:0
C-sim:0.9958
Retrieval image
Label:[**indoor**, **male**, **night**, ...]
H-dist:0
C-sim:0.9956
Retrieval image
Label:[**indoor**, portrait, **night**, ...]
H-dist:0
C-sim:0.9953
Retrieval image
Label:[**indoor**, **male**, portrait, ...]
H-dist:0
C-sim:0.9946
Retrieval image
Label:[**indoor**, **male**, portrait, ...]
H-dist:0
C-sim:0.9942
Retrieval image
Label:[**indoor**, **night**, **people**, ...]
H-dist:0
C-sim:0.9904
Retrieval image
Label:[**indoor**, **male**, portrait, ...]
H-dist:0
C-sim:0.9898
Retrieval image
Label:[**indoor**, **male**, portrait, ...]
H-dist:0
C-sim:0.9891
Retrieval image
Label:[**indoor**, **male**, **night**, ...]
H-dist:0
C-sim:0.9852

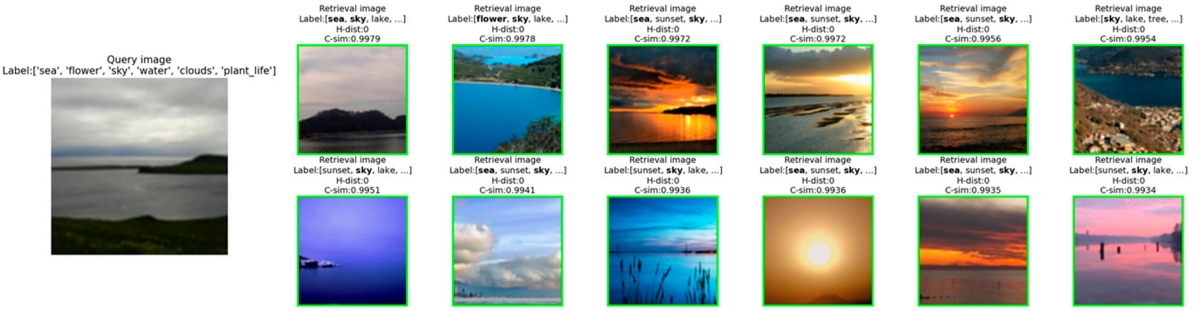

Query image
Label:['sea', 'flower', 'sky', 'water', 'clouds', 'plant_life']
Retrieval image
Label:[**sea**, **sky**, lake, ...]
H-dist:0
C-sim:0.9979
Retrieval image
Label:[**flower**, **sky**, lake, ...]
H-dist:0
C-sim:0.9978
Retrieval image
Label:[**sea**, sunset, **sky**, ...]
H-dist:0
C-sim:0.9972
Retrieval image
Label:[**sea**, sunset, **sky**, ...]
H-dist:0
C-sim:0.9972
Retrieval image
Label:[**sea**, sunset, **sky**, ...]
H-dist:0
C-sim:0.9956
Retrieval image
Label:[**sky**, lake, tree, ...]
H-dist:0
C-sim:0.9954
Retrieval image
Label:[sunset, **sky**, lake, ...]
H-dist:0
C-sim:0.9951
Retrieval image
Label:[**sea**, sunset, **sky**, ...]
H-dist:0
C-sim:0.9941
Retrieval image
Label:[sunset, **sky**, lake, ...]
H-dist:0
C-sim:0.9936
Retrieval image
Label:[**sea**, sunset, **sky**, ...]
H-dist:0
C-sim:0.9936
Retrieval image
Label:[**sea**, sunset, **sky**, ...]
H-dist:0
C-sim:0.9935
Retrieval image
Label:[sunset, **sky**, lake, ...]
H-dist:0
C-sim:0.9934

Figure A1. Detailed image retrieval results for Flickr25k.

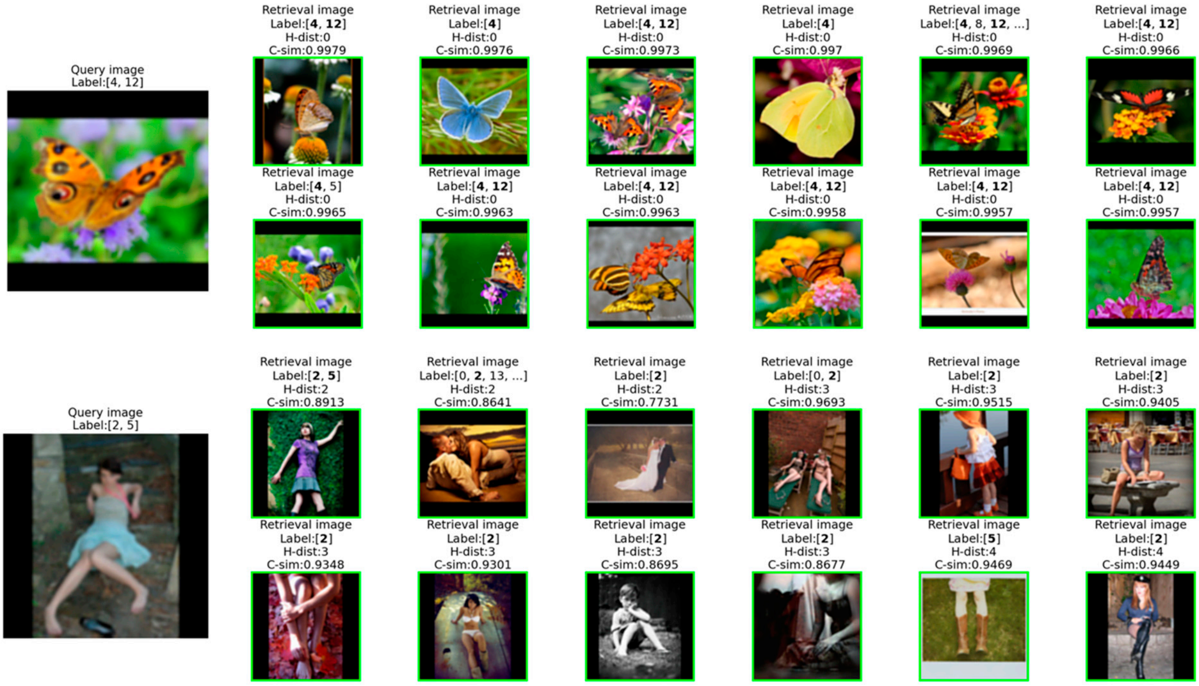


Figure A2. Detailed image retrieval results for NUS-WIDE.

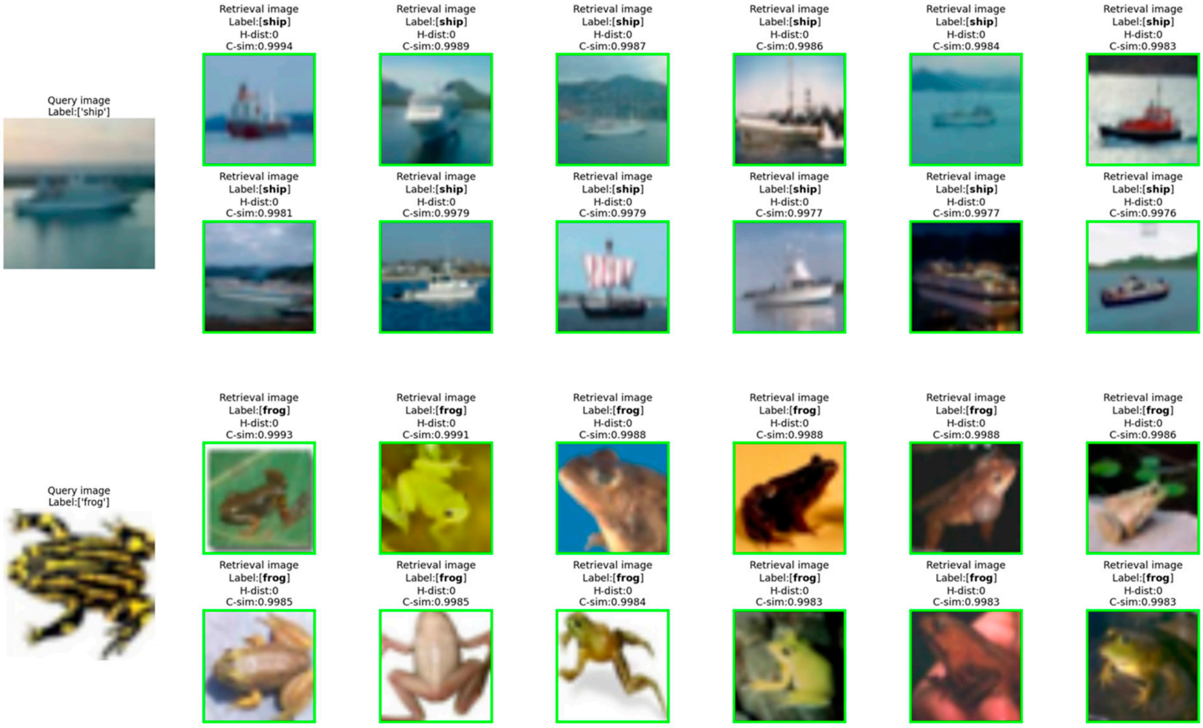


Figure A3. Detailed image retrieval results for CIFAR10 I.

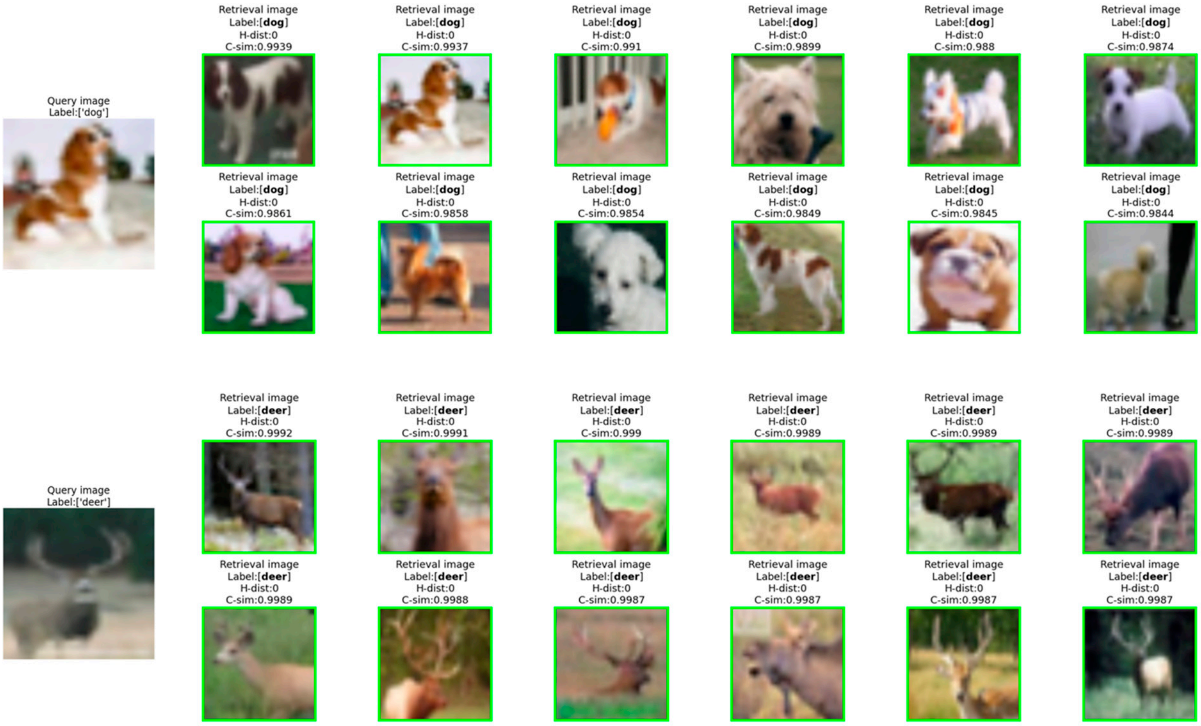


Figure A4. Detailed image retrieval results for CIFAR10 II.

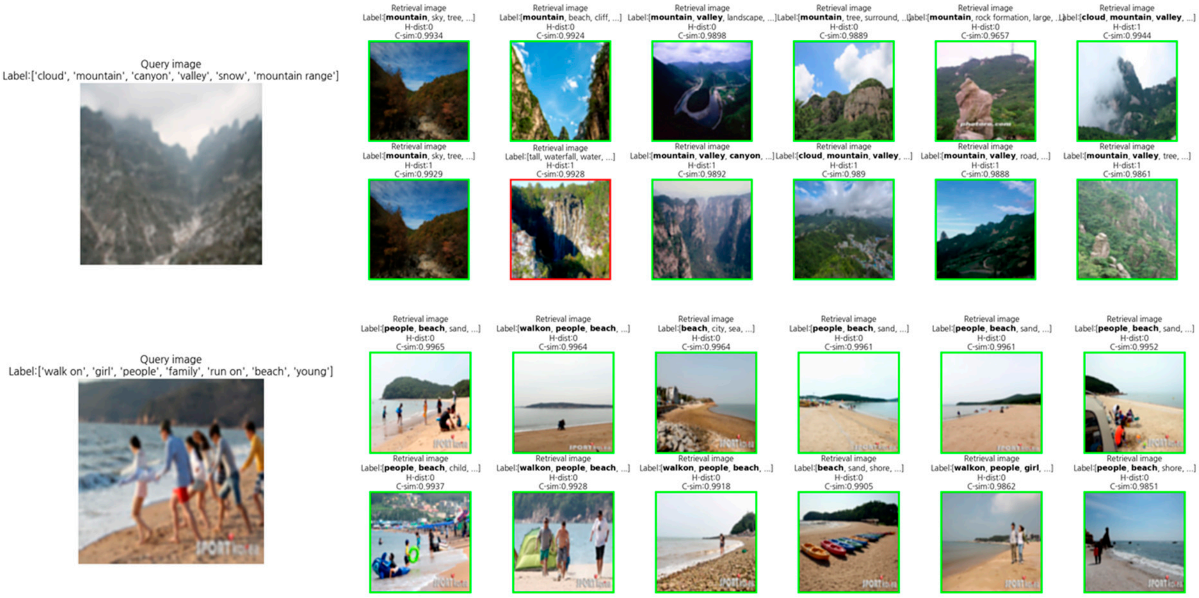

Query image
Label:['cloud', 'mountain', 'canyon', 'valley', 'snow', 'mountain range']
Retrieval image
Label:[mountain, sky, tree, ...]
H-dist:0
C-sim:0.9934
Retrieval image
Label:[mountain, beach, cliff, ...]
H-dist:0
C-sim:0.9924
Retrieval image
Label:[mountain, valley, landscape, ...]
H-dist:0
C-sim:0.9898
Retrieval image
Label:[mountain, tree, surround, ...]
H-dist:0
C-sim:0.9889
Retrieval image
Label:[mountain, rock formation, large, ...]
H-dist:0
C-sim:0.9657
Retrieval image
Label:[cloud, mountain, valley, ...]
H-dist:1
C-sim:0.9944
Retrieval image
Label:[mountain, sky, tree, ...]
H-dist:1
C-sim:0.9929
Retrieval image
Label:[tall, waterfall, water, ...]
H-dist:1
C-sim:0.9928
Retrieval image
Label:[mountain, valley, canyon, ...]
H-dist:1
C-sim:0.9892
Retrieval image
Label:[cloud, mountain, valley, ...]
H-dist:1
C-sim:0.989
Retrieval image
Label:[mountain, valley, road, ...]
H-dist:1
C-sim:0.9888
Retrieval image
Label:[mountain, valley, tree, ...]
H-dist:1
C-sim:0.9861
Query image
Label:['walk on', 'girl', 'people', 'family', 'run on', 'beach', 'young']
Retrieval image
Label:[people, beach, sand, ...]
H-dist:0
C-sim:0.9965
Retrieval image
Label:[walkon, people, beach, ...]
H-dist:0
C-sim:0.9964
Retrieval image
Label:[beach, city, sea, ...]
H-dist:0
C-sim:0.9964
Retrieval image
Label:[people, beach, sand, ...]
H-dist:0
C-sim:0.9961
Retrieval image
Label:[people, beach, sand, ...]
H-dist:0
C-sim:0.9961
Retrieval image
Label:[people, beach, sand, ...]
H-dist:0
C-sim:0.9952
Retrieval image
Label:[people, beach, child, ...]
H-dist:0
C-sim:0.9937
Retrieval image
Label:[walkon, people, beach, ...]
H-dist:0
C-sim:0.9928
Retrieval image
Label:[walkon, people, beach, ...]
H-dist:0
C-sim:0.9918
Retrieval image
Label:[beach, sand, shore, ...]
H-dist:0
C-sim:0.9905
Retrieval image
Label:[walkon, people, girl, ...]
H-dist:0
C-sim:0.9862
Retrieval image
Label:[people, beach, shore, ...]
H-dist:0
C-sim:0.9851

Figure A5. Detailed image retrieval results for KOLAA Image.

References

1. Sohn, K.; Yan, X.; Lee, H. Learning structured output representation using deep conditional generative models. *Advances in Neural Information Processing Systems* **2015**, *28*, 3483–3491.
2. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929,* **2021**. https://arxiv.org/abs/2010.11929.
3. Huiskes, M.J.; Lew, M.S. The MIR Flickr retrieval evaluation. In *Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval*, British Columbia, Vancouver, Canada, 30-31 October 2008; pp. 39-43. https://doi.org/10.1145/1460096.1460104.
4. Krizhevsky, A.; Hinton, G. Learning multiple layers of features from tiny images. *figshare*, 2009. https://www.cs.toronto.edu/~kriz/learning-features-2009.html.

5. Chua, T.S.; Tang, J.; Hong, R.; Li, H.; Luo, Z.; Zheng, Y. NUS-WIDE: A real-world web image database from National University of Singapore. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, Santorini, Fira, Greece, 8-10 July 2009; pp. 1-9. https://doi.org/10.1145/1646396.1646452.
6. Zhang, Y.; Zhang, Y. Research and implementation on digital image copyright protection system. In *Proceedings of the 2012 International Conference on Computer Science and Electronics Engineering*, Hangzhou, China, 23-25 March 2012; Volume 2, pp. 48-51. IEEE. https://doi.org/10.1109/ICCSEE.2012.314.
7. Khare, A.; Singh, U. K.; Kathuria, S.; Akram, S. V.; Gupta, M.; Rathor, N. Artificial intelligence and blockchain for copyright infringement detection. In *2023 2nd International Conference on Edge Computing and Applications (ICECAA)*, Namakkal, India, 19-21 July 2023; pp. 492-496. IEEE. https://doi.org/10.1109/ICECAA58104.2023.10212277.
8. Sun, X.; Zhou, J. Deep perceptual hash based on hash center for image copyright protection. *IEEE Access* **2022**, *10*, pp. 120551-120562. https://doi.org/10.1109/ACCESS.2022.3221980.
9. Kim, D.; Heo, S.; Kang, J.; Kang, H.; Lee, S. A photo identification framework to prevent copyright infringement with manipulations. *Applied Sciences* **2021**, *11(19)*, 9194. https://doi.org/10.3390/app11199194.
10. Yang, J.; Kim, S.; Lee, S.; Kim, W.; Kim, D.; Hwang, D. Robust authentication analysis of copyright images through deep hashing models with self-supervision. *Journal of Universal Computer Science* **2023**, *29(8)*, pp. 938-958. https://doi.org/10.3897/jucs.98824.
11. Zajić, G.; Kojić, N.; Reljin, B. Searching image database based on content. In *Proceedings of the 2011 19th Telecommunications Forum (TELFOR)*, Belgrade, Serbia, 22-24 November 2011; pp. 1203-1206. IEEE. https://doi.org/10.1109/telfor.2011.6143766.
12. Shen, Y.; Qin, J.; Chen, J.; Yu, M.; Liu, L.; Zhu, F.; Shen, F.; Shao, L. Auto-encoding twin-bottleneck hashing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 13-19 June 2020; pp. 2818-2827. https://doi.org/10.1109/cvpr42600.2020.00289.
13. Kipf, T. N.; Welling, M. Semi-supervised classification with graph convolutional networks. *In Proceedings of the International Conference on Learning Representations*, 2020. https://openreview.net/forum?id=SJU4ayYgl.
14. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; Polosukhin, I. Attention is all you need. *In Advances in Neural Information Processing Systems*, Long Beach, California, USA; https://dl.acm.org/doi/10.5555/3295222.3295349.
15. Charikar, M. S. Similarity estimation techniques from rounding algorithms. *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, Quebec, Montreal, Canada, 19-22 May 2002; pp. 380-388. https://doi.org/10.1145/509907.509965.
16. Korean Literature, Academic Works and Art Copyright Association. KOLAA Image Database. KOLAA, 2023. http://www.kolaa.kr.
17. Li, Y.; van Gemert, J. Deep Unsupervised Image Hashing by Maximizing Bit Entropy. *AAAI Conference on Artificial Intelligence* **2021**, *35*, pp. 2002-2010. https://doi.org/10.1609/aaai.v35i3.16390.
18. Qiu, Z.; Su, Q.; Ou, Z.; Yu, J.; Chen, C. Unsupervised Hashing with Contrastive Information Bottleneck. *arXiv preprint* **2021**, arXiv:2105.06138. https://arxiv.org/abs/2105.06138.
19. Liu, H.; Wang, R.; Shan, S.; Chen, X. Deep Supervised Hashing for Fast Image Retrieval. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* Las Vegas, NV, USA, 27-30 June 2016; pp. 2064-2072. https://doi.org/10.1109/cvpr.2016.227.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.