Leveraging Unlabeled Data for Sketch-based Understanding

Javier Morales
Department of Computer Science
University of Chile, Chile
javiermoralesr95@gmail.com

Nils Murrugarra-Llerena
Department of Computer Science
Weber State University
nmurrugarrallerena@weber.edu

Jose M. Saavedra
Universidad de los Andes, Chile
jmsaavedrar@miuandes.cl

Abstract

Sketch-based understanding is a critical component of human cognitive learning and is a primitive communication means between humans. This topic has recently attracted the interest of the computer vision community as sketching represents a powerful tool to express static objects and dynamic scenes. Unfortunately, despite its broad application domains, the current sketch-based models strongly rely on labels for supervised training, ignoring knowledge from unlabeled data, thus limiting the underlying generalization and the applicability. Therefore, we present a study about the use of unlabeled data to improve a sketch-based model. To this end, we evaluate variations of VAE and semi-supervised VAE, and present an extension of BYOL to deal with sketches. Our results show the superiority of sketch-BYOL, which outperforms other self-supervised approaches increasing the retrieval performance for known and unknown categories. Furthermore, we show how other tasks can benefit from our proposal.

1. Introduction

Sketch-based understanding plays an important role in the visual perception system. During the half last century, Hubel and Wiesel [12] had already shown how the biological visual cortex highly responds to edge patterns. More recently, Walther et al. [26] also showed the semantic power of image contour information through a study of functional Magnetic Resonance Imaging (fMRI). They found that the primary visual cortex produces similar responses when stimulated by a regular image or by its corresponding contour map.

Sketch understanding is deeply connected to cognition development [6]. Sketching is the means through which an infant starts to understand the natural environment, and also it enables people to externalize and communicate simple and complex ideas. Indeed, people draw schemes or maps to understand complex structures and unfold complex processes. In this vein, Mukherjee et al. [18] studied how we effortlessly associate a drawing with objects in the world. The authors found that the compositional nature of object concepts allows us to decompose objects and drawings into semantically meaningful parts.

Due to the critical role that sketch-based understanding plays in the visual perception process, together with the ubiquitous use of touch-screen devices that make sketching a convenient mechanism, the computer vision community has started to pay special attention to this area. For instance, the main computer vision conferences already include workshops to promote research and applications on this topic. In this vein, we have seen advances in a diversity of tasks like sketch classification [5, 28, 33], sketch-guided object localization [24], sketch-based image and video retrieval [2, 4, 7, 19, 20, 23, 32], sketch-to-photo translation [3, 21], among others.

However, as far as we know, the sketch-based models strongly rely on labeled data [27, 28]. These models need sketches to be annotated with their classes or connected with corresponding images (making pairs) to train supervised models. Having this strong dependence on labeled data raises three critical problems: i) it limits the applicability as labeling is an impractical task for industry, ii) it wastes a vast amount of unlabeled data, and iii) it limits the generalization of learned representations.

This work aims to tackle these limitations by leveraging unlabeled data and creating accurate representations from sketches. We study self-supervised approaches like VAE [15] and BYOL [8]), semi-supervised approaches and traditional supervised models [10] for sketch retrieval. Our semi-supervised VAE baselines adapt VAE, and add a classification branch via sampling concatenation or classifica-

tion loss. Also, we extend BYOL to work in the sketch domain.

We compare the proposed approaches under known and unknown categories. For known categories, the best performer is a supervised model (ResNet-50), as expected. However, it does not generalize well for unknown categories. While sketch-BYOL shows competitive performance for known categories, and is the best performer for unknown categories, showing a better generalization power. This finding was confirmed by embedding visualization and sketch retrieval examples. BYOL better differentiates categories, and shows more intuitive retrievals. Furthermore, we present an example of the utility of our approach to allow self-supervision in other tasks where making sketch-image pairs is a critical stage like sketch-to-photo generation, sketch-based localization, and sketch-based image retrieval.

In summary, our main contributions are:

- A study of multiple ways of mine unlabeled data to improve sketch understanding. This study considers semi-supervised and self-supervised approaches. For self-supervised models, we propose sketch-BYOL discovering which transformations are effective for sketch understanding.
- An strategy to allow self-supervision in tasks where making sketch-image pairs is critical.

2. Related work

Sketching is a new emerging modality with its characteristics and challenges. Sketches can communicate abstract ideas from humans to machines [19, 20], and they are subject to different human drawing styles [31]. Also, as opposed to a static image pixel representation, sketches can be modeled as a temporal stroke sequence [9, 28], and also as topological representations via graphs [27, 30].

Related to improving sketch-based retrieval, [28] develop a novel sketch hashing retrieval technique and a CNN--RNN network to understand millions of sketches accommodating their large variations in styles and abstractions. Similarly to combining CNN and RNN, [31] combines textual convolutional network with CNNs to create a self-supervised representation for sketches. Their main contribution is a set of geometric deformation to create variability and diversity in sketches, and they serve as pretext tasks for self-supervised learning. Also, from unsupervised learning, [1] learn a latent space to group different "visual prototypes" using a clustering layer.

Our work complements these efforts, and similarly to [31] uses self-supervised learning. Similarly, we identify sketch transformations such as rotation, line skip, flip, and crop under a BYOL framework [8].

2.1. Sketch-based classification

Image classification is the most popular task in computer vision. A diversity of models have been proposed for classification [5, 33] or learning representation from sketches [9, 28, 29] achieving high accuracy. These advances were achieved due to the availability of sketch datasets like QuickDraw or Sketchy [21].

Although we have seen good results in public datasets, we have a critical limitation in industry application. The models rely on a huge amount of labeled data, which is scarce or impractical in applications like e-commerce search engines. In this work, we propose to leverage unlabeled sketches to improve retrieval power, especially for unseen categories.

2.1.1 Sketch-based image retrieval

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. Recent works in this task include that of Bui et al. [2], proposing an incremental training process based on siamese networks; the work of Torres and Saavedra [23] that showed the effectiveness of learning low-dimensional embedding using a local-topology preserving dimensional reduction [17]. A natural extension of SBIR is the case where the input sketch includes color information. Here, Fuentes and Saavedra [7] recently presented an interesting approach extending the notion of triplets to quadruplet-based training.

As opposed to these related work, we deal with sketch retrieval under semi and self-supervised learning. We also show how our results are applied to increase the variability in making sketch-image pairs for training a sketch-based image retrieval model.

2.1.2 Sketch-based localization

The idea for a model is to localize all instances of an object in a regular image (scene). A sketch represents the target object. The model should respond with a bounding box enclosing the target object. In this context, Tripathi et al. [24] combines a siamese network, cross-attention, and a region proposal model to train a generalized sketch-based localization model.

Our results yielded by our proposal sketch-BYOL can also be applied to support this task, as we can add variability to the query sketch during the training stage.

2.1.3 Sketch-to-photo translation

Sketch-to-photo translation aims to produce a photorealistic image from an input sketch. Researchers have proposed a diversity of approaches to deal with this problem [3, 13, 21, 34], but to produce plausible results, they strongly depend on labeled sketch-photo pairs.

Our proposal can also fuel this task by generating sketch-photo pairs, in a self-supervised regimen, from edge maps to hand-drawn sketches.

2.1.4 Sketch-based video retrieval

Sketching is a powerful tool for representing static objects and dynamic scenes like videos. If an image is worth more than a thousand words, a sketch may be worth more than multiple images. For instance, a simple drawing representing a person with a left arrow can express the situation when someone moves in the right direction. Thus, sketch-based video understanding is another attractive task in this domain. Here, Collomosse et al. [4] introduced sketches for content-based video retrieval. More recently, Xu et al. [32] proposed a convnet-based model for fine-grained video retrieval, combining appearance and motion information with a relation module between sketch-video pairs.

As we can see, the last years have been marked by significant advances in the development of models or architectures addressing diverse problems based on sketch understanding However, the discussed advances share a common weakness. All of them depend on a huge amount of labeled data, which sets a limitation in real-world applications.

Therefore in this work, we explore and evaluate a diverse set of self and semi-supervised models in the sketch domain. We evaluate generative models like VAE [15] and discriminative models like BYOL [8]. We evaluate our results in terms of how well our models generalize to unknown objects, and particularly to unknown classes.

Furthermore, building image and sketch pairs is a traditional annotation process for the tasks above, trained under a supervised learning strategy. This process places a challenge on a self-supervised strategy. We will show that our proposal is an efficient and effective way to deal with this challenge. Having an image, we could start with its corresponding contour map and search for similar human-drawn sketches to add variability to the initial contour. This could also be regarded as a sketch-based augmentation.

2.2. Self-supervised learning

Self-supervision was mainly related to reconstruction-based generative models like Variational Autoencoder (VAE) [15]. It receives an input and encodes it to a low-dimensional vector, then decodes that vector to reconstruct the same input. VAE encoder produces two vectors a μ and a $log\ \sigma^2$, that together define a conditional probability distribution given the input.

However, more recently, we have seen high effectiveness of discriminative models. Here, Grill et al. [8] proposed BYOL achieving high performance on image representation learning. It comprises two networks, an online network, and a target network. BYOL is fed by two views from the same input image applying two different image transformations. It is then trained so that both networks produce the same latent vectors.

Therefore, inspired by BYOL, we propose sketch-BYOL working in the sketch domain, identifying specific transformations for increasing accuracy.

2.3. Semi-supervised learning

M2 [14] combines the output of an encoder with a label for reconstruction. The model uses the real label from the labeled data and the predicted label from the unlabeled data. A variation of this process is the **Y** shaped model, where a classifier is trained only with the labeled data.

We adapt and evaluate these two approaches for the sketch domain. Additional details come in the next section.

3. Approach

3.1. Self-supervised approaches

3.1.1 Variational Autoencoder (VAE)

The proposed architecture is shown in Figure 1, we utilize a ResNet50 [10] as the encoder and an inverted ResNet50 for the decoder. We also use two fully-connected layers to extract latent vectors (μ and $log\ \sigma^2$) from the encoder. The model considers sketches of size 256×256.

For sketch retrieval purposes, we utilize μ vectors with size of 32[1].

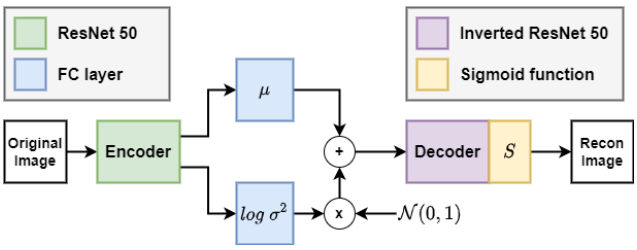

Figure 1. Proposed architecture for the VAE model. the encoder consists of a ResNet50 model and the latent space vectors are extracted with fully-connected layers. The decoder is an Inverted ResNet50 with a sigmoid activation function due to binary representation for sketches.

We represent sketches with strokes (value 0) and background (value 1). Then, we simplify the decoder output

[1]From preliminary experiments, this configuration achieves the best performance

with a sigmoid function and use a pixel-wise binary cross-entropy loss for reconstruction. We also use the KL divergence (KLD) loss to better distribute the categories. KDL is weighted by $\beta = 0.1$ [11], which showed to improve the results. Thus, the VAE unlabeled loss $u\mathcal{L}_{VAE}$ is defined in Equation 1.

$$u\mathcal{L}_{VAE} = reconstruction + \beta KLD \tag{1}$$

3.1.2 Sketch-BYOL

We follow the same architecture from BYOL [8], depicted in Figure 2. It has a ResNet50 for both encoders and an MLP, consisting of a fully connected and a regularization layer with a ReLU activation. The model receives 224×224 sketches whose values range between 0 and 255. Both the online network and the target network are initialized with weights pre-trained on ImageNet [16]. We use the original squared $L2$ norm between the prediction and the target vectors.

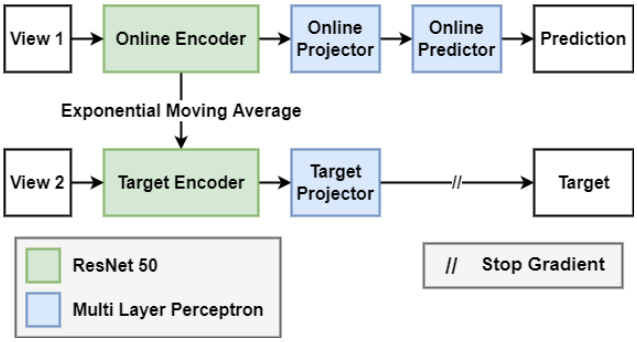

Figure 2. Architecture of sketch-BYOL. It has two encoders based on a ResNet50 model, two projectors and a predictor, each having a fully-connected and a regularization layer with a ReLU activation function. The lower branch produces a target vector for the upper branch to predict.

This model receives two different views or transformations of the same sketch on each branch. In the image domain, researchers identify six transformations with different selection probabilities. These include color variations and Gaussian filtering. However, these transformations do not make sense in a sketch context. Thus, we propose a set of four specific transformations for sketches, which were selected via ablations studies. These transformations are:

- **Random Line Skip (probability 0.5):** We randomly delete 10% of the lines in the sketch.
- **Random Rotation (probability 0.5):** We randomly rotate the sketch with an angle between -30 to 30 degrees.
- **Random Horizontal Flip (probability 0.5):** We randomly flip the sketch horizontally.
- **Random Sized Crop (probability 1.0):** We make a squared crop of the sketch in a random position, with the size of the side being also random between 0.3 and 1.0 times the size of the original sketch.

3.2. Semi-supervised approaches

3.2.1 M2 Semi-supervised model

The proposed architecture is shown in Figure 3, it utilizes an AlexNet backbone for both the encoder and the classifier, and an inverted AlexNet model for the decoder. Here, we choose AlexNet over ResNet50 because the last showed signs of underfitting in this scenario. The latent space consists of the concatenation of μ (32D) with the classification vector (128D). We represent sketches with value 0 for strokes, and value 1 for background, thus we add a sigmoid function at the end.

To train our model we use a traditional generative loss L, identically as in VAE. However, in this semi-supervised context, we take advantage of the two worlds, the labeled and unlabeled data, thus we propose two losses, the labeled loss $l\mathcal{L}_{M2}$ and the unlabeled one $u\mathcal{L}_{M2}$ that are defined in Equations 2 and 3.

$$l\mathcal{L}_{M2} = \mathcal{L} + 0.1N \cdot CE(y_{true}, y_{pred}) \tag{2}$$

$$u\mathcal{L}_{M2} = \sum y_{pred} \cdot \mathcal{L} + \mathcal{H}(y_pred) \tag{3}$$

where N is the length of the training dataset, and $\mathcal{H}$ is the entropy of predictions. In addition, for $u\mathcal{L}_{M2}$, $\mathcal{L}$ is weighted by the confidence of the predictions.

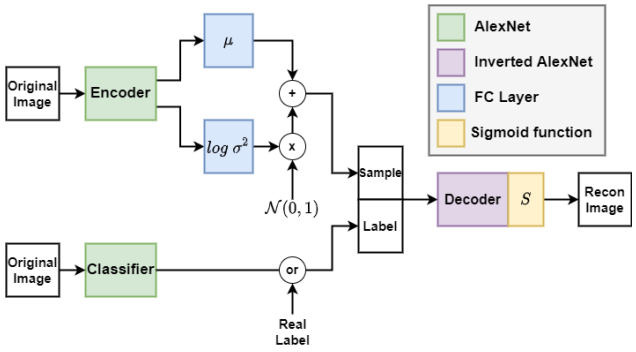

Figure 3. Proposed architecture for M2. Both the encoder and the classifier are AlexNet models, the vectors from the encoder are extracted with fully-connected layers. The decoder is an Inverted AlexNet, it receives the output of the encoder and the classifier to reconstruct the sketch, it also has a sigmoid activation function due to the binary values of a sketch.

3.2.2 Semi-supervised Variational Autoencoder

The proposed architecture, shown in Figure 4, utilizes an AlexNet backbone for the encoder, an inverted AlexNet model for the decoder, for the same reasons given for the M2 model, and a single fully-connected layer with a softmax activation function as the classifier. We choose a latent space of 32 dimensions.

Unlike the M2 model, the output of the classifier is not used as part of the feature vector, we only use μ. Like with previous VAE models, we used sketches with dimensions of 256×256, and binary representations for sketches. We use a sigmoid activation layer in the output of the decoder, binary cross-entropy as a reconstruction loss, and cross-entropy as the classification loss. We used a β weight for the KLD loss of 0.1 and an α weight for the classifier loss of 0.1[2].

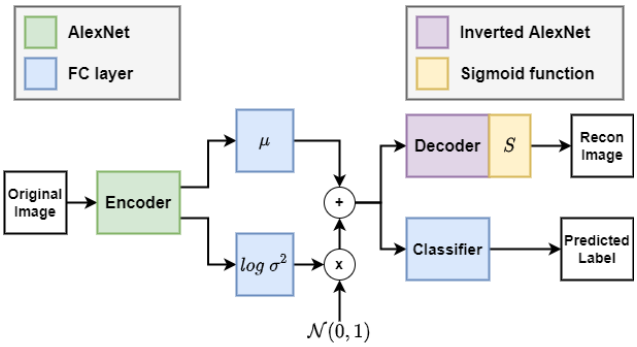

Figure 4. Proposed architecture for the semi-supervised VAE. The encoder consists of an AlexNet model and the latent space vectors are extracted with fully-connected layers. The decoder is an Inverted AlexNet with a sigmoid activation function due to the binary values of a sketch, while the classifier consists of a single fully-connected layer with a softmax activation.

4. Experiments

4.1. Datasets

For the experiments, we use sketches from the *The Quick, Draw! dataset*[3], a collection of 50 million drawings with 345 classes. To evaluate the effect of using unlabeled data in our models, we define four training configurations with different percentages of labeled data, as shown in the Table 1.

For each configuration, we randomly select 128 classes. For testing, we define two sets, each one with 100 instances per class:

- ***Known QD*** with the same 128 classes of the train data.

[2]Selected from preliminary experiments

[3]https://quickdraw.withgoogle.com/data

Name	% labels	#classes	samples/class
Unlabeled QD	0%	128	1000
10% labeled QD	10%	128	1000
50% labeled QD	50%	128	1000
Labeled QD	100%	128	1000

Table 1. Training datasets built from *The Quick, Draw! Dataset*.

- ***Unknown QD*** with other 128 classes not contained in the train dataset.

4.2. Evaluation protocol

Both VAE and BYOL are trained with the *Unlabeled QD training dataset*. For M2 and semi-supervised VAE models, we use the *10% and 50% labeled QD training datasets*. Then, we evaluate the trained models on our test datasets. We evaluate sketch retrieval with the **accuracy** of a kNN classifier ($k = 5$), and **mAP@5**. The first metric measures how the classes are distributed in the generated latent space, while the second measures how relevant are the sketches retrieved for each query. The latent spaces of VAE and semi-supervised VAE have 32 dimensions. For the M2 model, the latent space is defined by the output of the encoder of size 32 and the output of the classifier of size 128, with 160 dimensions total. Finally, for *sketch-BYOL*, we use the output of a ResNet 50 as feature vectors (2048D).

4.3. Quantitative experiments

Model	Accuracy	mAP@5	Type
VAE	0.338	0.307	self
M2 10% labeled	0.528	0.492	semi
M2 50% labeled	0.663	0.624	semi
SSVAE 10% labeled	0.490	0.449	semi
SSVAE 50% labeled	0.672	0.648	semi
sketch-BYOL	0.634	0.597	self
ResNet	**0.687**	**0.655**	supervised

Table 2. Comparisons on Known QD testing dataset, which contains the same classes as the training datasets. Best results are highlighted in bold. Accuracy is computed from a kNN classifier with k=5.

Table 2 shows the results with known categories. Supervised ResNet outperforms other baselines, and in contrast, VAE obtains the worst results having about half the performance in both metrics. In the semi-supervised scenario, adding a percentage of labels in the training datasets improves the results. We observed adding 50% of the data makes the models competitive. On the other hand, *sketch-BYOL* achieves competitive performance, being only five points below the Resnet in both metrics. It is important to highlight *sketch-BYOL* among the top performers without using any labels and being a self-supervised approach.

BYOL can be suitable for real-world datasets with no annotations.

Model	Accuracy	mAP@5	Type
VAE	0.330	0.310	self
M2 10% labeled	0.291	0.267	semi
M2 50% labeled	0.318	0.298	semi
SSVAE 10% labeled	0.403	0.358	semi
SSVAE 50% labeled	0.422	0.390	semi
sketch-BYOL	**0.627**	**0.590**	self
ResNet	0.575	0.528	supervised

Table 3. Comparison on Unknown QD testing dataset, which contains different classes than the training datasets. Best results are highlighted in bold. Accuracy is computed from a kNN classifier with k=5.

When we evaluate with unseen categories (Table 3), all supervised and semi-supervised models decrease their performance metrics. ResNet drops around 10 points in both metrics, while the semi-supervised models suffer an even greater loss, getting closer to VAE. Both self-supervised models don't show a big change in performance, and while VAE keeps being far from competitive, *sketch-BYOL* becomes the best performing model in both metrics.

4.4. Qualitative experiments

To understand the behavior of the inferred feature spaces, we visualize the latent space of the studied models together with the class distribution. To this end, we project the real space to 2D by the t-SNE approach [25]. We use a subset of 8 classes, randomly selected, from each evaluation dataset, and observe the differences. For the semi-supervised models, we only show results with 50% of labeled data.

In Figure 5, we observe the distribution of known categories in the latent space of each model. First, VAE learns some clusters with some overlapping categories. For example, the hospital category overlaps with helmet, harp, and, camouflage categories. It also seems to occupy the space uniformly, which seems to be the effect of the KLD loss. With the M2 model, we observe well-defined class clusters, but with outliers of all classes in the middle, this might be the effect of concatenating two different vectors to form its latent space. Both the semi-supervised VAE and *sketch-BYOL* produce class clusters with very little overlapping, and only the camouflage category seems harder to classify. Interestingly, *sketch-BYOL* learns differences within some classes. Lightning, alarm clock, and carrot categories show two groups each.

When we repeat the protocol evaluation with unknown categories (see Figure 6), VAE presents similar properties as before, forming category clusters with overlap categories, and instances distributed uniformly in the latent space. On the other hand, both semi-supervised models decrease the quality of their latent spaces. For M2, instead of having one cluster per category, it shows multiple smaller clusters with some overlap between them, while the semi-supervised VAE has a latent space similar to VAE However, *sketch-BYOL* is the only model that still achieves well-defined category clusters, which is consistent with the results from Tables 2 and 3.

We also show results for sketch retrieval in Figures 7 and 8 with known categories. In the first figure, for a snorkel query, VAE and semi-supervised VAE are only able to retrieve a few sketches of the correct category, while M2 and *sketch-BYOL* have no trouble with this query preserving fine-grained details such as contours and its breathing tube. In the second figure, for a car query, VAE clearly has the worst performance finding no relevant sketches such as clouds and trains. The remaining models retrieve car sketches with different forms and wheels styles.

When we evaluate with unknown categories, as shown in Figures 9 and 10, we see that all VAE-based models have trouble with the queries. In the first figure, for a television query, *sketch-BYOL* finds relevant television results preserving square shape, and TV antenna, while the other models find sketches with very little relevance, confusing with kitchen, drawers, and traffic lights. In the second figure, for a flower query, *sketch-BYOL* still is the best performer, except this time both VAE and semi-supervised VAE retrieve a few relevant flowers.

5. Application

Our sketch-BYOL can be employed to find hand-drawn sketches associated with edgemaps. In this manner, it would possible to produce sketch-image pairs required in tasks like sketch-based image retrieval, sketch-based localization, and sketch2photo translation. The main advantage is to create more realistic and human-based sketches minimizing human participation, thus allowing self-supervision. Selecting the top k results, we can generate sketch variability to capture different human interpretations and can be naturally used for coarse-grained sketch-based image retrieval.

Figure 11 shows the retrieval performance of our model in the QuickDraw dataset when the query is an edge-map produced by *PiDiNet* [22]. We can regard this strategy as sketch-based data augmentation.

6. Conclusions

In this work, we study how to handle unlabeled data to improve sketch-based understanding in terms of retrieval performance. Our sketch-BYOL approach yields competitive results on queries from known categories and is the best performer for queries of unknown categories, showing a better generalization. We also analyze our approach via

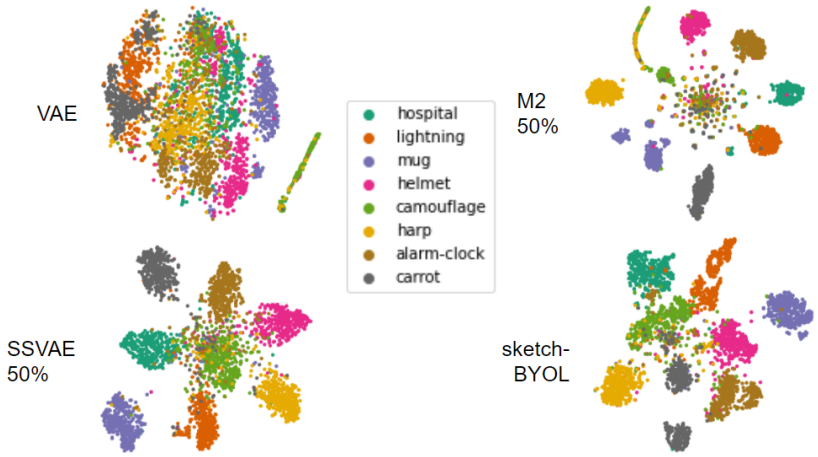

Figure 5. t-SNE visualization for known categories. Most baselines shows clear group boundaries, with exception of VAE.

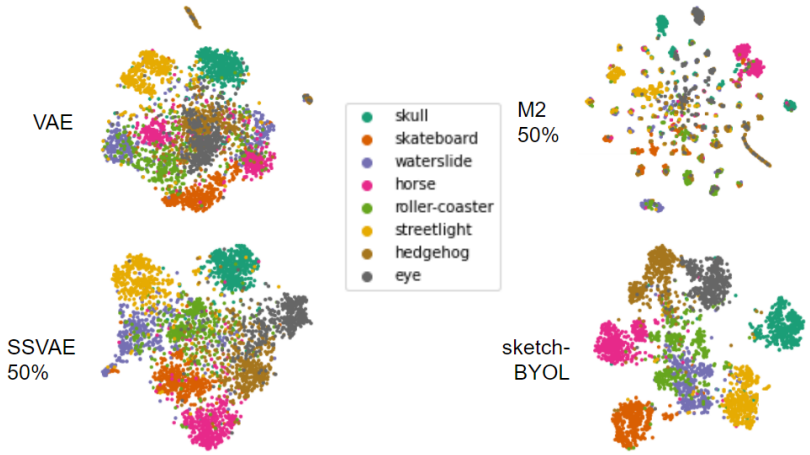

Figure 6. t-SNE visualization for unknown categories. Sketch-BYOL show clear group boundaries as opposed to other baselines.

Figure 7. Sample snorkel sketch-retrieval results from known categories.

embeddings visualizations through t-SNE projections. Finally, we show that our proposal is highly relevant in applications like sketch2photo translation or sketch-based image retrieval, where making sketch-image pairs is required.

In future work, we plan to extend sketch-BYOL to manage sketch-based image retrieval via its two branches (bimodal-BYOL). The former will encapsulate sketches, and the latter will encapsulate images. Then, commonalities in these two modalities are extracted via the loss function.

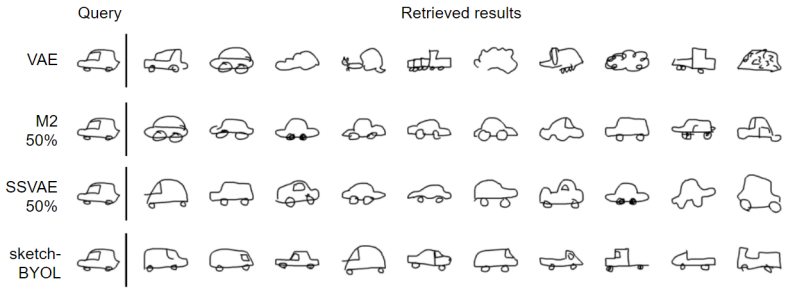

Figure 8. Sample car sketch-retrieval results from known categories.

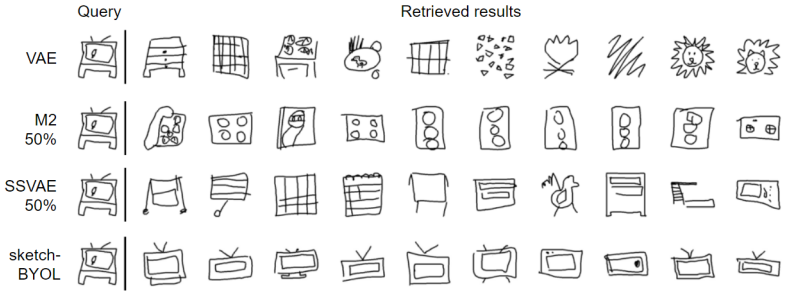

Figure 9. Sample television sketch-retrieval results from unknown categories.

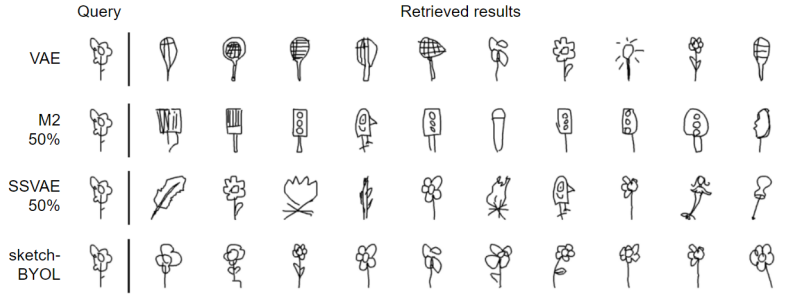

Figure 10. Sample flower sketch-retrieval results from unknown categories.

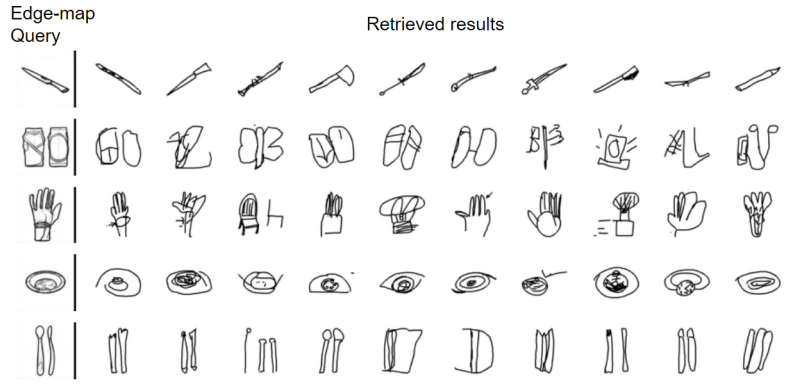

Figure 11. Finding sketches from an image. We convert an image to an edge map using *PiDiNet* [22], and sketch-BYOL retrieves the most similar sketches.

References

[1] *An Unsupervised Deep Learning Model to Discover Visual Similarity Between Sketches for Visual Analogy Support*, volume Volume 8: 32nd International Conference on Design Theory and Methodology (DTM) of *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 08 2020. V008T08A003. 2

[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, 01 2018. 1, 2

[3] Wengling Chen and James Hays. Sketchygan: Towards diverse and realistic sketch to image synthesis. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9416–9425, 2018. 1, 3

[4] John Collomosse, G McNeill, and Y Qian. Storyboard sketches for content based video retrieval. pages 245 – 252, 2009. 1, 3

[5] Mathias Eitz, James Hays, and Marc Alexa. How do humans sketch objects? *ACM Trans. Graph. (Proc. SIGGRAPH)*, 31(4):44:1–44:10, 2012. 1, 2

[6] Kenneth Forbus, Jeffrey Usher, Andrew Lovett, Kate Lockwood, and Jon Wetzel. Cogsketch: Sketch understanding for cognitive science research and for education. *Topics in Cognitive Science*, 3(4):648–666, 2011. 1

[7] Anibal Fuentes and Jose M. Saavedra. Sketch-qnet: A quadruplet convnet for color sketch-based image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2021, virtual, June 19-25, 2021*, pages 2134–2141. Computer Vision Foundation / IEEE, 2021. 1, 2

[8] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 21271–21284. Curran Associates, Inc., 2020. 1, 2, 3, 4

[9] David Ha and Douglas Eck. A neural representation of sketch drawings. In *International Conference on Learning Representations*, 2018. 2

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. 1, 3

[11] I. Higgins, L. Matthey, A. Pal, Christopher P. Burgess, Xavier Glorot, M. Botvinick, S. Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017. 4

[12] David H Hubel and Torsten N. Wiesel. *Brain and Visual Perception: The Story of a 25-Year Collaboration Illustrated Edition*. Oxford University Press, 2004. 1

[13] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, 2017. 3

[14] Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in neural information processing systems*, pages 3581–3589, 2014. 3

[15] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2014. 1, 3

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):84–90, May 2017. 4

[17] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. UMAP: uniform manifold approximation and projection. *J. Open Source Softw.*, 3(29):861, 2018. 2

[18] Kushin Mukherjee, Robert X. D. Hawkins, and Judith W. Fan. Communicating semantic part information in drawings. In Ashok K. Goel, Colleen M. Seifert, and Christian Freksa, editors, *Proceedings of the 41th Annual Meeting of the Cognitive Science Society, CogSci 2019: Creativity + Cognition + Computation, Montreal, Canada, July 24-27, 2019*, pages 2413–2419. 1

[19] Nils Murrugarra-Llerena and Adriana Kovashka. Image retrieval with mixed initiative and multimodal feedback. In *British Machine Vision Conference, BMVC 2018*. 1, 2

[20] Nils Murrugarra-Llerena and Adriana Kovashka. Image retrieval with mixed initiative and multimodal feedback. *Computer Vision and Image Understanding*, 207:103204, 2021. 1, 2

[21] Patsorn Sangkloy, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. Scribbler: Controlling deep image synthesis with sketch and color. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6836–6845, 2017. 1, 2, 3

[22] Zhuo Su, Wenzhe Liu, Zitong Yu, Dewen Hu, Qing Liao, Qi Tian, Matti Pietikäinen, and Li Liu. Pixel difference networks for efficient edge detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5117–5127, 2021. 6, 8

[23] Pablo Torres and Jose M. Saavedra. Compact and effective representations for sketch-based image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2021, virtual, June 19-25, 2021*, pages 2115–2123. Computer Vision Foundation / IEEE, 2021. 1, 2

[24] Aditay Tripathi, Rajath R. Dani, Anand Mishra, and Anirban Chakraborty. Sketch-guided object localization in natural images. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part VI*, volume 12351 of *Lecture Notes in Computer Science*, pages 532–547. Springer, 2020. 1, 2

[25] Laurens van der Maaten and Geoffrey E. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008. 6

[26] Dirk B. Walther, Barry Chai, Eamon Caddigan, Diane M. Beck, and Li Fei-Fei. Simple line drawings suffice for functional mri decoding of natural scene categories. *Proceedings of the National Academy of Sciences*, 108(23):9661–9666, 2011. 1

[27] Peng Xu, Timothy M. Hospedales, Qiyue Yin, Yi-Zhe Song, Tao Xiang, and Liang Wang. Deep learning for free-hand sketch: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2022. 1, 2

[28] Peng Xu, Yongye Huang, Tongtong Yuan, Kaiyue Pang, Yi-Zhe Song, Tao Xiang, Timothy M. Hospedales, Zhanyu Ma, and Jun Guo. Sketchmate: Deep hashing for million-scale human sketch retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1, 2

[29] Peng Xu, Yongye Huang, Tongtong Yuan, Tao Xiang, Timothy M. Hospedales, Yi-Zhe Song, and Liang Wang. On learning semantic representations for large-scale abstract sketches. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(9):3366–3379, 2021. 2

[30] Peng Xu, Chaitanya K Joshi, and Xavier Bresson. Multigraph transformer for free-hand sketch recognition. *IEEE Transactions on Neural Networks and Learning Systems*, 2021. 2

[31] Peng Xu, Zeyu Song, Qiyue Yin, Yi-Zhe Song, and Liang Wang. Deep self-supervised representation learning for free-hand sketch. *IEEE Transactions on Circuits and Systems for Video Technology*, 31:1503–1513, 2021. 2

[32] Qian Yu, Jifei Song, Yi-Zhe Song, Tao Xiang, and Timothy M. Hospedales. Fine-grained instance-level sketch-based image retrieval. *Int. J. Comput. Vision*, 129(2):484–500, 2021. 1, 3

[33] Qian Yu, Yongxin Yang, Feng Liu, Yi-Zhe Song, Tao Xiang, and Timothy M. Hospedales. Sketch-a-net: A deep neural network that beats humans. *Internationa Journal of Computer Vision*, 122(3):411–425, 2017. 1, 2

[34] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017. 3