

Sketch-QNet: A Quadruplet ConvNet for Color Sketch-based Image Retrieval

Anibal Fuentes, Jose M. Saavedra

Impresee Inc.

600 California St, San Francisco, USA

{anibal.fuentes, jose.saavedra}@impresee.com

Abstract

*Architectures based on siamese networks with triplet loss have shown outstanding performance on the image-based similarity search problem. This approach attempts to discriminate between positive (relevant) and negative (irrelevant) items. However, it undergoes a critical weakness. Given a query, it cannot discriminate weakly relevant items, for instance, items of the same type but different color or texture as the given query, which could be a serious limitation for many real-world search applications. Therefore, in this work, we present a quadruplet-based architecture that overcomes the aforementioned weakness. Moreover, we present an instance of this quadruplet network, which we call **Sketch-QNet**, to deal with the color sketch-based image retrieval (CSBIR) problem, achieving new state-of-the-art results.*

1. Introduction

Sketch-based image retrieval (SBIR) is a growing field in computer vision that consists of retrieving a collection of photos resembling a sketched query. Aiming to make the querying process as easy as possible, the input query is formulated as a simple hand-drawing, composed uniquely of strokes as showed in Fig. 1. In this way, people just need to draw what they are looking for or what they are thinking of.



Figure 1. Examples of simple sketches

An SBIR system brings tremendous advantages in different searching contexts. For instance, in e-commerce, the

search process is a critical component since it is the first contact with a potential buyer. Thus, the success of e-commerce will depend on the success of its search engine. In a world where customer engagement seems to be a key concept, SBIR plays an important role, not only in terms of effectiveness but also in terms of customer experience. Moreover, the widespread use of mobile devices fosters this kind of querying modality, since touch-screen interfaces allow people to draw easily and in an entertaining way.

During the last decade, we have seen a resurgence in the interest in SBIR. Works based on low-level features [12, 6, 9] and mid-level representations [13] marked the beginning of this resurgence period. However, the explosion of deep learning-based approaches also brings significant improvements to SBIR models [19, 15]. Furthermore, some researchers were also focused on optimizing the deep visual representations through deep hashing-based approaches [10]. After all, the architectures based on siamese backbones with triplet loss trained incrementally showed the best performance on different SBIR datasets [2]. More recently, a sketch representation generated from a transformer-based model was proposed by Ribeiro et al. [11] with competitive results.

Although sketch-based retrieval has shown to be a powerful tool for searching environments like e-commerce, it also has some limitations related to its expressiveness capacity. The input sketch is composed of strokes that represent the shape of the product that the user is looking for, but it lacks relevant attributes in visual perception like color or textures, which are also relevant in applications like e-commerce search.

Contrary to the advances on sketch-based querying, methods dealing with sketches, including color, like those shown in Fig. 2, have not followed the same direction, having still few works facing this challenging problem [1],[17]. Therefore, our contribution in this work is to propose a quadruplet-based convolutional network architecture that allows us to incorporate an extra visual attribute to the learned feature space. We present an instance of this quadruplet architecture called **Sketch-QNet** focused on the

color sketch-based image retrieval problem. We also describe a methodology for training the proposed architectures because of the scarcity of appropriate datasets.

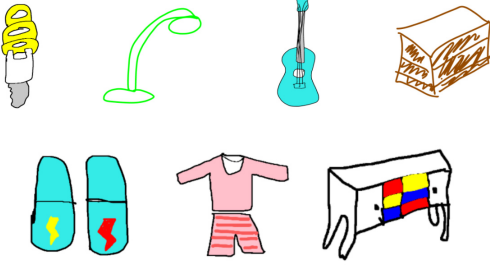


Figure 2. Examples of color sketches

This document is organized as follows: Section 2 describes the related work on CSBIR, Section 3 describes the proposed Sketch-QNet architecture, Section 4 shows a description of our experiments together with the corresponding results, and finally, Section 5 presents the conclusions of our work.

2. Related Work

Sketch-based image retrieval has undergone significant growth during the last decade. Initially, through low-level-based approaches [14, 9, 6] that use oriented gradients as features. Later, with mid-level-based approaches like the work proposed by Saavedra et al. [13], which combines low-level features with a clustering strategy to learn a set of primitives called *keyshapes*, that are then used to represent sketches. However, the big leap in performance came with the explosion of the deep learning-based methods. In this context, the architectures showing the best results on similarity search are those combining siamese nets with triplet loss [19, 2, 15], especially when fine-grained search is desired. Other researchers have also studied architectures focused on obtaining optimized representations through a binary feature space [10].

Among the diversity of SBIR methods, the work of Bui et al. [2] mainly attracts our interest. This method proposes a 4-stage incremental methodology for training a network capable of producing a feature space where sketches and photos can exist together. The four stages are designed in such a way that they can incrementally improve their discriminatory power. To this end, they also use siamese and triplet networks jointly with cross-entropy loss but trained from a coarse-grained similarity at the beginning to a fine-grained similarity at the end. This idea showed to be very effective, as it achieves the state of the art results in different public SBIR datasets with colorless sketches.

In color sketch-based image retrieval, the goal is to retrieve photos sharing both shape and color w.r.t. a given query, in such a way that it allows us to formulate queries

for retrieving targets like *that special red-blue chair* or a *fancy blue lamp*. Although we have seen tremendous advances in the direction of sketch-based retrieval, there are still few works dealing with colorful sketches. A recent approach dealing with color sketches is the method proposed by Dutta and Biswas [5], where a colorless sketch is used to define the shape of the target images, and color information is provided by a keyword. However, in the domain of querying by sketches, we can incorporate visual attributes into the drawing itself, making the query formulation easier for users.

Closer to our proposal is the work presented by Bui and Collomose [1], which processes shape and color independently using the well-known BoW strategy and features produced by gradient fields. Similar to this work is that proposed by Xia et al. [17] which extracts shape features through a convolutional neural network to retrieve the N most similar images. The resulting ranking is then resorted using color histograms. In the two proposals, two scores are computed, one for shape similarity and the other for color. Both scores are merged at the end, setting a weight for each attribute. Contrary to these approaches, we aim to construct a convolutional architecture capable of representing shape and color information in one model.

So far, we know that siamese architectures alongside a triplet loss have shown the best performance in the context of similarity search. These architectures are designed to receive triplets as inputs. A triplet is defined as a (I_a, I_p, I_n) tuple, where I_a is an anchor image, the sketch in our case. I_p is a positive photo that is considered similar to the query, while I_n is a negative photo, one considered different from the sketch. Therefore, under this approach, the model tries to learn a feature space where positive pairs come closer to each other, while negative pairs remain afar. Indeed, triplet loss tries to minimize the following expression:

$$\max(0, \delta(I_a, I_p) + \lambda - \delta(I_a, I_n)) \quad (1)$$

where $\delta(\cdot)$ is a distance function, and λ is the minimum separation between positive and negative distances (the margin).

As we can observe from Eq. 1, given a query q , the triplet loss tries to separate positive photos from negative ones by a margin of λ , no matter how different or similar the positive pairs are from q . When we incorporate other attributes like color, we would expect the feature space to be able to differentiate between photos sharing shape and color from those that only share the shape. Designing an architecture capable of incorporating this kind of extra visual information is a challenge we need to deal with.

Therefore, in this work, we propose a novel quadruplet architecture named Sketch-QNet for learning a feature space in the context of color sketch-based image retrieval where colorful sketches and photos can exist together. Our

work is inspired by siamese networks as well as the incremental methodology proposed by Bui et al. [2].

Close to our proposal is the work of Seddati et al. [16] about quadruplets networks for SBIR. However, there are significant differences. They use a quadruplet architecture in the context of colorless sketch-based retrieval to discriminate between two kinds of positive objects, those sharing the same class and those representing the same instance for an input query. In contrast, our proposal focuses on color sketch-based retrieval. Our model uses a cross-entropy branch to allow the model to retrieve objects from the same class as the query. Furthermore, the architecture we propose is focused on discriminating objects sharing color and shape with the query (*the positives*) from those that only share shape (*the positive-negatives*). Consequently, our approach requires defining a suitable methodology to form up quadruplets that serve as inputs to the proposed model.

3. Sketch-QNet for CSBIR

In this section, we describe **Sketch-QNet**, a quadruplet convnet for CSBIR able to produce a shared feature space for photos and color sketches that takes into account shape and color features. Inspired from [2], we propose a partially shared network and a training methodology in multiple stages. The basis of our proposal is for the network to learn shape-based features in the early stages of the training and jointly learn shape and color features in the last stage. This is achieved by using a quadruplet neural network with a custom loss function.

3.1. Idea Behind The Method

The core of our proposal is the use of a *quadruplet*, which will be used to feed our model during training. A quadruplet Γ_i is defined as follows:

$$\Gamma_i = (q^i, p_+^i, p_{+-}^i, p_-^i) \quad (2)$$

where, i is the i -th input during the training phase, q is a color sketch, p_+ is a photo with the same class and color as q , p_{+-} is a photo of the same instance than p_+ but with a different color, and finally p_- is a photo from a different class than q . A quadruplet must be accompanied by a distance function $D(\cdot, \cdot)$, applied as follows:

$$D_+ = D(q, p_+) \quad (3)$$

$$D_{+-} = D(q, p_{+-}) \quad (4)$$

$$D_- = D(q, p_-) \quad (5)$$

without loss of generality, we define $D(\cdot, \cdot)$ to be the Euclidean distance. Our target goal is to train a network capable of producing a feature space where the following relations must be accomplished:

$$D_+ < D_{+-} < D_- \quad (6)$$

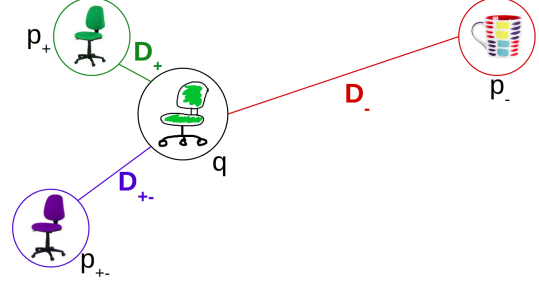


Figure 3. Distances between images must comply $D_+ < D_{+-} < D_-$.

A schematic illustration of how the target feature space should behave is depicted in Fig. 3.

Therefore, we need to appropriately train a convnet for extracting embeddings from photos and sketches in order to fulfill the inequalities in Eq. 6. To solve this problem we separate the equation into two inequalities and add a margin to each one, obtaining Eq. 7.

$$\begin{aligned} D_+ + \alpha \cdot \lambda &< D_{+-} \\ D_{+-} + (1 - \alpha) \cdot \lambda &< D_- \end{aligned} \quad (7)$$

In Eq. 7, λ is a positive number representing the margin between D_+ and D_- , and α is a trade-off parameter between 0 and 1 which indicates how closer is p_{+-} from p_+ and p_- . In this work, p_{+-} is considered as a positive sample, but not as relevant as p_+ . Therefore we choose $\alpha < 0.5$ to allow p_{+-} to get closer to p_+ than to p_- . We will evaluate the impact of α afterwards.

The following two losses are inferred from Eq. 7:

$$\begin{aligned} Loss_{triplet 1} &= \max\{0, D_+ + \alpha \cdot \lambda - D_{+-}\} \\ Loss_{triplet 2} &= \max\{0, D_{+-} + (1 - \alpha) \cdot \lambda - D_-\} \end{aligned} \quad (8)$$

It is interesting to note the dynamic of the proposed loss functions during training. At the beginning, neither $Loss_{triplet 1}$ nor $Loss_{triplet 2}$ are less than 0. Therefore the optimizer will focus on minimizing the expression $D_+ + \lambda - D_-$. It will try to separate positives from negatives samples according to the margin λ . Accomplished that, the optimizer will focus on simultaneously satisfying the two specific losses of Eq. 8.

3.2. Training Quadruplets and Dataset

In this section, we describe how quadruplets are built during training in order to optimize our convolutional architecture according to the proposed loss function (Eq. 8).

As shown in Eq. 2, for each q_i , we need to select three regular images (photos) that complete the quadruplet Γ_i . However, due to the lack of a suitable dataset focused on this problem, we develop a methodology for training our



Figure 4. Example of training photos.

proposed architecture where color queries are generated from a target photo.

For training, we use a collection of photos obtained from an online store. This dataset consists of 10,600 items distributed among 140 classes, mainly formed with kids and household items. Some of the products we can find in this collection belong to categories like toys, clothing, baby items, appliances, sports, among others. Figure 4 shows a sample of the training photos.

Since the system must be able to distinguish color, we increase the number of items in the training collection by changing the colors from the original images. To this end, each item is processed in the HSV color space, adding a constant value to the hue channel. This allows the dataset to grow by a factor of 5. We also generate gray-toned photos of different brightness, transforming the original images to gray-scale, segmenting the foreground by thresholding, and applying a contrast enhancement. The final extended dataset consists of 56,337 items.

A critical problem when training a neural network that learns color sketch-based embeddings is the lack of datasets with plenty of color drawings. To handle this problem, we generate a synthetic color sketch from each item of the training catalog through a function $s(p)$, where $s(\cdot)$ transforms the photo p to its corresponding sketch. This process is illustrated in Fig. 5. First, p is smoothed by an edge-preserving smoothing filter [7], then its color range variation is reduced to have flat colors, and finally, edges are highlighted. To obtain flat colors, we apply k-means clustering in the RGB space over the smoothed images. The number of clusters is selected randomly between 7 and 10. To highlight edges, we apply the Canny edge detector [3] over the smoothed image.

Therefore, a quadruplet Γ_i as defined in Eq. 2 is generated by each item p_i from the training collection. Here the anchor sketch q is $s(p_i)$, p_+ is p_i , p_{+-} is p_i but with a different color, finally, p_- is randomly selected from items having a different class than p_i .

3.3. Backbone Architecture

We use a Squeeze and Excitation [8] ResNext 50 [18] network as backbone. Since this is a cross-domain prob-

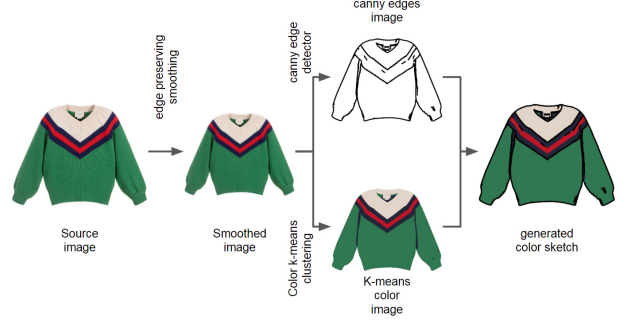


Figure 5. Color sketch generation process.

lem including photos and color sketches, we need two network branches to produce the embeddings. The first layers of both networks use non-shared weights to extract specific features from each domain, while the final layers share weights to learn a common representation between both domains.

At the top of the network, two fully connected layers are used. The last one is devoted to learn a 1024-d feature vector that is then normalized by the Euclidean norm. This layer also feeds a classification layer. The complete Sketch-QNet architecture is presented in Fig. 6.

3.4. Training Procedure

The training procedure is inspired by the work of Bui et al. [2] in which an incremental training procedure is carried out. In our case, we divide the procedure into three stages, as shown in Fig. 6, training each one as follows:

- **Stage 1:** This stage aims to train the backbones before they are plugged into the quadruplet architecture. In this step, each backbone is trained in a classification problem with the classical cross-entropy loss function. The sketch backbone is trained using the sketch dataset proposed by Eitz et al. [6], and the photo backbone is trained on the ImageNet dataset [4].
- **Stage 2:** This stage aims to train both networks together, trying to learn a shared embedding for both, photos and sketches. The loss is defined as a multi-task function that combines contrastive loss with cross-entropy. The network is trained so that the learned embeddings produce short distances between objects from the same class and a large distance between elements from different classes. The definition of the used loss is shown in Eq. 9, where β is set to 2 during training, CE_{sk} is the cross-entropy for sketch classification, and CE_{ph} is for photo classification.

$$loss_{pt 2} = CE_{x_{sk}} + CE_{x_{ph}} + \beta \cdot contrastive(x_{sk}, x_{ph}) \quad (9)$$

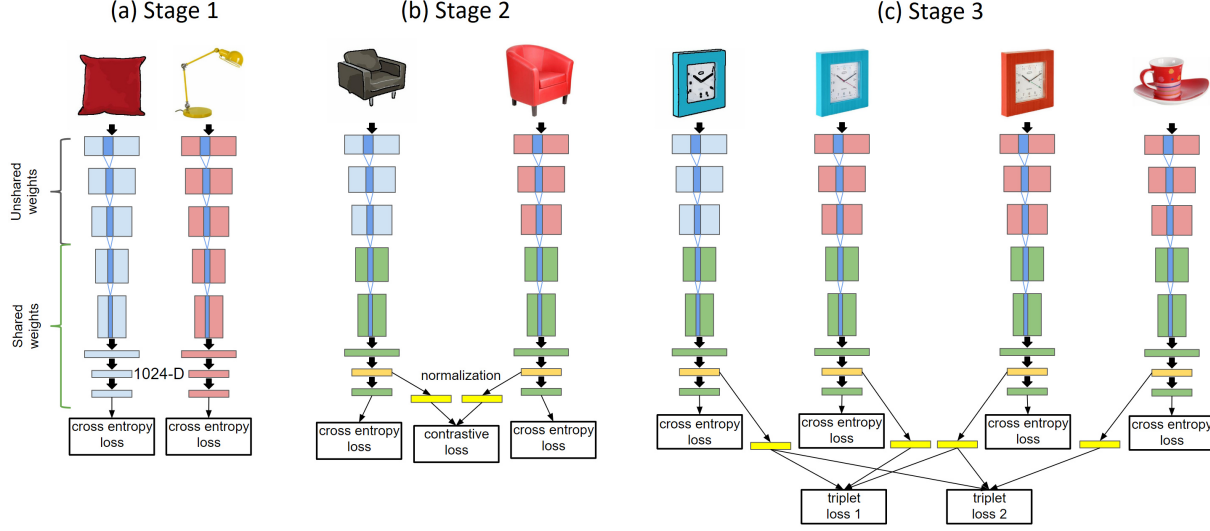


Figure 6. Architecture and training stages.

- **Stage 3:** The goal of this stage is to learn embeddings that take into account color and shape features, according to Section 3.1. Thus, a quadruplet network is trained receiving four inputs, the first one is an anchor sketch and the rest are photos. In this way, we have four branches, the first one processing the input sketch and the other three focused on processing photos.

The objective function to minimize is presented in Eq. 10. It consists of adding the cross-entropy loss to each branch to provide information about the classes, together with the triplet losses given in Eq. 8. The weight β is initialized at 2, and increased by 0.5 each epoch to give more importance to triplet losses as training progresses.

$$loss_{pt\ 3} = CE_{x_{sk}} + CE_{x_{p+}} + CE_{x_{p+-}} + CE_{x_{p-}} + \beta \cdot \{Loss_{triplet\ 1} + Loss_{triplet\ 2}\} \quad (10)$$

4. Experimental Results

4.1. Experimental Settings

We use the following experimental settings for training:

- In stages 1 and 2, the network was trained for 10 epochs, while in stage 3 it was trained for 25 epochs, and the best epoch was selected by evaluating the loss on the validation set.
- The sketches are centered, and white padding of about 10% of the largest dimension is added. In the case of photos, they are resized, keeping their aspect ratio, and adding a white background if necessary. Both sketches

and photos are resized to 224×224 before passing through the network.

- Data augmentation was applied to both photos and sketches, consisting of random rotation between -20 and 20 degrees, random resize between 90% and 110% of the original size of the images, random flip and color augmentation, changing contrast, saturation and the value of the hue channel.

4.2. Baseline Approach

So far we know, the current methods dealing with color sketches for image retrieval are the approaches proposed by Bui and Collomosse [1] using gradient fields together with a BoW strategy; and the proposal of Xia et al. [17] that uses a convnet to extract shape features. In general terms, both proposals are similar as they have two independent paths to process shape and color, producing two scores for each of these attributes. In the end, these scores are weightily merged to compute the final one.

Based on the specifications provided by the aforementioned CBIR proposals, we implemented two baselines that are described below.

4.3. Baseline 1

For shape description, we trained a convnet following the work of Bui et al. [2], using a SE-ResNext-50 [8] as backbone. For color description, we use a color histogram computed from RGB color spaces, quantized into $5 \times 5 \times 5$ cells. In addition, to keep spatial coherence, we split the images into a 2×2 grid, computing the color histogram for each cell, and finally, all of them are concatenated. Therefore, this baseline produces a 500-size color histogram.

For similarity search, we first compute two distances, one for shape and the other for color description that then are merged by a weighted distance function as shown in Eq. 11. Here, $D(S_{sk}, S_{ph})$ is the distance between shape representations of a sketch and a photo, $D(C_{sk}, C_{ph})$ is the distance between the corresponding color based representations, and γ is a weight between 0 and 1.

$$D(sk, ph) = (1 - \gamma) \cdot D(S_{sk}, S_{ph}) + \gamma \cdot D(C_{sk}, C_{ph}) \quad (11)$$

4.3.1 Baseline 2

This baseline uses the same strategy as Baseline 1 to describe shapes and slightly modifies the color representation. The differences in color processing are that we do not use spatial splitting, and only the pixels of strokes are considered to construct the color histogram. A bigger difference is related to the manner in which similarity is computed. Here, for color-based similarity we use the same function as proposed by Bui and Collomosse[1], that is defined as below.

$$\text{Sim}_c(sk, ph_i) = \frac{1}{M_{sk} M_{ph_i}} \times \sum_{b \in sk \cap ph_i} (1 + \ln f_{sk,b})(1 + \ln f_{ph_i,b}) \text{IDF}_b, \quad (12)$$

where $M_{sk} = \sqrt{\sum_{b \in sk} (1 + \ln f_{sk,b})^2}$, $M_{ph_i} = \sqrt{\sum_{b \in ph_i} (1 + \ln f_{ph_i,b})^2}$, $\text{IDF}_b = 1 + \ln \frac{N}{f_b}$, N is the number of indexed images in the dataset, f_b is the number of images containing the bin b , $f_{sk,b}$ and $f_{ph_i,b}$ are the counts of the bin b in the query sketch and the photo i respectively. This similarity measure resembles tf-idf for color space. Finally, the similarity between a sketch sk and a photo ph_i is presented by Eq. 13, which represents the geometric mean between the color similarity, and the cosine distance of the shape features (shape similarity).

$$\text{Sim}(sk, ph_i) = \text{Sim}_c(sk, ph_i)^\omega \left(\frac{S_{sk} \cdot S_{ph_i}}{\|S_{sk}\| \|S_{ph_i}\|} \right)^{(1-\omega)} \quad (13)$$

Both baselines were tested in the Bui's dataset [1], where Baseline 2 reaches a mAP of 0.30, outperforming the performance reported previously (mAP=25.8). In contrast, Baseline 1 yielded a lower performance (mAP=24.55) but very close to the state-of-the-art. These baselines will be used for comparison in the following sections.

4.4. Testing Dataset

Our interest rests on making the querying formulation easier for users, in such a way that they can simply draw-ing what they are looking for, incorporating color information in the drawing itself. Therefore, we are interested in

testing the proposed methods with queries representing the high variability of drawing styles (see Fig. 2). To this end, we present a challenging testing dataset containing sketches that incorporate color in the form of shadows, color stripes, color strokes, etc. To our knowledge, this dataset is the first one that covers such variability of styles, and it is described in the following lines.

The dataset used for testing consists of photos of home & decor items from online stores, distributed in 187 classes, among kitchen items, garden, decoration, and others. It is worth noting that the evaluation dataset is entirely different from the one used in training. The number of items in the testing catalog is 10560, which was increased to 50482, by using the color augmentation technique presented in section 3.2. Additionally, we collected 446 hand-drawn sketches based on photos from the dataset. To this end, we asked ten different people to make drawings resembling some example photos. Each target photo was shown for a few seconds, and then the user drew it from memory.

To evaluate the performance of the described methods, we employ the following metrics, reminding that each sketch has only one groundtruth photo:

- **Mean Reciprocal Rank (MRR)** : This measures the average of the inverse rank of the correct answer (groundtruth photo).
- **Recall Ratio (RR)** : This shows the percentage of queries that retrieve the groundtruth photo after retrieving N items.
- **Mean Average Precision (mAP)** : We use mAP to evaluate the precision of retrieving photos from the same class of the query (regardless of the color).

4.5. Baseline Results

4.5.1 Baseline 1

Table 1 shows the MRR and mAP achieved by the first baseline approach for different values of γ in the range $[0, 1]$. We can observe that smaller values of γ produce higher mAP, ignoring color information, focusing instead on the class of the objects. On the other hand, as γ increases, then the MRR also increases because the retrieved photos with similar color as the query go to the first positions. This happens until a point where both metrics start decreasing because the retrieval system begins to overuse the color features. We select $\gamma = 0.6$ as the best value because it reaches a good trade-off between MRR and mAP (0.1511 and 0.1437, respectively).

4.5.2 Baseline 2

Table 2 shows the MRR and mAP achieved by this approach for different values of ω in the range $[0, 1]$. As in the first

γ	0.0	0.2	0.4	0.6	0.8	1.0
MRR	0.081	0.105	0.138	0.151	0.143	0.022
mAP	0.160	0.163	0.160	0.143	0.095	0.011

Table 1. MRR and mAP for the baseline 1 changing the value of γ .

ω	0.0	0.05	0.1	0.2	0.4	0.6
MRR	0.081	0.150	0.150	0.118	0.057	0.030
mAP	0.160	0.125	0.099	0.057	0.024	0.013

Table 2. MRR and mAP for the baseline 2 changing the value of ω .

baseline, ω has a similar effect to γ , but achieves a lower performance than the first baseline. This is related to the fact that the method does not consider spatial information in the color features.

4.6. Sketch-QNet Results

We follow the training procedure explained in section 3.4. We train the network with the loss of Eq. 8 with $\lambda = 1.5$ and $\alpha \in \{0.1, 0.25, 0.5, 0.75\}$. The evaluation metrics from stages 2 and 3 of the training are shown in Table 3. The best mAP is reached at stage 2 of the training process because this stage considers class information only. On the other hand, in stage 3, mAP decreases, but MRR increases more than 50% compared to stage 2. This growth in MRR is explained by the fact that stage 3 focuses on bringing the photos of the same **color and shape** as the query closer to it in the feature space. We can also note that higher values of α make mAP decrease because when α is larger, photos of the same class but with a different color than the sketch are mapped further away from it. This effect shows that λ in our proposal has a similar behavior as the γ parameter in the first baseline approach.

Stage	Stg. 2	Stg 3 $\alpha=0.10$	Stg. 3 $\alpha=0.25$	Stg. 3 $\alpha=0.50$	Stg. 3 $\alpha=0.75$
MRR	0.126	0.208	0.206	0.197	0.192
mAP	0.186	0.133	0.083	0.051	0.037

Table 3. MRR and mAP for Sketch-QNet at different stages of training.

Another way to compare the performance of the proposed methods is through a recall ratio chart. In Fig. 8 we present the recall ratio for the baseline 1 with $\gamma = 0.6$, the stage 2 of Sketch-QNet and the stage 3 of Sketch-QNet with $\alpha = 0.10$. The figure shows that the proposed method in stage 3 has a higher performance than the baseline. For instance, in stage 3 only 37 photos need to be retrieved (from the 50,482 dataset images) to get a relevant result for 50% of the queries, while for the baseline, 72 images must be

retrieved to reach the same percentage. Of course, it is important to note the performance of a method when retrieving a small number of photos, since users commonly prefer to change the query if there is not a hit among the first results.

In Figure 7, we include examples of different retrieved photos, for the baseline 1 and for stage 3.

5. Conclusions

In this work, we present a novel quadruplet-based convnet architecture together with a loss function that extends the behavior of siamese and triplets networks in the context of the similarity search problem, which is limited to handle positive and negative pairs only. Our proposal allows us to incorporate new intermediate pairs, which can be selected according to visual similarity with the anchor. Therefore, this proposal allows us to generate a feature space that can additionally discriminate among positive results, adding an extra visual feature to the model like color or texture.

To show the advantages of our proposal, we trained an instance of the quadruplet model named Sketch-QNet, that achieves new state-of-the-art results on the problem of color sketch-based retrieval. In addition, to handle the lack of datasets in this problem, we also present a methodology for generating a training dataset from a collection of photos.

References

- [1] Tu Bui and John Collomosse. Scalable sketch-based image retrieval using color gradient features. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1–8, 2015. 0, 1, 4, 5
- [2] Tu Bui, Leonardo Ribeiro, Moacir Ponti, and John Collomosse. Sketching out the details: Sketch-based image retrieval using convolutional neural networks with multi-stage regression. *Computers & Graphics*, 71:77–87, 2018. 0, 1, 2, 3, 4
- [3] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, Nov 1986. 3
- [4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 3
- [5] T. Dutta and S. Biswas. s-sbir: Style augmented sketch based image retrieval. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 3250–3259, 2020. 1
- [6] Mathias Eitz, James Hays, and Marc Alexa. How do humans sketch objects? *ACM Trans. Graph. (Proc. SIGGRAPH)*, 31(4):44:1–44:10, 2012. 0, 1, 3
- [7] Eduardo S. L. Gastal and Manuel M. Oliveira. Domain transform for edge-aware image and video processing. *ACM TOG*, 30(4):69:1–69:12, 2011. Proceedings of SIGGRAPH 2011. 3
- [8] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. *CoRR*, abs/1709.01507, 2017. 3, 4



Figure 7. Example of top 20 retrieved photos for baseline 1 (left) and stage 3 of S-QNet (right).

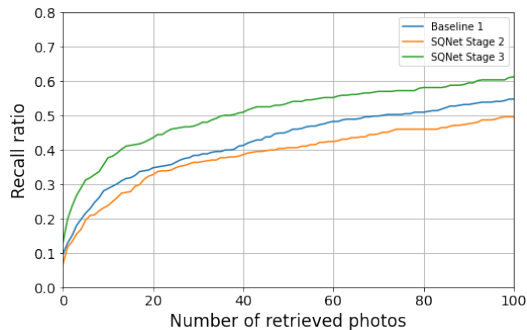


Figure 8. Recall ratio for baseline 1, stage 2 and 3 of Sketch-QNet.

- [9] Rui Hu and John Collomosse. A performance evaluation of gradient field hog descriptor for sketch based image retrieval. *Computer Vision and Image Understanding*, 117(7):790–806, July 2013. 0, 1
- [10] L. Liu, F. Shen, Y. Shen, X. Liu, and L. Shao. Deep sketch hashing: Fast free-hand sketch-based image retrieval. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2298–2307, 2017. 0, 1
- [11] Leo Sampaio Ferraz Ribeiro, Tu Bui, John Collomosse, and Moacir Ponti. Sketchformer: Transformer-based representation for sketched structure. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 0
- [12] J. M. Saavedra. Sketch based image retrieval using a soft computation of the histogram of edge local orientations (s-helo). In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 2998–3002, Oct 2014. 0
- [13] Jose M. Saavedra and Juan Manuel Barrios. Sketch based image retrieval using learned keyshapes (LKS). In *Proceedings of the British Machine Vision Conference 2015, BMVC 2015, Swansea, UK, September 7-10, 2015*, pages 164.1–164.11, 2015. 0, 1
- [14] Jose M. Saavedra, Benjamin Bustos, Tobias Schreck, Sang Min Yoon, and Maximilian Scherer. Sketch-based 3D Model Retrieval using Keyshapes for Global and Local Representation. In *Eurographics Workshop on 3D Object Retrieval*, pages 47–50, 2012. 1
- [15] Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. The sketchy database: Learning to retrieve badly drawn bunnies. *ACM Transactions on Graphics (proceedings of SIGGRAPH)*, 2016. 0, 1
- [16] Omar Seddati, Stéphane Dupont, and Saïd Mahmoudi. Quadruplet networks for sketch-based image retrieval. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, page 184–191, New York, NY, USA, 2017. 2
- [17] Yu Xia, Shuangbu Wang, Yanran Li, Lihua You, Xiaosong Yang, and Jian Jun Zhang. Fine-grained color sketch-based image retrieval. In *CGI*, 2019. 0, 1, 4
- [18] Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. *CoRR*, abs/1611.05431, 2016. 3
- [19] Q. Yu, F. Liu, Y. Song, T. Xiang, T. M. Hospedales, and C. C. Loy. Sketch me that shoe. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 799–807, 2016. 0, 1