

# SketchTransfer: A Challenging New Task for Exploring Detail-Invariance and the Abstractions Learned by Deep Networks

Alex Lamb  
MILA

lambalex@iro.umontreal.ca

Sherjil Ozair  
MILA

sherjilozair@gmail.com

Vikas Verma

MILA, Aalto University, Finland

vikas.verma@aalto.fi

David Ha  
Google Brain

hadavid@google.com

## Abstract

*Deep networks have achieved excellent results in perceptual tasks, yet their ability to generalize to variations not seen during training has come under increasing scrutiny. In this work we focus on their ability to have invariance towards the presence or absence of details. For example, humans are able to watch cartoons, which are missing many visual details, without being explicitly trained to do so. As another example, 3D rendering software is a relatively recent development, yet people are able to understand such rendered scenes even though they are missing details (consider a film like Toy Story). The failure of machine learning algorithms to do this indicates a significant gap in generalization between human abilities and the abilities of deep networks. We propose a dataset that will make it easier to study the detail-invariance problem concretely. We produce a concrete task for this: SketchTransfer, and we show that state-of-the-art domain transfer algorithms still struggle with this task. The state-of-the-art technique which achieves over 95% on MNIST → SVHN transfer only achieves 59% accuracy on the SketchTransfer task, which is much better than random (11% accuracy) but falls short of the 87% accuracy of a classifier trained directly on labeled sketches. This indicates that this task is approachable with today’s best methods but has substantial room for improvement.*

## 1. Introduction

Humans experience a high resolution world which is rich in details, yet are able to discern small amounts of highly relevant information from the world. Thus humans learn abstractions that store important pieces of information while discarding others. For example, Figure 2 shows how only a

subset of the visual details in a dollar bill are remembered by a person asked to produce a sketch without a reference.

The quality of the abstractions learned by humans may be an important factor in the flexibility and adaptability of human perception: we are able to understand objects and our environment even as details change.

Disturbingly, a growing body of evidence shows that while deep networks have visual perception which is competitive with humans (at least in some cases), the abstractions learned by deep networks differ substantially from what humans learn.

An exploration by [39] showed that deep networks correctly classify objects when only high-frequency components of the image are preserved but fail completely when only low-frequency components are preserved (the low frequency version of image is a blurred version of that image). This is the opposite of the inductive bias demonstrated by humans.

The adversarial examples literature [35] has found that neural network’s predictions can be changed by extremely small (adversarially selected) perturbations which are imperceptible to human vision. At the same time these small perturbations have either a very small effect on humans (when given very little time to process an image) [8] or no effect at all.

The BagNet project [3] explored using bags of local features for Imagenet classification and found that such models could achieve competitive results. Additionally, they found that if one performed style transfer on real images [13] and retained the textures while scrambling the image’s content, standard convolutional Imagenet classifiers retained strong performance. This is evidence that convnets trained on datasets similar to Imagenet may learn primarily from local textures while discarding global structure, indicating a strong lack of detail invariance. On artificial images with the local texture from one image but the global

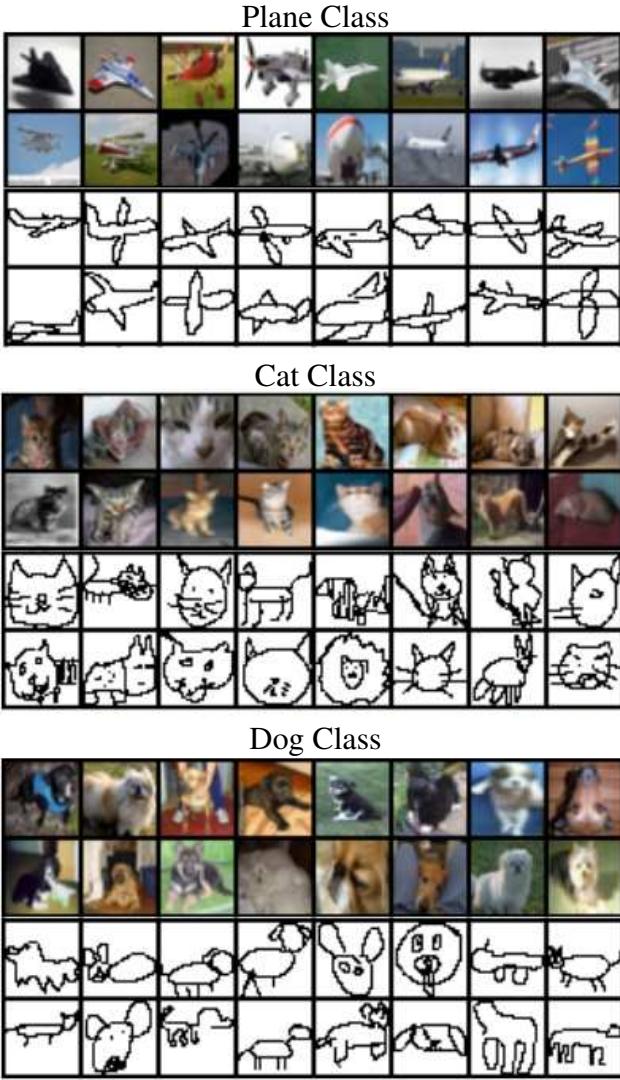


Figure 1. The SketchTransfer task involves recognizing sketches while only having access to labeled examples of real images and unlabeled sketch images. This is a challenging task for today’s state-of-the-art transfer learning algorithms.

shape and content of another image, imangenet classifier’s nearly always prefer the class associated with the texture rather than the content [14]. Additionally this can be partially addressed and generalization improved by training on artificially stylized examples in the imangenet dataset.

We create a dataset, which we call SketchTransfer, where a neural network is evaluated on the quality of its abstractions learned without explicit supervision. More specifically, we constructed a dataset in which a model must be able to classify images of sketches while only having access to labels for real images. As an example of why this task is difficult, consider sketches of dogs and cats. Many of these sketches choose to focus on the face of the animal. Oftentimes the only clear difference between a dog sketch

and a cat sketch is the shape of the ears, with dogs having round ears and cats having pointed ears. While the shape of the ears is a feature which is present in real images of cats and dogs, a neural network may not pick up on this highly salient feature.

While ideally a network could perform this task with just real data and without any access to data points with missing details, this may make the task too difficult. To make the task easier, we give the network access to unlabeled sketches with the long-term goal of creating networks that can perform well with as little sketch data as possible or perhaps no sketch data at all (in practice we found the task is challenging even with a substantial amount of sketch data).

An emerging view in machine learning is that a great deal of learning is accomplished in a “self-supervised way” by relying primarily on the structure in unlabeled natural data as a form of supervision. A successful algorithm on the SketchTransfer dataset could provide evidence that detail invariance is also achievable without the use of explicit supervision of data with missing details.

Our new dataset, which we call SketchTransfer, provides the following contributions:

- A new dataset for transfer learning, which is approachable with today’s state of the art methods but is still very challenging.
- A demonstration that even well-regularized algorithms fail to generalize across different levels of detail.
- A brief exposition of a few state-of-the-art approaches to transfer learning and experiments with them on the SketchTransfer dataset.

## 2. SketchTransfer

The SketchTransfer training dataset consists of two parts: labeled real images and unlabeled sketch images. The test dataset consists of labeled sketch images. To make the task as straightforward as possible, we used the already popular CIFAR-10 dataset as the source of labeled real images.

We used the quickdraw dataset [23] as the source of sketch images. This dataset consists of 345 classes and was collected by asking volunteers to quickly sketch a given class with a 20 second time limit. For the SketchTransfer dataset, we rendered the QuickDraw images at a fixed resolution of 32x32.

The quickdraw dataset contains many more classes than CIFAR-10 and the classes are slightly different and generally more detailed. To solve this we defined a correspondence between the classes in CIFAR-10 and a relevant subset of the quickdraw classes (shown in Table 1). For example we map the quickdraw classes “Car” and “Police Car” onto the CIFAR-10 class automobile.



Figure 2. On the left, a person’s sketch of a dollar bill from memory. On the right, the same person’s sketch with access to a reference. This indicates that humans only remember and can perform recognition with only a small handful of salient aspects of data and have substantial detail invariance. (Image credit: *The empty brain* [9])

One small issue is that CIFAR-10 contains a deer class and QuickDraw doesn’t. Thus we elected to make our SketchTransfer test set only have 9 classes, but for the convenience of researchers we keep the CIFAR-10 training set the same (having 10 classes). Our final dataset of sketches consists of 9 classes which correspond directly to CIFAR-10 classes. This sketch dataset has a total of 90000 training images and 22500 test images (10000 and 2500 per class respectively).

Examples from both real and sketch images across all the classes can be seen in Figure 1 and Figure 3. One important property of SketchTransfer is that aside from sharing common classes, there is no explicit pairing between the real images and the sketch images.

We received permission from the Google Creative Lab to use the Quickdraw dataset [23].

### 3. Related Datasets

Datasets of human sketches make it possible for algorithms to learn representations closely aligned with human priors. Prior to QuickDraw [23], the Sketch dataset [7] of 20K hand sketches was used to explore feature extraction techniques. A later work, the Sketchy dataset [33], provided 70K vector sketches paired with corresponding photo images for various classes to facilitate a larger-scale exploration of human sketches. ShadowDraw [25] used a dataset of 30K raster images combined with extracted vectorized features to construct an interactive system that predicts what a finished drawing looks like based on a set of incomplete brush strokes from the user’s digital canvas. In addition to human sketches, ancient Eastern scripts also have sketch-like properties. For example, [4] considered using an RNN and VAEs to generate sketched Japanese characters. Understanding and recognizing these historical Japanese characters has become an important and widely studied problem in the digital humanities field [5, 6].

In developmental childhood psychology, [27, 26] per-

formed a series of experiments in which they tasked children of various ages with drawing sketches of certain classes, and then studying the properties of the sketches using convolutional neural networks. Their main finding was that visual features become more distinctive in the drawings produced by older children and that this goes beyond differences in visuomotor controls. This provides some evidence that for humans the deeper knowledge gained about objects from growing older, is reflected in sketches.

Table 1. Class correspondences used to construct the sketch part of the SketchTransfer dataset.

| Cifar-10 Class | QuickDraw Classes                         |
|----------------|---|
| Airplane       | Airplane                                  |
| Automobile     | Car<br>Police Car                         |
| Bird           | Bird<br>Duck<br>Flamingo<br>Owl<br>Parrot |
| Cat            | Penguin<br>Swan<br>Cat<br>Lion<br>Tiger   |
| Deer           | n/a                                       |
| Dog            | Dog                                       |
| Frog           | Frog                                      |
| Horse          | Horse                                     |
| Ship           | Cruise Ship<br>Sailboat<br>Speedboat      |
| Truck          | Truck<br>Firetruck                        |



Figure 3. Examples from all of the classes in the SketchTransfer dataset, with real images from the class and sketch images from the class.

## 4. Baselines

We evaluated several techniques on the proposed Sketch-Transfer dataset. First we considered techniques which only use the labeled source data (real images) including the use of regularization. As these models don't see the sketches during training, we might expect that it would be difficult for these models to generalize well on sketches. For techniques which take advantage of the unlabeled data, we primarily focus on methods which perform well on existing benchmarks. One such benchmark is SVHN to MNIST, which involves transfer learning between very detailed color images of digits (SVHN) and plain black-and-white digit images (MNIST) [28]. One key idea is to try to make the model's representations on source datapoints and target domain datapoints more similar, which we discuss in Section 4.3. Another key idea is to enforce a consistency based objective on the target domain, which is also widely used in semi-supervised learning [37]. The idea with this is to discourage the classifier's decision boundary from passing through regions where the data in the target domain has high density (and thus encourage connected regions of high density to share the same predicted label). Virtual adversarial training (Section 4.4) and virtual mixup training (Section 4.6) are both motivated by this notion of consistency.

### 4.1. Training only on Labeled Source Data

As the simplest baseline, we consider simply training a network on the CIFAR-10 dataset and evaluating on sketches.

### 4.2. Regularized Training on Source Data

One possible approach is to still only use the real labeled images, but use regularization. A possibility is that this will force the model to use the more salient features in the data to predict on real images which will then be usable when predicting on sketches. For example, Mixup [41], which is a state-of-the-art regularizer for deep networks, uses the linear interpolations of images and the corresponding interpolation of the labels for training (Equation 1).

$$\begin{aligned}\tilde{x} &= \lambda x_i + (1 - \lambda)x_j, \\ \tilde{y} &= \lambda y_i + (1 - \lambda)y_j.\end{aligned}\quad (1)$$

Furthermore, [36] found that the use of Manifold Mixup improves robustness to several classes of artificial image distortions. In its original formulation, Mixup [41] is only applicable to training with labeled data, although semi-supervised [37] and unsupervised variants [1] has also been explored lately.

### 4.3. Adversarial Domain Adaptation

Domain adversarial training [12] consists of taking the hidden representations learned by a network on a source do-

main and a target domain and using an adversarially trained discriminator  $d$  to make the representations  $g(x)$  follow the same overall distribution. Since the discriminator is trained on hidden states (for both source and target domains), it does not require labeled data from the target domain.

$$\begin{aligned} \min_f \mathcal{L}_y(f; \mathcal{X}_s, \mathcal{Y}_s) + \lambda_d \mathcal{L}_d(g; \mathcal{X}_s, \mathcal{X}_t) \\ \mathcal{L}_d(g; \mathcal{X}_s, \mathcal{X}_t) = \sup_d \mathbb{E}_{x \sim \mathcal{X}_s} [\ln d(g(x))] \\ + \mathbb{E}_{x \sim \mathcal{X}_t} [\ln(1 - d(g(x)))] \end{aligned} \quad (2)$$

The adversarial domain adaptation objective (Equation 2) augments the usual classifier objective on source data  $\mathcal{X}_s$  and a hyperparameter  $\lambda_d$  is used to adjust the weight of the adversarial loss, which aims to make features  $g(x)$  look similar for examples from the source and target distributions.

#### 4.4. Virtual Adversarial Training

The virtual adversarial training (VAT) [30] algorithm is based on the Consistency regularization principle [32, 24]. The central idea of this algorithm is to use predicted-labels of unlabeled samples to generate adversarial perturbations, and use these adversarial perturbations instead of random perturbations for Consistency regularization. We always used the FGSM attack for computing the virtual adversarial perturbations [15].

$$\mathcal{L}_v(f; \mathcal{X}) = \mathbb{E}_{x \sim \mathcal{X}} \left[ \max_{\|r\| \leq \epsilon} D_{\text{KL}}(f(x) \| f(x + r)) \right]. \quad (3)$$

Thus given some ball with a radius  $\|r\| \leq \epsilon$  around each of the data points we encourage the change in the model's prediction  $f$  to be small (Equation 3).

#### 4.5. Dirt-T

The Decision Boundary Iterative Refinement Training with a Teacher (or "Dirt-T") algorithm [34] starts with a pre-trained model and considers an iterative refinement procedure to enforce the cluster assumption on the unlabeled target data (the cluster assumption is the notion that decision boundaries are more likely to pass through regions of low-density). Because this is only done on the target domain, it leads to high loss on the source domain, but also allows for the possibility of distribution-shift between the source and target domains.

#### 4.6. Virtual Mixup Training (VMT)

The approaches which combine *consistency regularization* with Mixup training [42, 36, 38] have shown to achieve state-of-the-art results in semi-supervised learning

paradigm [37, 2]. VMT [28] extends these approaches in the paradigm of Unsupervised Domain Adaptation by combining Mixup training with the Domain Adversarial Training [12]. More formally, VMT [28] augments the loss function of Domain Adversarial Training and the entropy minimization loss with the Mixup loss.

In the case of target domain, since the targets are not available, similar to the [37, 2, 30], the pseudolabels of the unlabeled samples are used for mixing. The mixing of samples and pseudolabels are done as follows (where  $\lambda \sim \text{Beta}(\alpha, \alpha)$ ):

$$\begin{aligned} \tilde{x} &= \lambda x_i + (1 - \lambda)x_j, \\ \tilde{y} &= \lambda f(x_i) + (1 - \lambda)f(x_j), \end{aligned} \quad (4)$$

The Mixup loss function on pseudolabels (Equation 4), which can be augmented with other losses, is given as follows:

$$\mathcal{L}_m(f; \mathcal{X}) = \mathbb{E}_{x \sim \mathcal{X}} [D_{\text{KL}}(\tilde{y} \| f(\tilde{x}))]. \quad (5)$$

#### 4.7. Rotation Prediction

We can also consider adding self-supervised objectives, with the intuition that solving this objective may require the model to have a more thorough understanding of salient aspects than is required to predict the class. One such self-supervised objective that we consider, due to its simplicity, is randomly rotating input images by either 0 degrees, 90 degrees, 180 degrees, or 270 degrees, and having the model classify the degree of rotation as a 4-way classification task [10]. As this objective does not require labels, we can apply it on both the real images and the sketch images.

#### 4.8. CyCada and CycleGAN

The CyCada project [20] explored the use of the CycleGAN algorithm [44] for transfer learning to new domains. Essentially the CycleGAN consists of an encoder which maps from the source domain to the target domain and a decoder which maps back to the source domain. A discriminator is used to encourage the encoded domain to follow the same distribution as the target domain. Notably this can be done without having access to paired examples from the source and target domains.

### 5. Experiments

First we consider running all the baselines discussed in Section 4 on the full SketchTransfer dataset, with 10000 unlabeled images per class (or zero per class, for the baselines which don't use the unlabeled sketches). All numbers are accuracies. We ran two identical trials of each experiment with different random seeds and report the mean and standard deviation. The results of running with these baseline algorithms is presented in Table 2.

We note that our best result not using labeled sketch data is the same combination of techniques which achieved the best result in Virtual Mixup Training [28]. However on their tasks involving MNIST to SVHN and SVHN to MNIST transfer, this approach was able to almost match the performance of a classifier trained on labeled examples from the target domain. However on the SketchTransfer task there is still a clear gap between the 59% obtained with only unlabeled sketches and the 87% obtained by training on labeled sketches.

## 5.1. Analysis

While we demonstrated in Table 2 that a model trained only on real images performs better than average on sketches, and this performance is improved by the use of transfer learning with sketches, it gives little direct insight into the nature of this improvement. To address this we considered two different analysis on the learned models. To make this analysis as clean as possible, we only considered our best model trained only on real images, our best model trained on unlabeled sketches, and our best model trained on label sketches. The best model was selected according to the test accuracies in Table 2. All of these analysis were performed on the test set.

First we considered a confusion matrix analysis, given in Figure 4. The confusion matrix shows the distribution over predicted classes for each ground truth class. A stronger diagonal indicates higher accuracy. We can see that a model trained only on real images is most accurate on planes, trucks, and ships. The confusion matrix for the model trained with unlabeled sketches shows dramatically better classification on the bird, cat, dog, frog, and horse classes.

Additionally, we conducted an analysis where we collected the k-nearest neighbors for test images using different distance metrics (Figure 5). First, we considered a simple euclidean distance in pixel space, and this yielded extremely poor nearest-neighbors, with very low semantic and class similarity. Next we used different encoders and ran it until reaching an 8x8 spatial representation and used simple euclidean distance in this space. With a purely random encoder, this yielded poor results. However, with networks trained on real images, the quality of the nearest neighbors was substantially improved in terms of semantic similarity. We also observed that the network trained using transfer learning on unlabeled sketches had even more semantically relevant nearest neighbors.

## 6. Discussion

In this work, we propose the problem of whether a machine learning algorithm can develop representations of everyday things that can be mapped to human doodles, but giving the learning algorithm access to unlabelled doodle

data. We could also consider an alternative approach where we train an artificial agent to sequentially draw, one stroke at a time, a given pixel image (such as CIFAR-10 samples) and by constraining the number of brush strokes our agent is allowed to make, we may be able to similar the biological constraints of the human anatomy and thus also develop a human doodle-like representation of actual images.

This approach has been explored in *Artist Agent* [40] and *SPIRAL* [11] where an agent is trained to paint an image from a data distribution using a paint simulator. Subsequent works [43, 21] combined the *SPIRAL* algorithm with the sketch-rnn [17] algorithm and enhanced the drawing agent by giving it an internal *world model* [18] to demonstrate that the agent can learn to paint inside its imagination. While the aforementioned works train an agent with an objective function to produce doodles that are as photorealistic as possible, a recent work [31] trains an agent to optimize instead for the *content loss* defined by a neural style transfer algorithm [13], opening up the exciting possibility of training agents that can produce truly abstract versions of photos.

However, the doodle representations of everyday objects developed by humans are not merely confined to our constrained biological ability to draw objects sequentially with our hands, stroke-by-stroke, but cultural influence is also at play. The location meta-data from the original QuickDraw dataset provided interesting examples of this phenomenon. For instance, it has been observed that most Americans draw circles counterclockwise, while most people in Japan draw them clockwise [19]. Chairs drawn in different countries tend to have different orientations [22]. Snowman in hotter countries consists of two snowballs, while for colder countries consists tend to have three [29]. By giving our algorithm actual unlabelled doodle data produced by humans around the world, we give our machine learning algorithms a chance to learn the true representation of objects developed by all of humanity.

## 7. Future Work

The idea of SketchTransfer is quite general and we identify a few ways in which it could be extended:

- While CIFAR-10 does contain realistic images, it is limited in a few ways. First, the images are rather small, being only 32x32. Second, the number of labeled images is relatively small. Third, they generally only contain the object of interest and lack context, both spatial and temporal. Selecting a dataset which is stronger along any of these axis could be a useful improvement to the SketchTransfer task.
- Sketches are to a large extent an extreme case of images which lack irrelevant details. Various cartoons and shaded illustrations may be a reasonable middle-

Table 2. Performance of different baseline models on SketchTransfer. We ran two trials for each experiment and we report mean with standard deviation in parenthesis. We also report a supervised learning result for a model trained directly on the labeled sketch training set. Our results indicate that using the unlabeled sketch data improves substantially but still falls short of the a model trained directly on labeled sketches.

| Inst.<br>Norm | Domain<br>Adv. | Source<br>Mixup | Virtual<br>Mixup | Source<br>VAT | Target<br>VAT | DIRT-T | CyCada | Rotation<br>Pred. | Unlabeled<br>Sketches<br>Per Class | Test<br>Accuracy |
|---------------|----------------|-----------------|------------------|---------------|---------------|--------|--------|-------------------|------------------------------------|------------------|
| ✓             |                |                 | ✓                |               |               |        |        |                   | 10000                              | 17.55(4.95)      |
|               |                |                 |                  |               |               |        |        |                   | 0                                  | 23.29(0.37)      |
|               |                |                 |                  |               |               | ✓      |        |                   | 10000                              | 25.27(0.12)      |
|               |                |                 |                  |               |               |        | ✓      |                   | 10000                              | 27.90(0.38)      |
| ✓             | ✓              |                 |                  |               |               |        |        |                   | 10000                              | 30.31(0.19)      |
| ✓             |                | ✓               |                  |               |               |        |        |                   | 0                                  | 35.66(0.16)      |
| ✓             |                |                 |                  |               |               |        |        |                   | 0                                  | 35.84(1.56)      |
| ✓             |                |                 | ✓                |               |               |        |        |                   | 0                                  | 37.74(1.44)      |
| ✓             |                | ✓               |                  | ✓             |               |        |        |                   | 0                                  | 38.72(0.75)      |
| ✓             |                |                 |                  |               | ✓             |        |        |                   | 10000                              | 41.15(0.45)      |
| ✓             | ✓              | ✓               | ✓                | ✓             | ✓             |        |        |                   | 10000                              | 51.60(6.30)      |
| ✓             | ✓              | ✓               | ✓                | ✓             | ✓             | ✓      | ✓      |                   | 10000                              | 58.85(2.85)      |
| ✓             |                | ✓               |                  | ✓             |               |        |        |                   | Supervised                         | 87.25(0.45)      |

ground and serve as an easier analogue task to SketchTransfer.

- We elected to make the task easier by providing access to unlabeled sketch images. Another way to accomplish this might be to provide a very small number of labeled sketch images and use meta-learning to perform few-shot (or one-shot) classification of sketches. The task classification without using any sketch data during training would be challenging, yet [16] demonstrated strong generalization to changing environments by encouraging independent mechanisms as an inductive bias of the architecture.

## 8. Conclusion

Human perceptual understanding shows a great deal of invariance to details and emphasis on salient features which today’s deep networks lack. We have introduced SketchTransfer, a new dataset for studying this phenomenon concretely and quantitatively. We applied the current state-of-the-art algorithms on domain transfer to this task (along with ablations). Intriguingly we found that they only achieve 60% accuracy on the SketchTransfer task, which is between the 11% accuracy of a random classifier but falls dramatically short of the 90% classifier trained on the labeled sketch dataset. This indicates that this new SketchTransfer task could be a powerful testbed for exploring detail invariance and abstractions learned by deep networks.

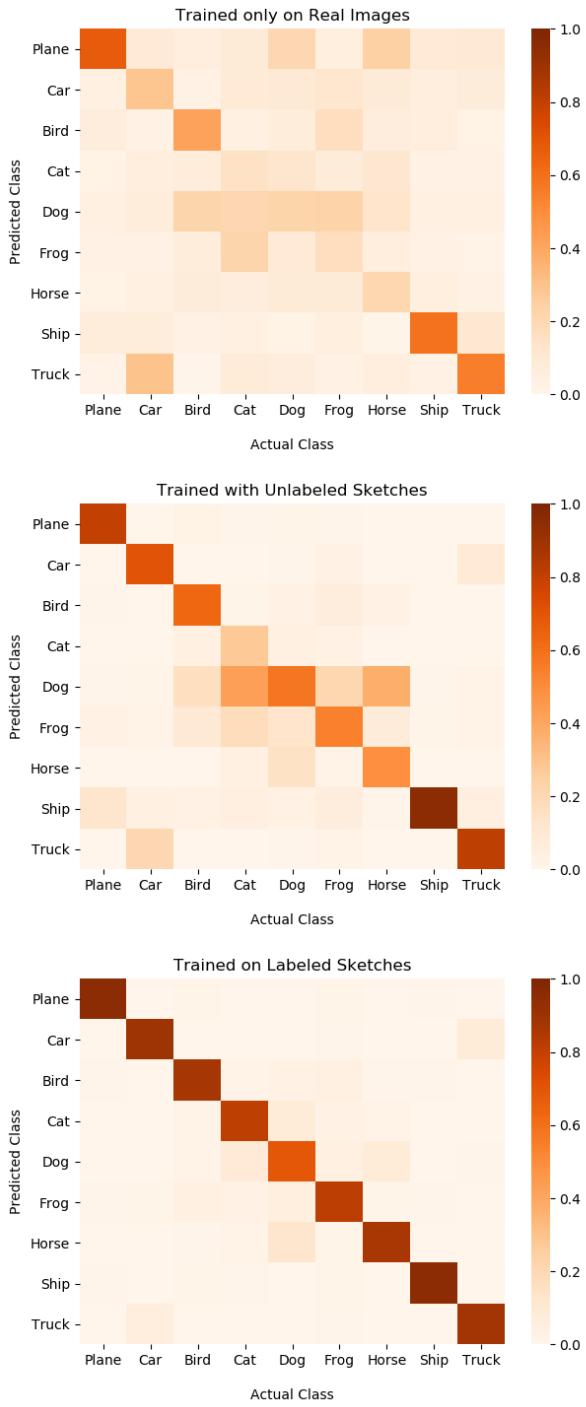


Figure 4. Confusion matrices indicating predicted and true classes for a model trained only on real images (top), our best transfer learning model (center), and a model trained on labeled sketch images (bottom). A stronger diagonal indicates better performance. The model trained only on real images has the best performance on planes and trucks. The model exploiting unlabeled images performs better on many more classes, but still struggles to separate cats and dogs as well as horses and dogs.

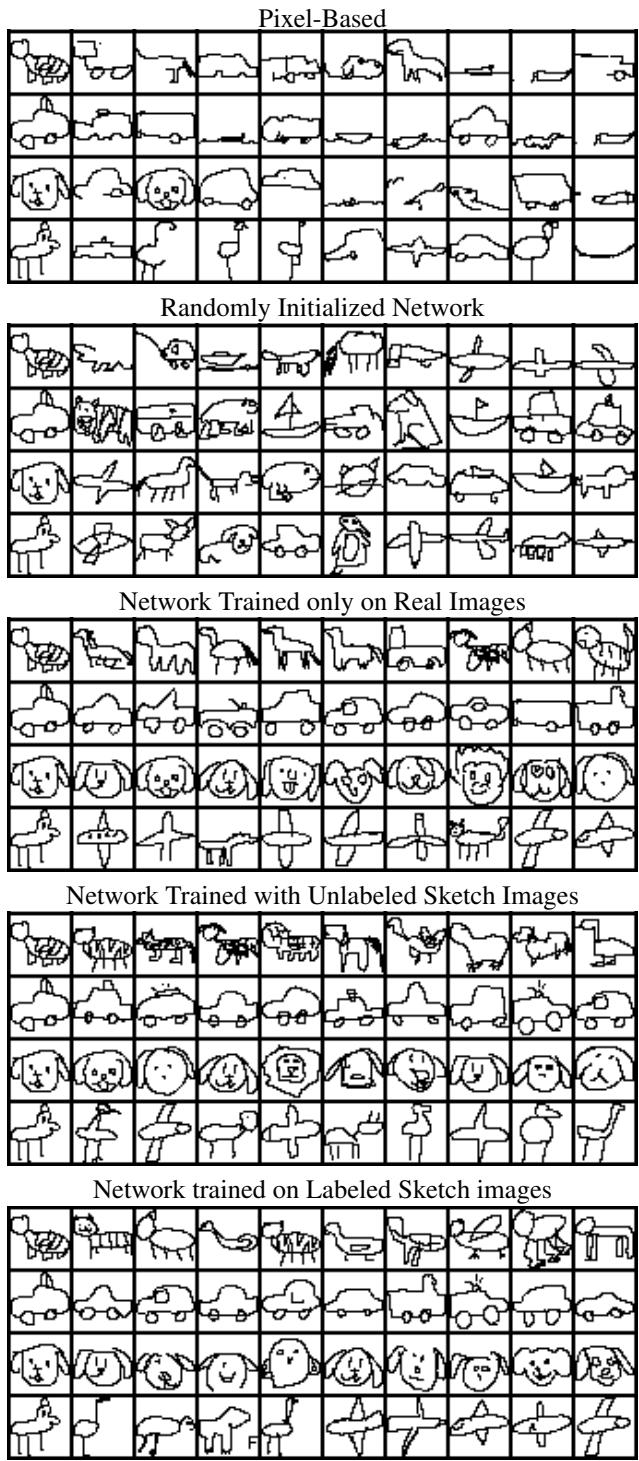


Figure 5. Nearest Neighbors on the test set using different distances (original on the left). Note that the distance learned after transfer learning is clearly more semantically meaningful, and even a well-regularized model using only the source data learned surprisingly good features for sketches.

## References

- [1] C. Beckham, S. Honari, A. Lamb, V. Verma, F. Ghadiri, R. D. Hjelm, and C. Pal. Adversarial mixup resynthesizers. *arXiv preprint arXiv:1903.02709*, 2019.
- [2] D. Berthelot, N. Carlini, I. J. Goodfellow, N. Papernot, A. Oliver, and C. Raffel. Mixmatch: A holistic approach to semi-supervised learning. *CoRR*, abs/1905.02249, 2019.
- [3] W. Brendel and M. Bethge. Approximating CNNs with bag-of-local-features models works surprisingly well on imagenet. In *International Conference on Learning Representations*, 2019.
- [4] T. Clanuwat, M. Bober-Irizar, A. Kitamoto, A. Lamb, K. Yamamoto, and D. Ha. Deep learning for classical Japanese literature. *arXiv preprint arXiv:1812.01718*, 2018.
- [5] T. Clanuwat, A. Lamb, and A. Kitamoto. End-to-end pre-modern Japanese character (kuzushiji) spotting with deep learning. *Jinmoncom 2018*, (2018):15–20, nov 2018.
- [6] T. Clanuwat, A. Lamb, and A. Kitamoto. Kuronet: Pre-modern Japanese kuzushiji character recognition with deep learning. *arXiv preprint arXiv:1910.09433*, 2019.
- [7] M. Eitz, J. Hays, and M. Alexa. How Do Humans Sketch Objects? *ACM Trans. Graph. (Proc. SIGGRAPH)*, 31(4):44:1–44:10, 2012.
- [8] G. F. Elsayed, S. Shankar, B. Cheung, N. Papernot, A. Kurakin, I. J. Goodfellow, and J. Sohl-Dickstein. Adversarial examples that fool both human and computer vision. *CoRR*, abs/1802.08195, 2018.
- [9] R. Epstein. The empty brain. *Aeon, May*, 18:2016, 2016. <https://aeon.co/essays/your-brain-does-not-process-information-and-it-is-not-a-computer>.
- [10] Z. Feng, C. Xu, and D. Tao. Self-supervised representation learning by rotation feature decoupling. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
- [11] Y. Ganin, T. Kulkarni, I. Babuschkin, S. Eslami, and O. Vinyals. Synthesizing programs for images using reinforced adversarial learning. *arXiv preprint arXiv:1804.01118*, 2018.
- [12] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.
- [13] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.
- [14] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.
- [15] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.
- [16] A. Goyal, A. Lamb, J. Hoffmann, S. Sodhani, S. Levine, Y. Bengio, and B. Schölkopf. Recurrent independent mechanisms. *arXiv preprint arXiv:1909.10893*, 2019.
- [17] D. Ha and D. Eck. A neural representation of sketch drawings. In *International Conference on Learning Representations*, 2018.
- [18] D. Ha and J. Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems 31*, pages 2451–2463. Curran Associates, Inc., 2018. <https://worldmodels.github.io>.
- [19] T.-H. Ha and N. Sonnad. How do you draw a circle? we analyzed 100,000 drawings to show how culture shapes our instincts. *Quartz*, 2017. <https://qz.com/994486/the-way-you-draw-circles-says-a-lot-about-you/>.
- [20] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. A. Efros, and T. Darrell. Cycada: Cycle-consistent adversarial domain adaptation. *arXiv preprint arXiv:1711.03213*, 2017.
- [21] Z. Huang, W. Heng, and S. Zhou. Learning to paint with model-based deep reinforcement learning. *arXiv preprint arXiv:1903.04411*, 2019.
- [22] R. Jana. Exploring and visualizing an open global dataset. *Google AI Blog*, 2017. <https://ai.googleblog.com/2017/08/exploring-and-visualizing-open-global.html>.
- [23] J. Jongejan, H. Rowley, T. Kawashima, J. Kim, and N. Fox-Gieg. The Quick, Draw!-AI Experiment. *Google AI Experiments*, 2017. <https://github.com/googlecreativelab/quickdraw-dataset>.
- [24] S. Laine and T. Aila. Temporal ensembling for semi-supervised learning. *CoRR*, abs/1610.02242, 2016.
- [25] Y. J. Lee, C. L. Zitnick, and M. F. Cohen. Shadowdraw: Real-time user guidance for freehand drawing. In *ACM SIGGRAPH 2011 Papers, SIGGRAPH '11*, pages 27:1–27:10, New York, NY, USA, 2011. ACM.
- [26] B. Long, J. Fan, Z. Chai, and M. C. Frank. Developmental changes in the ability to draw distinctive features of object categories, Jul 2019.
- [27] B. Long, J. E. Fan, and M. C. Frank. Drawings as a window into developmental changes in object representations. In *CogSci*, 2018.
- [28] X. Mao, Y. Ma, Z. Yang, Y. Chen, and Q. Li. Virtual mixup training for unsupervised domain adaptation. *arXiv preprint arXiv:1905.04215*, 2019.
- [29] M. Martino, H. Strobelt, O. Corne, and E. Phibbs. Forma fluens. <http://formafuens.io/>, 2017.
- [30] T. Miyato, S.-i. Maeda, M. Koyama, and S. Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018.
- [31] R. Nakano. Neural painters: A learned differentiable constraint for generating brushstroke paintings. *arXiv preprint arXiv:1904.08410*, 2019.
- [32] M. Sajjadi, M. Javanmardi, and T. Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems, NIPS'16*, pages 1171–1179, USA, 2016. Curran Associates Inc.

[33] P. Sangkloy, N. Burnell, C. Ham, and J. Hays. The Sketchy Database: Learning to Retrieve Badly Drawn Bunnies. *ACM Trans. Graph.*, 35(4):119:1–119:12, July 2016.

[34] R. Shu, H. H. Bui, H. Narui, and S. Ermon. A dirt-t approach to unsupervised domain adaptation. *arXiv preprint arXiv:1802.08735*, 2018.

[35] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[36] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, and Y. Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pages 6438–6447, 2019.

[37] V. Verma, A. Lamb, J. Kannala, Y. Bengio, and D. Lopez-Paz. Interpolation consistency training for semi-supervised learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI’19*, pages 3635–3641. AAAI Press, 2019.

[38] V. Verma, M. Qu, A. Lamb, Y. Bengio, J. Kannala, and J. Tang. Graphmix: Regularized training of graph neural networks for semi-supervised learning. *arXiv preprint arXiv:1909.11715*, 2019.

[39] H. Wang, X. Wu, P. Yin, and E. P. Xing. High frequency component helps explain the generalization of convolutional neural networks. *arXiv preprint arXiv:1905.13545*, 2019.

[40] N. Xie, H. Hachiya, and M. Sugiyama. Artist agent: A reinforcement learning approach to automatic stroke generation in oriental ink painting. In *ICML*. icml.cc / Omnipress, 2012.

[41] H. Zhang, M. Cissé, Y. N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. *CoRR*, abs/1710.09412, 2017.

[42] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[43] N. Zheng, Y. Jiang, and D. Huang. Strokenet: A neural painting environment. In *International Conference on Learning Representations*, 2019.

[44] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.