

Sketch_Based Image Retrieval Using Convolutional Neural Network with Multi_Step Training

Azita.Gheitasi¹, Hassan.Farsi¹, Sajad.Mohamadzadeh^{1*}

1.Department of Electrical and Computer Engineering, University of Birjand, Birjand, Iran

Received: 14 Feb 2023/ Revised: 10 Oct 2023/ Accepted: 27 Nov 2023

Abstract

The expansion of touch-screen devices has provided the possibility of human-machine interactions in the form of free-hand drawings. In sketch-based image retrieval (SBIR) systems, the query image is a simple binary design that represents the mental image of a person with the rough shape of an object. A simple sketch is convenient and efficient for recording ideas visually, and can outdo hundreds of words. The objective is to retrieve a natural image with the same label as the query sketch. This article presents a multi-step training method. Regression functions are used in the deep network structure to improve system performance, and various loss functions are employed for a better convergence of the retrieval system. The convolutional neural network used has two branches, one related to the sketch and the other related to the image, and these two branches can have the same or different architecture. After four training steps, a 56.48% MAP was achieved, indicating the desirable performance of the network.

Keywords: Sketch-Based Image Retrieval (SBIR); Deep Learning; Multi-Step Training; Contrastive loss; Triplet loss.

1- Introduction

The advances in multimedia technologies and the widespread use of the Internet have fundamentally changed human life. Audio, video, and images are known as multimedia data and can be useful in various fields, including military, medical, legal, and commercial [1]. Any system that can analyze and recover these data can be efficient and valuable. Among these, images are the most popular multimedia data [2].

The issue of image retrieval can be done in different ways, for example, content-based image retrieval or (CBIR) can be mentioned, which has been of interest in the past [3]. But here, discussed issue is sketch-based image retrieval. As stated, this issue involves retrieving a natural image with the same label as the query sketch. It mainly focuses on extracting representative and shared features from simple sketches and natural images [4]. Scale-invariant feature transform (SIFT) is one of the most common matching methods previously used in the remote sensing image registration[5]. The challenge in SBIR is that free-hand sketches are inherently abstract and symbolic, which magnifies the cross-domain discrepancy between sketches

and the real image. Deep learning methods are used to alleviate this problem [6].

For a better understanding of the subject, a description could be provided about the differences between sketches and real images. Sketches solely have the holistic shape and salient local shapes (and sometimes symbolic colors), while real images have details on shape, color, and texture. Most sketches contain no background, while real images can have cluttered and complex backgrounds. Even when a sketch and an edge map depict the same object or scene, their abstraction levels are dramatically different. This difference is due to the randomness of the sketch lines, simplification and missing details, disproportion, and unrealistic objects (several parts of objects are drawn unrealistically) in sketches [2]. In general, sketches represent the shape and spatial position, while real images include other useful information, such as color and texture [1]. Sketches are considered a highly scattered signal compared to real images, and their analysis is challenging due to the low input information and the abstractness of sketches. Therefore, comparing low-detail images with pixel-dense real images is difficult [7].

A method of collecting sketch data is edge detection techniques and algorithms, such as the fuzzy-based ACO algorithm [8] or using fuzzy cognitive map [9]. This paper, presents a comprehensive investigation of triplet embedding strategies evaluating on three databases (Quick-Draw, TU-

✉ Sajad Mohamadzadeh
s.mohamadzadeh@birjand.ac.ir

Berlin, and Sketchy). Similar to papers on deep networks for object recognition [10], the present study explores appropriate CNN architectures, weight-sharing schemes, and training methodologies to learn a low-dimensional embedding for the representation of both sketches and photographs in practical terms as a space amenable for a fast approximation of the nearest neighbor (ANN) search (e.g., L2 norm) for SBIR. Also, a novel triplet architecture and training methodology is proposed that is capable of generalizing across hundreds of object categories, and its performance is demonstrated in comparison to existing SBIR methods by a significant margin on leading benchmarks.

We propose a multi-step training methodology and investigate several network designs, comparing the Siamese architecture with the Heterogeneous and Hybrid ones. We aimed to develop a training strategy for partial sharing networks.

2- Related works

Sketch-based image retrieval (SBIR) has been studied since the early 1990s, and content-based retrieval (CBIR) were the subject of discussions from 1990 to 1994 [2]. This field remains attractive to researchers. For example, one researcher on CBIR has presented a method based on the combination of Hadamard matrix, discrete wavelet transform (HDWT2), and discrete cosine transform of DCT [11]. From 1994 onwards, studies on sketch-based image retrieval (SBIR) began [2]. Del Bimbo and et al. [12] introduced a module called the object localization, which separated and selected the main areas of an image with the help of rectangles, normalized these windows to be the same size, and then coded their spatial relationships. In this method, only the main subjects of the image were selected and compared. So far, all the reviewed works have employed pixel-based similarity metrics, but these metrics usually require costly computations and have little flexibility. Later, the feature extraction module was introduced to extract various feature types, which were robust to edge variations. Chans et al. [13] believed that users tended to ignore details when drawing the sketches and proposed a curvelet model to extract and encode the prominent edge segments of images. Rajendra and Cheng [14] used a multi-scale representation of edge maps to indicate changes in the level of detail in human-drawn sketches. They believed that the combination of scales preserved the details of the sketch. In another method, a binary mask was used for objects that spatially matched the real image. Another method is gallery displaying module, which uses K-means tree and best-bin-first strategy in combination. The combination of these two algorithms accelerated the recovery speed by several times [2, 15].

Another pixel-based method is OCM, which seeks the closest edge pixel in the sketch that is related to the image. More recently, with the introduction of deep learning and the

use of deep neural networks, research in the field of SBIR took a new form [16, 17]. Convolution networks are comprehensive and efficient in image processing and alleviate numerous deficiencies and ambiguities of data. Neural network-based methods are generally robust in identifying data patterns, superior in speed, flexible against environmental changes, and provide better performance than classic statistical models [18]. Recently, custom architectures such as Alex-Net, Google-Net [19] combined CNN models, and multi-objective ranking networks [20] have been used to rank and predict features. Sketch-A-Net is a deep networks designed for sketch-based image retrieval problem [6].

It explores recognition (rather than search) using a single-branch network resembling a short-form Alex-Net [10]. Sketch-A-Net is a component of the works of Bhattacharjee et al. [21] and Sain et al. [7]. Sketch-A-Net is also explored in the present study and compared with several other contemporary architectures.

An early work on multi-branch networks for sketch retrieval (of 3D objects) was the contrastive loss network by Wang et al. [22], which independently learned branch weights to bridge the domains of sketch and 2D renderings of silhouette edges. In a recent short paper, Qi et al. [23] propose a two-branch Siamese network with contrastive loss. Their results, although comparable with other methods using shallow features, are still far behind state-of-the-art by a large margin. As we show later, learning a single function to map disparate domains to the search space appears to underperform designs where branch weights are learned independently or semi-independently.

Triplet CNNs employ three branches [24, 25]: (i) an anchor branch, which models the reference object, (ii) one branch representing positive examples (which should be similar to the anchor) and (iii) another modeling negative examples (which should differ from the anchor). The triplet loss function is responsible for guiding the training stage considering the relationship between the three models. Triplet CNNs have recently been explored for face identification [26], tracking [27], photographic visual search [28], and sketched queries to refine search within a single object class (e. g. fine-grain search within a dataset of shoes) [7]. Similarly, a fine-grained approach to SBIR was adopted by the recent Sketchy system of Sangkloy et al. [29] in which careful reproduction of stroke detail is invited for object instance search. Researchers report that using a fully-shared network was better than using two branches without weight sharing. However, the authors in [29] suggest it is more beneficial to avoid sharing any layers in a cross-category retrieval context. Also, a hybrid design was explored by Bui et al. [30] using the same architecture on both branches but sharing certain layers. However, as their model learns a mapping between sketch and edge map (rather than image directly) its performance is limited. Furthermore, it is still unclear whether triplet loss works better than contrastive loss.

This paper uses a generic multi-step training methodology for cross-domain learning that leverages several loss functions in training shared networks as illustrated in Figures 1, 2 and 3. Also an extensive evaluation of ConvNet architectures and weight-sharing strategies is carried out.

3- Proposed Method

The present study proposes a multi-step training method and examines several network architectures. This method, training the network independently at first (without sharing the weights) and then training in shared manner to modify and improve the performance. Lastly other data sets are applied to modify the weights of the training system. Two functions are used in this training process: contrast loss and triple loss.

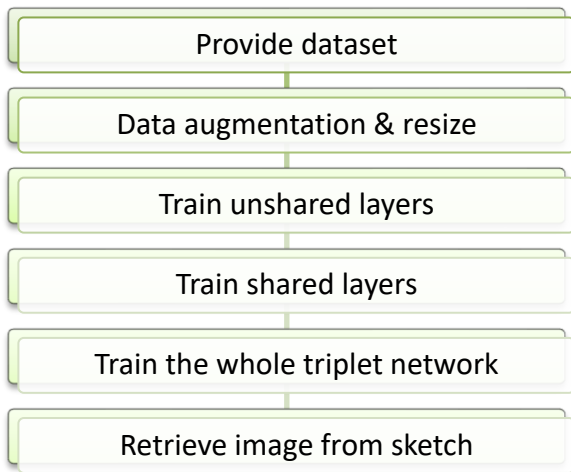


Fig. 1 Block diagram of the multi-step training SBIR system

Figure 1 shows the block diagram of the proposed multi-step training sketch-based image retrieval method. Datasets were collected in the first block. In the upcoming research, three well-known and extensive datasets in sketch-based image retrieval are used: Sketchy, TU-Berlin, and Quick-Draw. In the second block, some pre-processing is done on datasets for equalization before entering into the networks for training. First, all the images were set to 256×256 . Since the datasets contained sketches with different thicknesses, they were all equalized via the thinning method used in [10]. The data augmentation process was done (fully explained in Section 3.3. later). After pre-processing the data, the training phase began. The third block involved the unshared training step, which is the first step of the proposed training. At this step, the training was done independently without sharing the weights of the layers. That is, the sketch branch and the image branch were trained separately using Soft-max loss for a simple classification. The fourth block involved shared training, the second step of the proposed method. In this step,

a two-branch network was formed, and the unshared layers of the previous step were frozen. Soft-max loss and contrast loss (Eq. 3) functions were used to train shared layers in this step. In the next block, the third step of the proposed method, all the layers were defrosted. The training then continued by forming a triplet network and triplet loss and soft-max loss functions. After these steps, the image of the sketch was finally retrieved.

Table 1 shows the summary of the literature.

Table 1. The summary of the literature

<i>step</i>	<i>explanation</i>
1	Collecting datasets: (Sketchy, Tu-Berlin, Quick-Draw)
2	Pre-processing images: (resize all datasets to 256×256 , equalizing, and ...)
3	Unshared training: (for sketch branch training was done independently without sharing the weights of the layers, and for image branch training was done separately using soft-max loss for simple classification.)
4	Shared training: (we have two-branch network. unshared layers of the previous step were frozen. Soft-max and contrast loss functions were used)
5	Training triplet network: (all the layers were defrosted. Training continued by forming a triplet network and triplet and soft-max loss functions.)
6	Retrieval: (the image of the sketch was retrieved)

3-1- Architecture

Investigating a sketch-based image retrieval problem, requires at least one deep convolution bifurcation network. The branch architecture related to sketch and image can be the same or different. This paper, investigated Sketch-A-Net, Alex-Net, VGG-16 and InceptionV1 (Google-Net) for the sketch branch and Alex-Net, VGG-16, and InceptionV1 for the image branch. Low-level features are often learned in the lower layers of the convolutional network, while semantic features are obtained by training the upper layers. Therefore, in this process, the upper layers are trained jointly and the lower layers independently. All possible permutations with the mentioned architectures are explored for the sketch and image branches. When the architectures of the sketch and image branches are completely different, one or more fully connected layers are required to unify the branches.

Here, the loss functions used in the training process are described. Let $X^s = \{x^s\}$ and $X^l = \{x^l\}$ be collections of training sketches and images, respectively. The contrastive loss function accepts a pair of input examples (x^s, x^l) and regresses their embedding closer or pushes them away, depending on whether x^s and x^l are similar [10]. Let Y represents the label of a training pair (x^s, x^l) so that:

$$Y = \begin{cases} 0 & \text{if } (x^s, x^l) \text{ are similar} \\ 1 & \text{if } (x^s, x^l) \text{ are dissimilar} \end{cases} \quad (1)$$

The cross-domain Euclidean distance between the outputs of the two branches is calculated as:

$$D(x^s, x^l) = \left\| F_{\theta_s, \theta_c}^s(x^s) - F_{\theta_l, \theta_c}^l(x^l) \right\|_2 \quad (2)$$

Where parameters θ_s and θ_l represent domain-specific layers, θ_c is the shared part, and $F_{\theta_s, \theta_c}^S(x^S)$ and $F_{\theta_l, \theta_c}^l(x^l)$ are the embedding functions for sketch and image domains, respectively.

The contrastive loss is thus defined as:

$$\mathcal{L}_c(Y, X^S, X^l) = \frac{1}{2}(1 - Y)D^2(X^S, X^l) + \frac{1}{2}Y\{m - D^2(X^S, X^l)\}_+ \quad (3)$$

In which $\{\cdot\}_+$ is hinge loss function, and m is a defining margin and acceptable threshold for the dissimilarity of the sketch and image.

Triplet loss [7] maintains a relative distance between the anchoring example and both a similar and a dissimilar example. For the triplet input (X^S, X_+^l, X_-^l) , where X^S is an anchor sketch, and X_+^l is a similar and X_-^l is a dissimilar image, the triplet loss defined as:

$$\mathcal{L}_c(X^S, X_+^l, X_-^l) = \frac{1}{2}\{m + D^2(X^S, X_+^l) - D^2(X^S, X_-^l)\}_+ \quad (4)$$

The CNN network consists of three branches to accommodate the triplet input (X^S, X_+^l, X_-^l) : a sketch branch

(anchor) and two identical image branches (positive and negative). The value of margin m is set as 0.2 in all experiments Suggested in reference [10].

An intermediate, fully-connected (FC) layer is added without post-activation to learn the dimensionality reduction during the training steps. An embedding layer lower-dim is added between layer FC7 (D= 4096) and the output layer FC8 (D = 250) without activation ReLU (fig.1). The connection from FC7 to FC8 is linear. The presence of the domain reduction layer does not affect the performance of the classification layer.

3-2- Training

The proposed multi-step training has four steps:

- Step 1

In this step, the unshared layers learn the features distinctive to their domain without being mixed with other domains (figure 2).

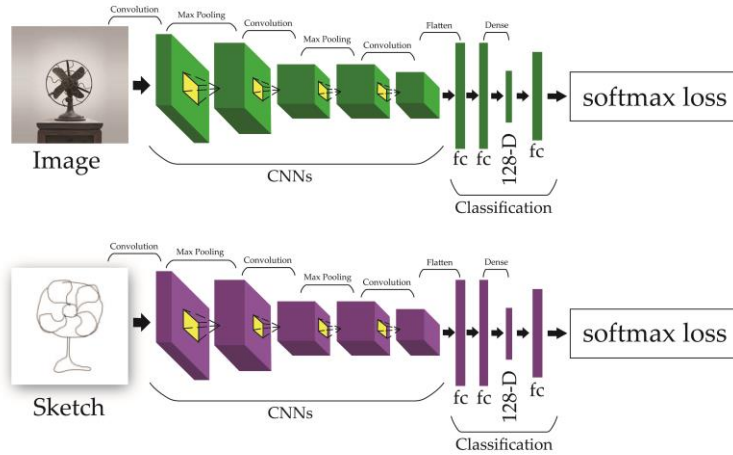


Fig. 2 Training the unshared layers

\mathcal{L}_E and \mathcal{L}_R denote the cross entropy and regularization losses:

$$\mathcal{L}_E(Z) = -\log\left(\frac{e^{zy}}{\sum_i e^{zi}}\right) \quad (5)$$

$$\mathcal{L}_R(\theta) = \frac{1}{2}\sum_i \theta_i^2 \quad (6)$$

So, in step 1, equations 7 and 8 show the representative model for each domain:

$$\arg \min_{\theta_s, \theta_c} \sum_l \mathcal{L}_E(F^S(X_i^S)) + \lambda \mathcal{L}_R(\theta_s, \theta_c) \quad (7)$$

$$\arg \min_{\theta_l, \theta_c} \sum_i \mathcal{L}_E(F^l(X_i^l)) + \lambda \mathcal{L}_R(\theta_l, \theta_c) \quad (8)$$

Where λ is the weight decay term, and θ_c was learned independently.

- Step 2

In this step, the shared layers learn the high-level common features between the two domains by comparing and contrasting the low-level features from both domains (figure 3).

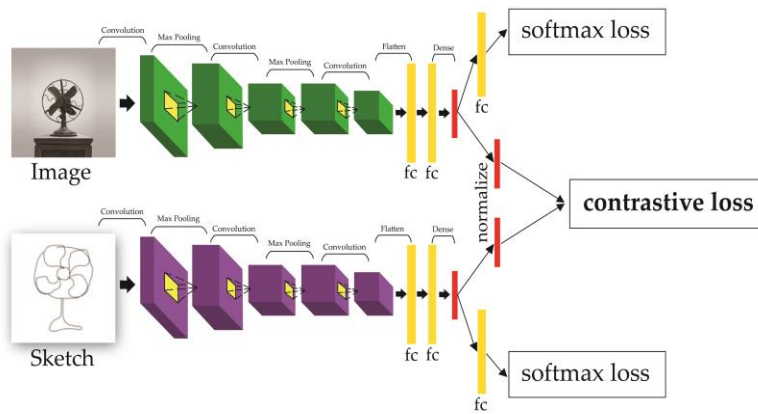


Fig. 3 Training the shared layers

Equation 9 shows the model for the two domains together:

$$\arg \min_{\theta_c} \sum_i \mathcal{L}_E(F^S(X_i^S)) + \sum_i \mathcal{L}_E(F^I(X_i^I)) + \alpha \sum_i \mathcal{L}_C(Y_i, X_i^S, X_i^I) + \lambda \mathcal{L}_R(\theta_C) \quad (9)$$

In which α is the weight of the regression term. As [10] suggests, $\alpha = 2.0$ in all experiments.

- Step 3

In this step, at the beginning of training, two loss functions are applied equally, and then the weight of the triple loss is increased ($\alpha = 2.0$). Figure 3 and Equation 10 display the learning regression in this step.

$$\arg \min_{\theta_S, \theta_I, \theta_C} \sum_i \mathcal{L}_E(F^S(X_i^S)) + \sum_i \mathcal{L}_E(F^I(X_{i+}^I)) + \sum_i \mathcal{L}_E(F^I(X_{i-}^I)) + \alpha \sum_i \mathcal{L}_T(X_i^S, X_{i+}^I, X_{i-}^I) + \lambda \mathcal{L}_R(\theta_S, \theta_I, \theta_C) \quad (10)$$

- Step 4

In this step, the model is modified further by repeating Step 3 on another dataset (figure 4). This training method allows shared and unshared layers to be trained independently in separate steps. In this method, the possibility of partial sharing across the branches is provided, which further reduces overfitting due to the significant reduction of training parameters. At the same time, learning flexibility is maintained for each domain.

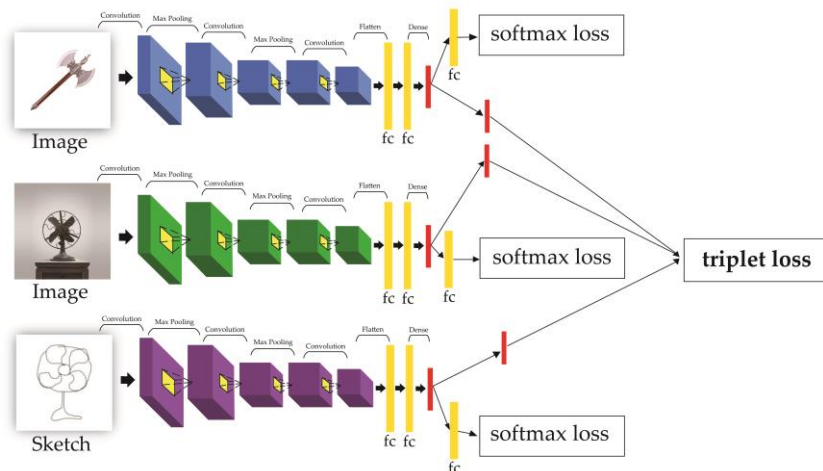


Fig. 4 Training the whole triplet network

It appears that triple and contrast loss functions are important in the training process, but they are not enough to adjust the

training. Therefore, the soft-max loss function was also used in all training steps. Past research has also shown that the

soft-max loss function plays an important role in the convergence of training [10].

3-3- Data Augmentation

Data augmentation is essential in preventing overfitting, especially when the training data is limited. In the proposed method, the following procedures were used to increase the data.

1. A random cut with a dimension of 225×225 as input for Sketch-A-Net network, 227×227 for Alex-net network, and 224×224 for VGG and Inception networks.
2. A random rotation in the range of [-5,5] degrees;
3. A random scaling in the range of [0.9 – 1.1];
4. A random horizontal rotation;
5. The method used only for sketches is called line ranking [10].

This method, is applicable for sketches with at least ten lines. The lines of the sketch are divided into four equal groups based on their importance so that the lines of the first group are the primary lines (the most important lines that related to the more coarse structure of the object) this group of lines is always kept, and the lines of the following groups decrease in importance each time. When one of the groups (except group one) is removed, a new sketch image is obtained every time [10].

4- Exprimental Results

The proposed multi-step training process was tested on several architectures of convolutional networks with sketch and image input. The impact of data augmentation operations on the training process was also evaluated.

4-1- Evaluation Ceriteria

4.1.1 Precision

Precision is one of the most common evaluation criteria used in classification problems. It is based on the ratio of the correctly classified samples to the total number of identified samples (samples that are incorrectly and correctly classified) [31]. The formula for calculating precision is as follows.

$$Precision = \frac{TP}{TP+FP} \times 100 \quad (11)$$

Where TP shows the correctly identified samples and FP shows the misidentified samples.

4.1.2 Recall

The recall is a measure obtained from the ratio of correctly classified samples to the sum of samples that are correctly identified and samples that are incorrectly rejected [32]. It is expressed as the below formula.

$$Recall = \frac{TP}{TP+FN} \quad (12)$$

4.1.3 Mean Average Precision

Average precision is calculated as the weighted mean of precisions at each threshold. The weight is the increase in recall from the prior threshold. The mean average precision is the average of AP of each class [33].

$$AP = \frac{1}{N} \sum_r P_{interp}(r) \quad (13)$$

Where $P_{interp}(r)$ is precision in point recall (r), and MAP is the average AP in each dataset class.

4.1.4 Kendall's Correlation Coefficient (τ_b)

Let $(x_1, y_1), \dots, (x_n, y_n)$ be a set of observations of the joint random variables X and Y, such that all the values of (x_i) and (y_i) are unique (ties are neglected for simplicity). Any pair of observations (x_i, y_i) and (x_j, y_j) , where $i < j$, are considered concordant if the sorting order of (x_i, x_j) and (y_i, y_j) agrees. That is if either both $x_i > x_j$ and $y_i > y_j$ or both $x_i < x_j$ and $y_i < y_j$ are true. Otherwise, they are discordant [10].

The Kendall τ coefficient is defined as:

$$\tau = \frac{(\text{number of concordant pairs}) - (\text{number of discordant pairs})}{\text{number of pairs}} = \frac{1 - \frac{2(\text{number of discordant pairs})}{\binom{n}{2}}}{\binom{n}{2}} \quad (14)$$

In which $\binom{n}{2} = \frac{n(n-1)}{2}$ is the binomial coefficient for the number of ways to choose two from n items.

4-2- Datasets

The proposed networks were evaluated using three datasets.

1) Tu-Berlin:

It is one of the most famous datasets in sketch-based image retrieval and includes 250 classes with 80 images in each, providing a total of 20,000 PNG images of hand-drawn sketches with a size of 128×128 (Figure 6(a)) [34, 35]. This dataset was used for training and testing the first three training steps.

2) Quick-Draw: This dataset has highly simple sketches. It contains 330,000 sketches and 204,000 images with a size of 256×256, divided into 110 classes (Figure 6(b)). It was used to adjust and modify the training model in the fourth step.

3) Sketchy: It is a large dataset of sketches and original images. It contains 75471 hand-drawn images with 125 classes. Of these, 100 classes are shared with the Tu-Berlin dataset, and 25 classes are new

(Figure 6(c)) [36]. This dataset was used to evaluate the proposed model.

Since the Tu-Berlin dataset includes only sketches, Internet databases such as Creative Commons [10] and older datasets such as Flickr-15 [37] or Google search engine were also searched to obtain the original images.

4-3- Training and Testing

A total of 25% of the Tu-Berlin images were selected randomly as the training set, and the remaining 75% were

used as the test set. For simplicity, Sketch-A-Net architecture was used for the sketch branch, and Alex-Net architecture for the image branch. Slight changes were made in the Sketch-A-Net architecture to share the weights between the two networks in such a way that layers 6-7 were taken from the Alex-net network, and layers 4-5 were modified as a combination of the two networks. The sketch branch was trained from the beginning, while the image branch was trained using the pre-trained weights from ImageNet. Figure 5 shows the results of these steps for the proposed multi-step training process.

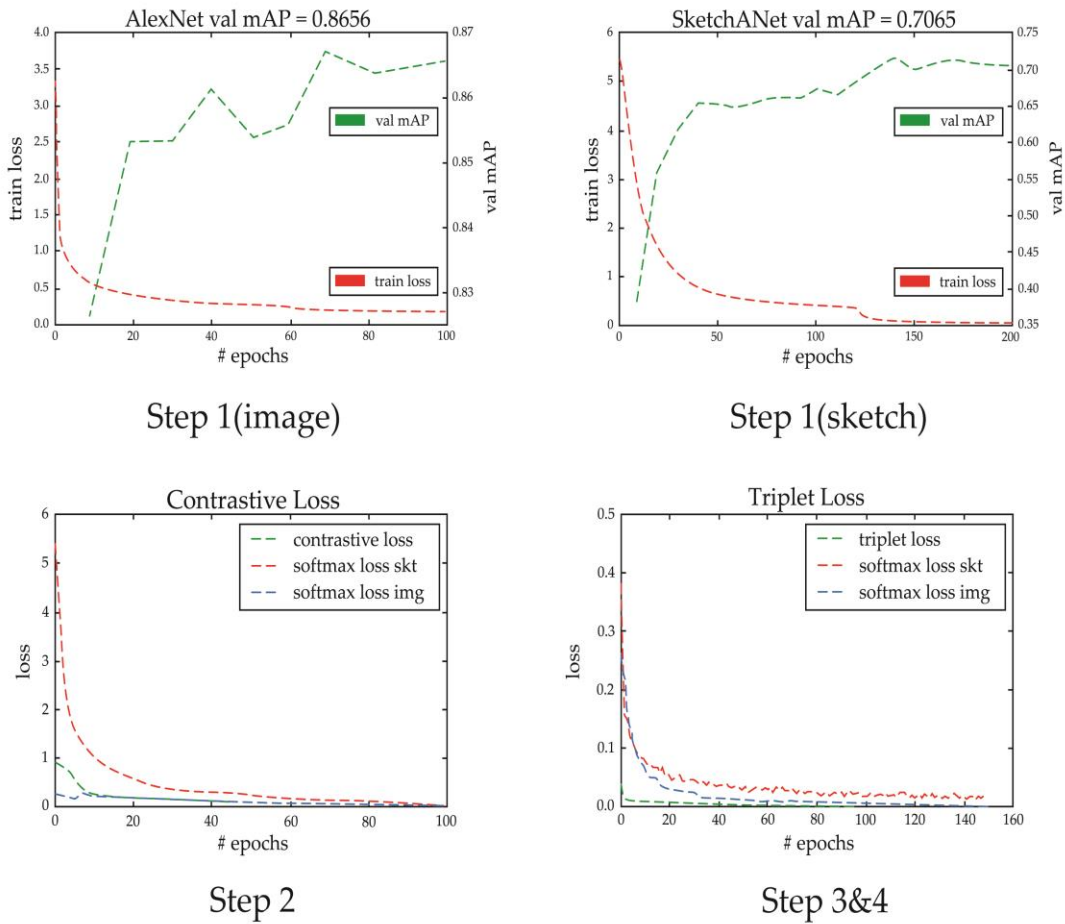


Fig. 5 4 step training of the Sketch-A-Net –and Alex-Net model

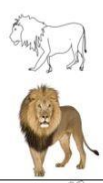








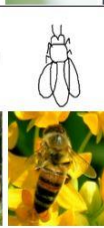


lable dataset	Lion	bee	airplane	sheep
Tu-Berlin				
Quick-Draw				
Sketchy				

Figure 6. Examples of the three datasets used in the multi-steps training SBIR

4-4- Testing Different Architectures

Four different examples of convolution-based network architectures were tested for the sketch and image branches. Different sharing layers were applied for each possible combination according to their architecture and network structure.

The investigations, showed that partial sharing always worked better than full sharing or no sharing at all. However, the layer of each network with the best performance in sharing could only be determined by testing. For example, for Alex-Net - Alex-Net mode, the best performance was achieved when Conv 5 layer was shared. In AlexNet-VGG16, the best performance sharing belonged to sharing the layer FC 7, and in Sketch-A-Net – Alex-Net, sharing layer FC-6 sharing achieved the best performance. In VGG 16-VGG 16, sharing block 5 performed better, and in Inception V1-Inception V1, sharing incept.4e achieved better performance. Subsequently, all possible permutations and sharing were tested to determine the optimal performance of the reviewed architectures. Figure7 shows the results of this review. As the Sketch-A-Net architecture can only be applied to the sketch-edge map mode and does not work on natural images. Therefore, this architecture was not used for the image branches except for one where the images were turned into edge maps.

As the diagram results show, the sketch branch architecture should not be more complicated than the image branch architecture. As can be seen, the designs of VGG16-AlexNet, Inception V1-AlexNet, and Inception V1-VGG16 are better

than their counterparts. Also, if Inception V1 architecture is selected for the image branch, Sketch-A-Net would be more suitable for the sketch branch than Alex-Net or VGG-16, even though it has a simpler architecture.

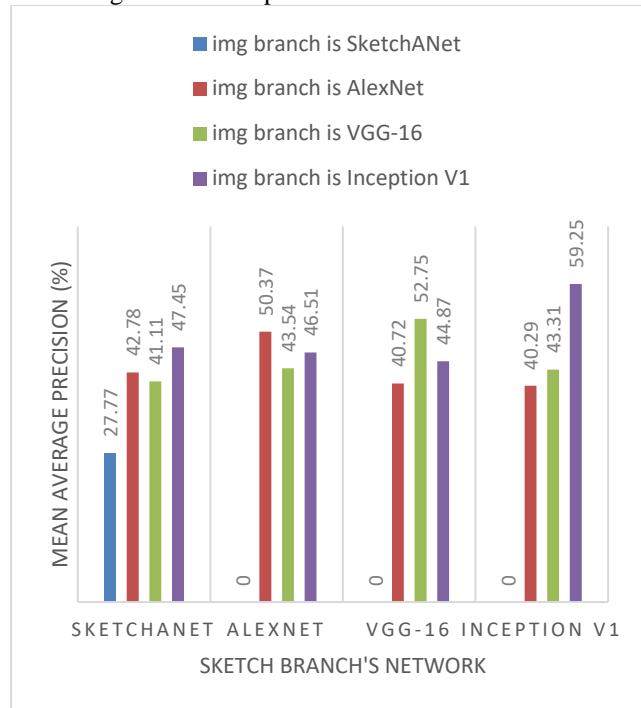


Figure 7. The best performance of different combinations of networks on Sketchy dataset

It can also be seen that using the same architecture for sketch and image branches leads to better performance. Subsequently, the best performance belongs to the design of Inception V1-Inception V1. This architecture was applied to the Sketchy dataset, and the increased output in the range of 64 to 1024 was examined. It was observed that as the dimensions increased, the MAP improved continuously. However, this also led to an increase in the retrieval speed. Therefore, the MAP evaluation criteria and retrieval speed were balanced by selecting a dimension of 256 with a 56.32 map and a recovery time of 6.2 ms for the final model.

4-5- Evaluation the Final Model

The proposed model, using Inception V1-Inception V1, Inception e4 block sharing, and the output dimension of 256 on the Sketchy dataset, was compared with other works. Table 2 shows the comparison of the proposed multistep method with several other research based on the MAP criteria.

Table 2. SBIR comparison based on MAP criteria

<i>method</i>	<i>Dim.</i>	<i>mAP (%)</i>
Siamese with contrastive loss[23]	64	19.54
Rst-SP-SHELO[31]	3060	20.05
Triplet sketch-edgemap[30]	100	24.45
Query-adaptive re-ranking CNN[21]	5120	32.30
Sketchy triplet[29]	1024	35.91
proposed Step 2	256	42.12
sketret [38]	256	43.70
proposed Step 3	256	48.53
cross modal (binary) [39]	64	50.60
cross modal [39]	64	52.30
SBTKNet[40]	512	55.30
hybrid cnn (without shape feature)[41]	64	55.30
proposed Step 4	256	56.48

The Siamese method (Table 2) uses contrastive loss, and introduces. A novel convolutional neural network for SBIR based on the Siamese network [23]. This method primarily draws output feature vectors for input sketch-image pairs with similar labels closer and pushes irrelevant pairs away. This is achieved by jointly tuning two convolutional neural networks which linked by one loss function. As can be seen, the results of this method are lower than all the presented research.

Another method is the Rst-SP-SHELO (Table 2). This method includes RST-SHELO and improved version of SHELO (Soft Histogram of Edge Local Orientations), which is an advanced, efficient method for describing sketches. In this research, the sketch token approach is used to detect image contours utilizing mid-level features. The square root normalization is used for a better normalization of SHELO and improved performance of the retrieval system. The result of this research is marginally better than the Siamese method with contrastive loss but is yet to be desirable.

In the triplet sketch-edge map method [30], convolutional neural networks and triplet loss are used. The SBIR problem is proposed as a cross-domain modeling problem where a depiction invariant embedding of sketch and photo data is learned by regression over a Siamese CNN architecture with half-shared weights and modified triplet loss function. The results of this method are better than the previous two methods but are still insufficient.

Another method shown in Table 2 is the Query-adaptive re-ranking CNN, which uses the localization technique. It also uses the Sketch-A-Net architecture to locate the candidate object proposals, exploit appearance information to resolve the ambiguities in object proposals and refine the search results. In this research, adaptive search is formulated as a subgraph selection problem and solved by the maximum flow algorithm. The results of this method are better than the previous ones (approx. 32.30).

The Sketchy triplet method is used in [29]. This method trains the Sketchy dataset by cross-domain convolutional networks that embed sketches and photos in a common feature space. The results are similar to that of the GN-Triplet network (Google-Net) with triplet loss.

Another example is the SKETRET, which is a ZS-SBIR retrieval method. In this research, a new framework is introduced, which adapts the bi-level domain of sketch and image features using adversarial learning. This framework alleviates the mentioned problems by providing modality-independent features and a class-discriminative latent space. This research achieves slightly better results than the proposed method in the second step.

Binary and non-binary cross-modal methods [39] also involve a ZS-SBIR problem. The study [39] proposes a novel progressive cross-modal semantic network, which first, explicitly aligns the sketch and image features to semantic features and then projects the aligned features to a common space for subsequent retrieval. Cross-reconstruction loss functions are often used to improve the alignment features, and multi-modal Euclidean loss is used for the similarity between the image-sketch pair retrieval features. The results for the binary and non-binary modes (Table 2) are higher than the proposed method in the third step.

SBTK-Net and hybrid CNN (without shape feature) methods achieve similar results (Table 2). In the SBTK-Net method, a simple and efficient framework is proposed that does not require large computational training resources. In the training and inference steps, only one CNN has been used. A pre-trained Image Net CNN (i.e., Res-Net 50) has been set with three learning objectives: Domain balanced quadruplet loss for learning distinctive features; semantic classification loss to preserve the learned semantic knowledge; semantic knowledge preservation loss to reduce the computational cost and increase the accuracy of the process. In the hybrid CNN method (without the shape feature), sketch recognition supposedly benefits from learning the appearance and shape representation. Therefore, a new architecture called hybrid CNN is proposed, that consists of A-NET and S-NET, describing the appearance and shape information, respectively.

As Table 2 shows, the proposed method of the present study achieves higher results after finishing all four steps than other methods.

Table 3 compares the performance of the proposed multi-step training system with other studies based on the percentage of precision criterion.

Table 3. SBIR comparison based on the precision criterion

<i>method</i>	<i>precision (%)</i>
Sketchy triplet[29]	53.42
cross modal (binary)[39]	61.50
cross modal[39]	61.60
proposed Step 2	63.21
proposed Step 3	69.35
fine-grained sbir[42]	78.02
semi supervised learning[16]	76.22
proposed Step 3	78.36

The results show that the Sketchy triplet and binary and non-binary cross-modal methods have a lower precision than the proposed method. The fine-grained SBIR method in [42] investigates the FG-SBIR problem. The introduced [42], FG-SBIR framework [42] starts retrieving as soon as the user starts drawing. Also, a mutual retrieval framework based on reinforcement learning is developed that directly optimizes the rank of the ground-truth photo over a complete sketch drawing episode. In addition, in the semi-supervised learning method, (FG-SBIR), a novel semi-supervised framework for cross-modal retrieval has been introduced, along with a discriminator-guided mechanism to guide against unfaithful generation and a distillation loss-based regularizer to provide tolerance against noisy training samples. In this research, generation and retrieval are considered two conjugate problems, and a common learning method is devised for each module to benefit mutually. These two methods have acceptable precision, but the proposed method achieves better result. After completing the four steps.

Table 4 shows the performance of the proposed multi-step training system using Kendall's correlation coefficient (τ_b) [10].

Kendall's correlation coefficient is used in limited studies on SBIR, but it is a suitable evaluation criterion. As Table 3 shows, the proposed multi-step training method performs better in terms of Kendall's correlation coefficient criterion than methods such as Triplet sketch-edge map and Sketchy triplet.

Table 4. The comparison based on Kendall's correlation coefficient (τ_b)

<i>method</i>	<i>Dim.</i>	τ_b
Triplet sketch-edgemap[30]	100	0.22
proposed Step 2	256	0.33
proposed Step 3	256	0.36
Sketchy triplet[29]	1024	0.37
proposed Step 4	256	0.48

In this article, we investigated the performance of four CNN network architectures and evaluated all possible permutations for the image branch and sketch in order to find the best combination of the network as well as the appropriate loss function with it, in order to optimize and increase the accuracy of retrieve. Our simultaneous attention to the network architecture, different methods of data augmentation and its impact on the training process and finding the appropriate loss function with the help of training weighting for each network combination has made this research unique. On the other hand, we have tried to use

datasets that includes different image styles due to the breadth and diversity of the subject, so that we could investigate and cover the challenges related to the dataset.

5- Conclusions

This paper proposed a hybrid convolutional neural network that uses dual and triple architectures for sketch-based image learning and retrieval. Various experiments and examinations of different convolutional neural networks (e.g., Sketch-A-Net, Alex-Net, VGG-16 and Inception V1), determined the best network architecture combination model for the proposed retrieval system. Regression functions were used in the deep neural network structure to improve system performance. Different layers were tested for weight sharing, and investigations and methods suggestions were carried out for preprocessing the training data. Various Loss functions were used for better convergence of the retrieval system. Three large, well-known datasets (Sketchy, TU-Berlin and Quick-Draw) were used in the training, testing, and evaluation process. Lastly, the final model was examined based on three evaluation criteria: MAP=56.48%, Precision=78.36%, and $\tau_b=0.48$. The entire training process of the proposed model was carried out on Pytorch platform. Further research on this topic could continue by exploring multi-domain learning, for example sketch-photo 3D models mapping or multi-style artwork retrieval. Recently, deep convolutional generative adversarial networks (DC-GANs) have shown great potential for sketch-based issues and so might offer an interesting alternative to SBIR for sketch-photo matching. Currently DC-GANs suffer limitations in variety of object classes that can be explored when trained.

References

- [1] D. Birari, D. Hiran, and V. Narawade, "Survey on Sketch Based Image and Data Retrieval", in ICCCE 2019 Springer, 2020, pp. 285-290.
- [2] Y. Li, and W. Li, "A survey of sketch-based image retrieval", Machine Vision and Applications, Vol. 29, No. 7, 2018, pp. 1083-1100.
- [3] S. Mohammadzadeh, and H. Farsi, "Image retrieval using color-texture features extracted from Gabor-Walsh wavelet pyramid", Journal of Information Systems and Telecommunication, Vol. 2, No. 1, 2014, pp. 31-40.
- [4] L. Liu, F. Shen, Y. Shen, X. Liu, and L. Shao, "Deep sketch hashing: Fast free-hand sketch-based image retrieval", in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2862-2871.
- [5] Z. Hossein-Nejad, H. Agahi, and A. Mahmoodzadeh, "Remote Sensing Image Registration based on a Geometrical Model Matching", Journal of Information Systems and Telecommunication (JIST), Vol. 5, No. 36, 2021, pp. 41.
- [6] J. Zhang, F. Shen, L. Liu, F. Zhu, M. Yu, L. Shao, H. T. Shen, and L. Van Gool, "Generative domain-migration hashing for sketch-to-image retrieval", in Proceedings of the European conference on computer vision (ECCV), 2018, pp. 297-314.
- [7] A. Sain, A. K. Bhunia, Y. Yang, T. Xiang, and Y. Z. Song, "Style me up: Towards style-agnostic sketch-based image

- retrieval", in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 8504-8513.
- [8] Z. Dorrani, H. Farsi, and S. Mohamadzadeh, "Image edge detection with fuzzy ant colony optimization algorithm", *International Journal of Engineering*, Vol. 33, No. 12, 2020, pp. 2464-2470.
- [9] E. Askari, and S. Motamed, "Computational Model for Image Processing in the Minds of People with Visual Agnosia using Fuzzy Cognitive Map", *Journal of Information Systems and Telecommunication (JIST)*, Vol. 2, No. 42, 2023, pp. 102.
- [10] T. Bui, L. Ribeiro, M. Ponti and J. Collomosse, "Sketching out the details: Sketch-based image retrieval using convolutional neural networks with multi-stage regression", *Computers & Graphics*, 2018, Vol. 71, pp. 77-87.
- [11] H. Farsi, and S. Mohamadzadeh, "Combining Hadamard matrix, discrete wavelet transform and DCT features based on PCA and KNN for image retrieval", *Majlesi Journal of Electrical Engineering*, Vol. 7, No. 1, 2013, pp. 9-15.
- [12] A. Del Bimbo, and P. Pala, "Visual image retrieval by elastic matching of user sketches", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 2, 1997, pp. 121-132.
- [13] Y. Chans, Z. Lie, D. P. Lopresti, and S. Y. Kung, "Feature-based approach for image retrieval by sketch", in *Multimedia Storage and Archiving Systems II*, 1997, Vol. 3229, pp. 220-231.
- [14] R. K. Rajendran, and S. F. Chang, "Image retrieval with sketches and compositions", in 2000 IEEE International Conference on Multimedia and Expo. ICME 2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No. 00TH8532), 2000, Vol. 2, pp. 717-720.
- [15] M. Rezaei, and M. Rezaei, "Foreground-Back ground Segmentation using K-Means Clustering Algorithm and Support Vector Machin", *Journal of Information Systems and Telecommunication (JIST)*, Vol. 1, No. 41, 2023, pp. 65.
- [16] A. K. Bhunia, P. N. Chowdhury, A. Sain, Y. Yang, T. Xiang, and Y. Z. Song, "More photos are all you need: Semi-supervised learning for fine-grained sketch-based image retrieval", in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 4247-4256.
- [17] S. Dey, P. Riba, A. Dutta, J. Lladós, and Y. Z. Song, "Doodle to search: Practical zero-shot sketch-based image retrieval", in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2179-2188.
- [18] A. Gheitasi, H. Farsi, and S. Mohamadzadeh, "Estimation of hand skeletal postures by using deep convolutional neural networks", *International Journal of Engineering*, Vol. 33, No. 4, 2020, pp. 552-559.
- [19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, "Going deeper with convolutions", in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1-9.
- [20] Q. Yu, F. Liu, Y. Z. Song, T. Xiang, T. M. Hospedales, and C. C. Loy, "Sketch me that shoe", in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 799-807.
- [21] S. D. Bhattacharjee, J. Yuan, W. Hong, and X. Ruan, "Query adaptive instance search using object sketches", in Proceedings of the 24th ACM international conference on Multimedia, 2016, pp. 1306-1315.
- [22] F. Wang, L. Kang, and Y. Li, "Sketch-based 3d shape retrieval using convolutional neural networks", in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1875-1883.
- [23] Y. Qi, Y. Z. Song, H. Zhang, and J. Liu, "Sketch-based image retrieval via siamese convolutional neural network", in 2016 IEEE International Conference on Image Processing (ICIP), 2016, pp. 2460-2464.
- [24] A. Fuentes, and J. M. Saavedra, "Sketch-qnet: A quadruplet convnet for color sketch-based image retrieval", in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 2134-2141.
- [25] K. R. V. Vakili_Zare, and H. Rezaei, "K_Nearest Neighbor Classification Using Data and Deep Neural Networks", in 3rd International Conference on Soft Computing, 2019, pp. 1034-1040.
- [26] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering", in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 815-823.
- [27] X. Wang, and A. Gupta, "Unsupervised learning of visual representations using videos", in Proceedings of the IEEE international conference on computer vision, 2015, pp. 2794-2802.
- [28] A. Gordo, J. Almazan, J. Revaud, and D. Larlus, "Deep image retrieval: Learning global representations for image search", in European conference on computer vision, 2016, pp. 241-257.
- [29] P. Sangkloy, N. Burnell, C. Ham and J. Hays, "The sketchy database: learning to retrieve badly drawn bunnies", *ACM Transactions on Graphics (TOG)*, Vol. 35, No. 4, 2016, pp. 1-12.
- [30] T. Bui, L. Ribeiro, M. Ponti, and J. Collomosse, "Compact descriptors for sketch-based image retrieval using a triplet loss convolutional neural network", *Computer Vision and Image Understanding*, Vol. 164, 2017, pp. 27-37.
- [31] J. M. Saavedra, "Rst-shelo: sketch-based image retrieval using sketch tokens and square root normalization", *Multimedia Tools and Applications*, Vol. 76, No. 1, 2017, pp. 931-951.
- [32] C. Bai, J. Chen, Q. Ma, P. Hao, and S. Chen, "Cross-domain representation learning by domain-migration generative adversarial network for sketch-based image retrieval", *Journal of Visual Communication and Image Representation*, Vol. 71, 2020, pp. 102835.
- [33] a. P. R. G. G. Rajput, "Sketch Based Image Retrieval in Large Databases Using Edge Features", *International Journal of Recent Technology and Engineering (IJRTE)*, Vol. 08, 2020, pp. 2277-3878.
- [34] A. Dutta, and Z. Akata, "Semantically tied paired cycle consistency for zero-shot sketch-based image retrieval", in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 5089-5098.
- [35] Q. Liu, L. Xie, H. Wang, and A. L. Yuille, "Semantic-aware knowledge preservation for zero-shot sketch-based image retrieval", in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 3662-3671.
- [36] T. Dutta, A. Singh, and S. Biswas, "Style guide: zero-shot sketch-based image retrieval using style-guided image generation", *IEEE Transactions on Multimedia*, Vol. 23, 2020, pp. 2833-2842.
- [37] P. Torres, and J. M. Saavedra, "Compact and effective representations for sketch-based image retrieval", in

- Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 2115-2123.
- [38] R. Chavhan, "Zero-Shot Sketch Based Image Retrieval", INDIAN INSTITUTE OF TECHNOLOGY BOMBAY, 2021.
- [39] C. Deng, X. Xu, H. Wang, M. Yang, and D. Tao, "Progressive cross-modal semantic network for zero-shot sketch-based image retrieval", *IEEE Transactions on Image Processing*, Vol. 29, 2020, pp. 8892-8902.
- [40] O. Tursun, S. Denman, S. Sridharan, E. Goan, and C. Fookes, "An efficient framework for zero-shot sketch-based image retrieval", *Pattern Recognition*, Vol. 126, 2022, pp. 108528.
- [41] X. Zhang, Y. Huang, Q. Zou, Y. Pei, R. Zhang, and S. Wang, "A hybrid convolutional neural network for sketch recognition", *Pattern Recognition Letters*, Vol. 130, 2020, pp. 73-82.
- [42] A. K. Bhunia, Y. Yang, T. M. Hospedales, T. Xiang, and, Y. Z. Song, "Sketch less for more: On-the-fly fine-grained sketch-based image retrieval", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9779-9788.