Adversarial Knowledge Transfer from Unlabeled Data

Akash Gupta*
University of California, Riverside
agupt013@ucr.edu

Rameswar Panda*
MIT-IBM Watson AI Lab
rpand002@ucr.edu

Sujoy Paul
University of California, Riverside
spaul003@ucr.edu

Jianming Zhang
Adobe Research
jianmzha@adobe.com

Amit K. Roy-Chowdhury
University of California, Riverside
amitrc@ece.ucr.edu

ABSTRACT

While machine learning approaches to visual recognition offer great promise, most of the existing methods rely heavily on the availability of large quantities of labeled training data. However, in the vast majority of real-world settings, manually collecting such large labeled datasets is infeasible due to the cost of labeling data or the paucity of data in a given domain. In this paper, we present a novel **A**dversarial **K**nowledge **T**ransfer (**AKT**) framework for transferring knowledge from internet-scale unlabeled data to improve the performance of a classifier on a given visual recognition task. The proposed adversarial learning framework aligns the feature space of the unlabeled source data with the labeled target data such that the target classifier can be used to predict pseudo labels on the source data. An important novel aspect of our method is that the unlabeled source data can be of different classes from those of the labeled target data, and there is no need to define a separate pretext task, unlike some existing approaches. Extensive experiments well demonstrate that models learned using our approach hold a lot of promise across a variety of visual recognition tasks on multiple standard datasets. Project page is at `https://agupt013.github.io/akt.html`.

CCS CONCEPTS

• **Computing methodologies** → **Transfer learning**; **Image representations**.

KEYWORDS

Adversarial Learning, Knowledge Transfer, Feature Alignment.

ACM Reference Format:
Akash Gupta, Rameswar Panda, Sujoy Paul, Jianming Zhang, and Amit K. Roy-Chowdhury. 2020. Adversarial Knowledge Transfer from Unlabeled Data. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413688

*Joint first authors
[1]Images source: www.pixabay.com

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10.
https://doi.org/10.1145/3394171.3413688

(a) Transfer Learning (Source: Labeled, Target: Labeled)

(b) Unsup. Domain Adaptation (Source: Labeled, Target: Unlabeled)

(c) Semi-Supervised Learning (Source: Unlabeled, Target: Labeled)

(d) AKT (Ours) (Source: Unlabeled, Target: Labeled)

Figure 1: Comparison of different learning paradigms. Our proposed approach Adversarial Knowledge Transfer (AKT) transfers knowledge from unlabeled source data to the labeled target dataset without requiring them to have the same distribution or label space. Best viewed in color.[1]

1 INTRODUCTION

Deep learning approaches have recently shown impressive performance on many visual tasks by leveraging large collections of labeled data. However, such strong performance is achieved at a cost of creating these large datasets, which typically requires a great deal of human effort to manually label samples. Recently, much progress has been made in developing a variety of ways such as transfer learning [31, 35, 49], unsupervised domain adaptation [11, 29, 30] and semi-supervised learning [1, 3, 14, 23, 32, 42, 43, 45, 45, 54] to overcome the lack of labeled data in target domain. While these approaches have shown to be effective in many tasks, they often depend on existence of large scale annotated data to train a source model [50] (in case of transfer learning: Fig. 1a) or assume that

unlabeled data comes from a similar distribution with some domain gap (e.g., across animated and real images in case of unsupervised domain adaptation: Fig. 1b) or have the same data and label distribution as of the labeled data (in case of semi-supervised learning: Fig. 1c). On the other hand, self-taught learning or self-supervised learning [6, 17, 27, 34, 40, 47] methods can transfer knowledge to the target task using unlabeled source data, which may not have the same distribution and the same label space as that of the target task. In particular, unlabeled data is first used to learn feature representation using a pretext task and then the learned feature is adapted to the target labeled dataset through finetuning. However, despite their reasonable performance on common benchmarks, it is still unclear how to design efficient pretext tasks for specific downstream tasks. Defining a pretext task is a challenging problem on its own merit [19].

In this paper, we propose a novel **A**dversarial **K**nowledge **T**ransfer (**AKT**) framework for transferring knowledge from large-scale unlabeled data *without the need to define pretext tasks*. Our approach transfers knowledge from unlabeled source data to the labeled target data without requiring them to have the same distribution or label space (see Fig. 1d). The proposed adversarial learning helps to align the feature space of unlabeled source data with labeled target data such that the target classifier can be used to predict pseudo-labels on source data. Using the pseudo labels of source data, we jointly optimize the classifier with the labeled target dataset. Unlike other self-supervised methods [34, 52], we do not employ two stage training where a model is first trained on a pretext task and then finetuned on target task. Instead, our method operates just like a solver, which updates the model with labeled and unlabeled data simultaneously, making it highly efficient and convenient to use.

Our approach is motivated by the observation that many randomly downloaded unlabeled images (e.g., web images from other object classes–which are much easier to obtain than images specifically of the target classes) contain basic visual patterns (such as edges, corners) that are similar to those in labeled images of the target dataset. Thus, we can transfer these visual patterns from the internet-scale unlabeled data for learning an efficient classifier on the target task. Note that the source and target domains in our approach may have some relationship, but we do not require them to have the same distribution or label space while transferring knowledge from the unlabeled data.

1.1 Approach Overview

An overview of our approach is illustrated in Figure 2. Given a small labeled target data and large-scale unlabeled source data, our objective is to boost the performance of the target classifier by exploiting relevant information from unlabeled samples as compared to using only the labeled target data. To this end, we adopt an adversarial approach to train the target classifier that consists of three modules: the classifier network **M**, the pseudo label generator **G** and the discriminators, $\mathbf{D}_I$ for instance-level feature alignment and $\mathbf{D}_G$ for group-level feature alignment. Note that learning the classifier **M** is our main goal.

In our approach, to train the target classifier, (1) we first input the target samples to the classifier network **M** to extract the features. (2) We then input the source samples to the pseudo label generator **G** to extract their features and pseudo-labels. (3) We use features of the target and source samples as input to both the discriminators $\mathbf{D}_I$ and $\mathbf{D}_G$ to distinguish whether the instance-level and group-level input features are from the source or target domain respectively. While instance-level discriminator tries to align individual samples from source domain to the target domain, the group level discriminator aligns one batch of source samples with one batch of target samples by considering feature means. With these two adversarial losses, the network propagates gradients from $\mathbf{D}_I$ and $\mathbf{D}_G$ to **G**, which encourages generation of similar feature distributions of source samples in the target domain. (4) Finally, we pass target samples and source samples to the classifier **M** to compute the classification loss on target samples with their labels and source samples with the pseudo labels generated from the pseudo-label generator **G**.

1.2 Contributions

To summarize, we address a novel and practical problem in this paper - how to leverage information contained in the unlabeled data that does not follow the same class labels or generative distribution as the labeled data. Towards solving this problem, we make the following contributions.

(1) We propose a novel adversarial framework for transferring knowledge from unlabeled data to the labeled data without any explicit pretext task, making it highly efficient.

(2) We perform extensive experiments on multiple datasets and show that our method achieves promising results, while comparing with state-of-the-art methods, without requiring any labeled data from the source domain.

2 RELATED WORK

Our work relates to four major research directions: semi-supervised learning, domain adaptation, self-training and self-taught learning.

Semi-Supervised Learning has been studied from multiple perspectives (see review [4]). A number of prior works focus on adding regularizers to the model training which prevent overfitting to the labeled examples [14, 23, 42, 45]. Various strategies have been studied, including manifold regularization [3], regularization with graphical constraints [1], and label propagation [54]. In the context of deep neural networks, semi-supervised learning has also been extensively studied using ladder networks [3], temporal ensembling [23], stochastic transformations [43], virtual adversarial training [32], and mean teacher [45]. While most of the existing semi-supervised methods consider the unlabeled set to have the same distribution as the labeled set, we do not assume any correlation between unlabeled data and classification tasks of interest.

Domain Adaptation aims to transfer knowledge from the labeled source data to the unlabeled target data assuming that both contain exactly the same number of classes [11, 29, 30]. In contrast, our work focuses on the opposite case of knowledge transfer without any assumption on the label space. Though open-set domain adaptation [28, 36, 41] does not consider exactly the same classes, but they still have a few classes of interest that are shared between source and target data. In contrast, we don't require them to have the same distribution or shared label space while transferring knowledge from unlabeled data.

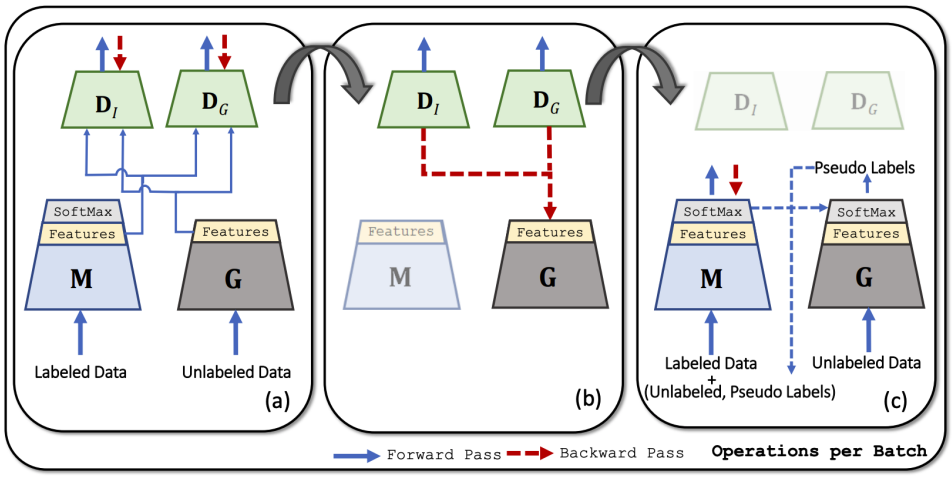

Figure 2: Overview of different operations occurring per batch in the proposed approach. Given a small target dataset, with true labels, and a large amount of unlabeled source data, (a) we first forward pass both labeled and unlabeled samples through the classifier M and the pseudo-label generator G, respectively, to extract their features. We then train a instance-level discriminator D_I to distinguish between target and source features for each instance and a group-level discriminator D_G to distinguish between mean of target and source features in a batch. (b) Next, we update G to fool both D_I and D_G using adversarial loss. (c) We copy weights of softmax layer from M to G and generate pseudo-labels for unlabeled samples. We then update the classifier M, which is the main output of our approach, using both labeled target and pseudo-labeled source data. Best viewed in color.

Self-Training is a learning paradigm where information from a small set of labeled data is exploited to estimate the pseudo labels of unlabeled data. Pseudo labeling [2, 26] is a commonly used technique where a model trained on the label set is first used to predict the pseudo labels of unlabeled set and jointly trained with both labeled and unlabeled data. However, this may result in incorrect labeling if the initial model learned from labeled instances is overfitted. Moreover, it makes an implicit assumption that the source sample features are well aligned with the target data which may not be true. In contrast, our approach makes no such assumption and trains a generator to produce more reliable pseudo-labels by aligning source sample features with the target data distribution.

Self-Taught Learning or Self-Supervised Learning mostly defines a pretext task to learn the feature representation from a large unlabeled set and then finetunes the learned model on the target task involving a much smaller labeled set. Some methods define the pretext task as denoising auto-encoder where the task is data reconstruction at the pixel level from partial observations [46]. The colorization problem [24, 52] is also a notable example, where the task is to reconstruct a color image given its gray scale version. Image inpainting [39] is also used as a pretext task, where the goal is to predict a region of the image given the surrounding. Solving Jigsaw puzzles [33] or its variant [34] has been used as pretext task for learning visual features from unlabeled data. However, defining pretext tasks for certain applications is a challenging problem on its own merit [19]. While our approach is related to self-taught learning, unlike existing works that often employ two stage training, we exploit unlabeled data along with labeled data using an end-to-end framework which updates the classifier with labeled and unlabeled data simultaneously, making it highly efficient.

3 METHODOLOGY

We propose an **A**dversarial **K**nowledge **T**ransfer (**AKT**) framework for transferring knowledge from unlabeled to labeled data without requiring any correspondence across the label spaces. Our goal is to leverage unlabeled images from other object classes which are much easier to obtain than images of the target classes to improve the test accuracy on the target task. We first precisely define the problem that we aim to solve in this work and then present our knowledge transfer approach followed by the optimization details.

3.1 Problem Statement

Consider a labeled dataset $\mathcal{X}_t = \{(x_t^1, y_t^1), (x_t^2, y_t^2), \cdots, (x_t^n, y_t^n)\}$ and an unlabeled source dataset $\mathcal{X}_s = \{x_s^1, x_s^2, \ldots, x_s^m\}$, where $y_t^i \in \mathbb{R}^c$ is the class label of target sample $\mathbf{x}_t^i$, c is the total number of target classes and $m >> n$. Our objective is to generate pseudo-labels $\bar{\mathbf{y}}_s = \{\bar{y}_s^1, \bar{y}_s^2, \ldots, \bar{y}_s^m\}$ for unlabeled samples in the label space of the target domain such that it improves the performance of the classifier when trained jointly. More specifically, we aim to use the unlabeled set $\mathcal{X}_s$ along with $\mathcal{X}_t$ to boost the performance of the classifier compared to using only $\mathcal{X}_t$.

3.2 Adversarial Knowledge Transfer

Let us define the classifier $\mathbf{M}(x), x \in \mathcal{X}_t \cup \mathcal{X}_s$ using a deep convolutional neural network and the pseudo-label generator $\mathbf{G}(x), x \in \mathcal{X}_s$ with the same architecture as the classifier. The goal of pseudo-label generator is to predict the labels of the unlabeled source samples, $\bar{\mathbf{y}}_s$ in such a way so that the source samples seem to have been drawn from the labeled data distribution. However, since the unlabeled source samples do not follow the same label space as the target data, we need feature alignment across unlabeled and labeled

samples such that more reliable pseudo-labels can be generated for the unlabeled source samples. To achieve this, we introduce two discriminators, namely instance-level and group-level discriminator which act as feature aligners between the classifier,**M**, and the pseudo-label generator, **G**. The instance-level discriminator, $\mathbf{D}_I$, learns to detect whether the feature is from the classifier or pseudo-label generator using an adversarial training. On the other hand, the group level-discriminator, $\mathbf{D}_G$, aims to learn holistic statistical information by distinguishing if the mean feature is from the classifier or the pseudo-label generator. Specifically, through these two discriminators, we try to discover both localized knowledge at the instance level and the global feature distribution knowledge while aligning features across both source and target domains.

Let the feature representation at k^{th} layer be defined as $\mathbf{M}^k(x)$ and $\mathbf{G}^k(x)$ for the classifier and the pseudo-label generator respectively. In our experiments, we choose the features from the last fully connected layer (denoted by L) as it has shown to be effective in many transfer learning tasks. We learn the parameters of **G**, $\mathbf{D}_I$, $\mathbf{D}_G$ and **M** using adversarial training, as described in Algorithm 1. The learned classifier is finally used in evaluation on the target task. Moreover, we adopt the same network architecture for both, the classifier and the pseudo-label generator, in order to ensure that the feature representations of both labeled and unlabeled samples are in the same space for feature alignment. Notice that our pseudo-label generator mimics the generator in Generative Adversarial Networks (GANs) [13] which generates an image from a random vector. However, unlike generators in GANs, the input to our generator is an image rather than a latent noise vector and it produces pseudo-labels for the given inputs.

3.3 Optimization

For transferring knowledge from unlabeled data, we propose to adapt the two-player min-max game based on the discriminators and the pseudo-label generator. Towards this, we need to learn the parameters of generator **G**, and the discriminators in a way that **G** is able to generate features such that both $\mathbf{D}_I$ and $\mathbf{D}_G$ are not able to discriminate between instance-level and group-level features from target or source distribution respectively. We consider classifier features as positive and pseudo-label generator features as negative and then train the discriminators using binary cross-entropy loss. On the other hand, the generator is trained to produce features such that the discriminators are fooled. Thus, similar to the GANs, the pseudo-label generator **G** and the discriminators, $\mathbf{D}_I$ and $\mathbf{D}_G$, play the following two-player min-max game:

$$\min_{\mathbf{G}} \max_{\mathbf{D}_I, \mathbf{D}_G} \mathbb{E}_{p_{M(x_t)}} \left[\log \mathbf{D}_I\Big(\mathbf{M}^L(x_t)\Big) + \log \mathbf{D}_G\Big(\mathbf{M}^L(x_t)\Big) \right] + \mathbb{E}_{p_{G(x_s)}} \left[\log\Big(1 - \mathbf{D}_I\Big(\mathbf{G}^L(x_s)\Big)\Big) + \log\Big(1 - \mathbf{D}_G\Big(\mathbf{G}^L(x_s)\Big)\Big) \right] \quad (1)$$

where, $p_{M(x_t)}$ and $p_{G(x_s)}$ correspond to the feature distribution of **M** in target domain and **G** in source domain. We solve the optimization problem in eqn. 1 alternately using gradient descent where we once fix the parameters of the generator **G** and train the discriminators $\mathbf{D}_I$ and $\mathbf{D}_G$ and vice versa as described below.

Algorithm 1 Adversarial Knowledge Transfer Training

Input: $\mathcal{X}_t = \{(\boldsymbol{x_t}^i, y_t^i)\}_{i=1}^n$, $\mathcal{X}_s = \{\boldsymbol{x_s}^j\}_{j=1}^m$
Output: Classifier Model **M**
for number of training iterations **do**
  Sample b tuples of $(\boldsymbol{x_t}, y)$ from $\mathcal{X}_t$
  Sample b samples of $\boldsymbol{x_s}$ from $\mathcal{X}_s$
  Step a. Minimize $\mathcal{L}_D$ using Eqn. 4 and update $\mathbf{D}_I$, $\mathbf{D}_G$
  Step b. Maximize $\mathcal{L}_G$ using Eqn. 5 and update **G**
  Step c. Minimize $\mathcal{L}_M$ using Eqn. 6 and update **M**
end for

Discriminator Training. Given two discriminators for instance-level and group-level feature alignment across the classifier **M** and the pseudo-label generator **G**, during training, we first update the weights of the discriminators $\mathbf{D}_I$ and $\mathbf{D}_G$ with features coming from **M** as positive and that coming from **G** as negative. For a batch of size b, instance-level discriminator loss ($\mathcal{L}_{D_I}$) and group-level discriminator loss ($\mathcal{L}_{D_G}$) on mean features are defined as follows.

$$\mathcal{L}_{D_I} = \frac{1}{b}\sum_{i=1}^{b} \log \mathbf{D}_I\big(\mathbf{M}^L(x_t^i)\big) + \frac{1}{b}\sum_{i=1}^{b} \log\Big(1 - \mathbf{D}_I\big(\mathbf{G}^L(x_s^i)\big)\Big) \quad (2)$$

$$\mathcal{L}_{D_G} = \log \mathbf{D}_G\Big(\frac{1}{b}\sum_{i=1}^{b} \mathbf{M}^L(x_t^i)\Big) + \log\Big(1 - \mathbf{D}_G\Big(\frac{1}{b}\sum_{i=1}^{b} \mathbf{G}^L(x_s^i)\Big)\Big) \quad (3)$$

Finally, the total loss for discriminators $\mathcal{L}_D$ is given as

$$\mathcal{L}_D = \lambda_{D_I}\mathcal{L}_{D_I} + \lambda_{D_G}\mathcal{L}_{D_G} \quad (4)$$

where, λ_{D_I} and λ_{D_G} are the weights for the instance and group discriminator losses respectively.

Pseudo-Label Generator Training. The objective of training **G** is to fool the discriminators such that the discriminator fails to distinguish whether the feature is coming from the classifier or pseudo-label generator. The loss function to train the pseudo-label generator **G** can be written as follows.

$$\mathcal{L}_G = \frac{1}{b}\sum_{i=1}^{b} \log\Big(1 - \mathbf{D}_I\Big(\mathbf{G}^L(x_s^i)\Big)\Big) + \log\Big(1 - \mathbf{D}_G\Big(\frac{1}{b}\sum_{i=1}^{b} \mathbf{G}^L(x_s^i)\Big)\Big) \quad (5)$$

Classifier Training. The classifier is updated at the end of each iteration using both labeled and unlabeled data. We can update **M** easily for the labeled data but since there are no labels for the unlabeled source data, we use prediction from the generator as true labels to compute the loss. The loss function to update the classifier for an iteration with batch size b can be represented as follows:

$$\mathcal{L}_M = \frac{1}{b}\sum_{i=0}^{b} \mathcal{L}\Big(\mathbf{M}(x_t^i), y_t^i\Big) + \lambda_s\, \mathcal{L}\Big(\mathbf{M}(x_s^i), \mathbf{G}(x_s^i)\Big) \quad (6)$$

where $\mathcal{L}(p, q)$ is the categorical cross-entropy loss multi-class and binary cross-entropy loss for multi-label classification, p as the true label and q as the predicted label. The loss originating from the source samples is used as a regularizing term with parameter λ_s so that it assists the learning of target task and not suppressing it.

4 EXPERIMENTS

We perform rigorous experiments on different visual recognition tasks, such as Object Recognition (single-label and multi-label), Character Recognition (Font and Handwritten), and Sentiment Recognition to verify the effectiveness of our approach. Our primary objective is to transfer knowledge from unlabeled data to the classifier network such that the test accuracy on the target data significantly improves over the training from scratch (i.e., training with only target data without any knowledge transfer) and approaches as close to that of supervised knowledge transfer methods that uses the labeled source data. Note that we do not require the source and target domains to have the same distribution or label space across all our experiments.

4.1 Implementation Details

All our implementations are based on PyTorch [37]. We choose VGG-16 [44] as the network architecture for both classifier and generator in all our experiments except CIFAR-10 and Pasca-VOC experiments. For CIFAR-10 experiment, to maintain the same network architecture as with other methods, we modify the last three fully connected layers to (512-512-10) from (4096-4096-10). In Pascal-VOC experiment we use AlexNet to compare with other state-of-the-art methods [20, 20, 33, 38, 39, 52, 53]. We choose the same network architecture for both classifier and pseudo-label generator as the features generated by these networks should lie in the same space and thus it helps in the feature alignment. We use three fully-connected (FC) layers (512-256-128) as the discriminator network for CIFAR-10 experiment and (4096-1024 -512) for all other experiments. We use SGD with learning rate 0.01, 0.01 and 0.001 for the classifier, pseudo-label generator and discriminator, respectively. We use momentum of 0.9 and weight decay of 0.0005 while training the classifier. The learning rate of the classifier and generator is reduced by a factor of 0.1 after 75 epochs. All the loss weights are set to 1 and kept fixed for all experiments. We train our models with a batch size of 96 for all experiments except for PASCAL-VOC where we use batch of 20. In each iteration, the generator is updated once using unlabeled source dataset and the discriminator is updated twice using labeled target and unlabeled source dataset.

4.2 Compared Methods

We compare with several methods that fall into two main categories. (1) Supervised knowledge transfer methods that use labeled source data, such as Finetuning and Joint Training (i.e., multi-task learning). (2) Unsupervised knowledge transfer methods (a.k.a self-supervised methods) that leverage unlabeled source data, such as Random Network [51], Pseudo Labels [26], Jigsaw [33], Colorization [52] and Split-Brain Autoencoder [53]. We additionally compare with the training from scratch baseline to show the effectiveness of our knowledge transfer approach for improving recognition accuracy on the target task. Below are the brief descriptions on the baselines.

Finetuning and Joint Training. In Finetuning, we first train a classifier using source data by assuming that the true labels are available during training and then perform an end-to-end finetuning on the target dataset. The Joint Learning baseline learns a shared representation using a single network on both source and target datasets. Since both of these methods use labeled source data, we call them as supervised knowledge transfer methods.

Random Network. Following [34, 51], we perform an experiment to obtain random pseudo labels for source data by clustering the features extracted from a randomly initialized network. We then train the classifier network using these pseudo-labels for source data and finetune it on the labeled target dataset.

Pseudo Labels. We first train a classifier on the labeled target data and obtain the pseudo labels for source data by making predictions using the classifier. We then update the network using both labeled target and pseudo-labeled source images to obtain an improved model for recognition on target task.

Jigsaw [33]. We use solving jigsaw puzzles as pretext task on unlabeled source dataset and then finetune on target task. We use same network (VGG-16) to make a fair comparison with our approach.

Colorization [52]. Following [52], we use colorization (mapping from grayscale to color version of a photograph) as a form of self-supervised feature learning on unlabeled source data and then finetune on labeled target data.

Split-Brain Autoencoder [53]. Split-Brain Autoencoder uses the cross-channel prediction to learn features on unlabeled data where one sub-network solves the problem of colorization (predicting a and b channels from the L channel in Lab colorspace), and the other perform the opposite (synthesizing L from a, b channels).

As character recognition task involves grayscale images, we perform colorization and split-brain autoencoder experiments on only object recognition and sentiment recognition tasks. We use publicly available code of both methods and set the parameters as recommended in the published works.

4.3 Object Recognition

The goal of this experiment is to verify the effectiveness of our proposed adversarial approach in both single and multi-label object recognition while leveraging unlabeled data (from different classes as the target task) which are abundantly available from the web.

4.3.1 Single-label Object Recognition. We conduct this experiment using CIFAR-10 [21] as the labeled target dataset and CIFAR-100 [21] as the unlabeled source dataset. Both datasets contain 50,000 training images and 10,000 test images of size 32×32. We use the same train/test split from the original CIFAR-10 dataset for our experiments. Note that the classes in CIFAR-10 and CIFAR-100 are mutually exclusive. Table 1 shows results of different methods on CIFAR-10 dataset. From Table 1, the following observations can be made: (1) The classifier trained using our approach outperforms all the unsupervised knowledge transfer (self-supervised) methods. Among the alternatives, Split-Brain baseline is the most competitive. However, our approach still outperforms it by a margin of 0.61% due to the introduced feature alignment using adversarial learning. (2) As expected Joint Training outperforms both Off-the-Shelf and Finetuning baseline as it learns a shared representation from both dataset by transferring knowledge across them.

4.3.2 Multi-label Object Recognition. We conduct this experiment on the more challenging multi-label PASCAL-VOC [9] as the labeled target and ImageNet [8] as the unlabeled source dataset.

Table 1: Experimental Results on Single-label Object Recognition task. The proposed approach, AKT outperforms all the unsupervised knowledge transfer methods.

Target: CIFAR-10 and Source: CIFAR-100	
Methods	Target Accuracy (%)
Scratch	92.49
Finetuning	93.27
Joint Training	93.32
Pseudo Labels [2]	92.85
Random Network [39]	92.37
Jigsaw [33]	75.85
Colorization [52]	92.57
Split-Brain [53]	92.60
AKT (Ours: only $\mathbf{D}_I$)	93.04
AKT (Ours: with $\mathbf{D}_I$ and $\mathbf{D}_G$)	**93.21**

Table 2: Comparison of our method with state-of-the-art self-supervised alternatives on PASCAL-VOC Multi-label Object Classification task. The reported results of all the self-supervised methods except Rotation, Rotation Decoupling, and Pseudo Labels are from [34].

Target: PASCAL-VOC and Source: ImageNet	
Methods	Target mAP (%)
Scratch	63.5
Finetuning	87.0
Joint Training	86.7
Pseudo Labels [2]	63.2
Random Network [39]	53.3
Jigsaw [33]	67.7
Jigsaw++ [34]	69.9
Colorization [52]	65.9
Split-Brain [53]	67.1
Rotation [12]	73.0
Rotation Decoupling [10]	74.5
AKT (Ours: only $\mathbf{D}_I$)	76.9
AKT (Ours: with $\mathbf{D}_I$ and $\mathbf{D}_G$)	**77.4**

We follow the standard train/test split [9] to perform our experiments. Table 2 shows the results. We have the following observations from Table 2. (1) Our proposed method outperforms all other self-supervised methods including the Rotation Decoupling method in [10] by a significant margin (2.9% improvement in mAP). Note that we train the source and target data jointly and since ImageNet is a large scale dataset of 1.1M images and PASCAL-VOC has only 4982 images, our method utilizes a small fraction of data from the ImageNet to achieve this state-of-the-art performance on the PASCAL-VOC dataset. (2) The performance gap between our method and supervised knowledge transfer methods begins to increase. This is expected as with a challenging multi-label object classification dataset, an unsupervised approach can not compete with a fully supervised knowledge transfer approach, especially, when the latter one is using true labels from a large scale source dataset like ImageNet. However, we would like to point out once more that, in practice, supervised knowledge transfer methods have serious issues with scalability as they require a tremendous amount of manual annotations. On the other hand, our approach can be trained on internet-scale datasets with no supervision. (3) The performance gap with Random Network baseline is much higher (53.3% vs 77.4%), justifying the relevance of our pseudo-label prediction using aligned features while transferring knowledge from a large unlabeled dataset. (4) Our proposed approach works the best while both instance-level and group-level discriminators are used for feature alignment (76.9% vs 77.4%).

Table 3: Results on Font Character Recognition task. Our approach outperforms all the self-supervised baselines and is very competitive against the supervised topline (~0.04%).

Target: Char74K and Source: EMNIST	
Methods	Target Accuracy (%)
Scratch	8.55
Finetuning	19.85
Joint Training	18.60
Pseudo Labels [2]	8.54
Random Network [39]	9.04
Jigsaw [33]	18.37
AKT (Ours: only $\mathbf{D}_I$)	**19.81**
AKT (Ours: with $\mathbf{D}_I$ and $\mathbf{D}_G$)	19.79

Table 4: Experimental Results on Handwritten Character Recognition. Note that the performance of Random Network is very competitive for this task.

Target: EMNIST and Source: MNIST	
Methods	Target Accuracy (%)
Scratch	92.24
Finetuning	93.85
Joint Training	93.80
Pseudo Labels [2]	89.16
Random Network [39]	92.35
Jigsaw [33]	50.90
AKT (Ours: only $\mathbf{D}_I$)	93.59
AKT (Ours: with $\mathbf{D}_I$ and $\mathbf{D}_G$)	**93.65**

4.4 Character Recognition

The goal of this experiment is to compare our approach with other alternatives on recognizing both font and handwritten characters.

4.4.1 Font Character Recognition. We use Char74K [7], a font character recognition dataset as the labeled target set and handwritten character split from EMNIST dataset [5] as the source dataset to perform this experiment. Char74K dataset consists of 74,000 images from 64 classes that includes English alphabets (a-z,A-Z) and numeric (0-9). EMNIST dataset consists of 145,600 images for 26 classes (a-z). For the Char74K dataset, we divide the data into 80/20

Table 5: Results on Advertisement Dataset. Our proposed approach performs the best among the self-supervised alternatives and is very competitive against supervised baselines.

Target: Advertisement and Source: BAM	
Method	Target Accuracy (%)
Scratch	25.02
Finetuning	29.00
Joint Training	29.51
Pseudo Labels [2]	25.02
Random Network [39]	25.51
Jigsaw [33]	28.38
Colorization [52]	16.12
Split-Brain [53]	27.71
AKT (Ours: only $\mathbf{D}_I$)	28.70
AKT (Ours: with $\mathbf{D}_I$ and $\mathbf{D}_G$)	**28.87**

train/test split and use the standard train/test split for EMNIST. From Table 3, we have the following observations. (1) Similar to the object recognition results, our proposed approach outperforms both Random Network and Pseudo Labels baseline by a significant margin (11% improvement over Random Network). Among the alternatives, Jigsaw baseline is the most competitive. However, we still outperform the Jigsaw baseline by a margin of about 1.5% in recognizing font characters. (2) Our approach is very competitive to the supervised knowledge transfer baselines (19.81% vs 19.85%) but significantly outperforms the training from scratch baseline by 11%, thanks to the effective knowledge transfer from unlabeled source data to the target dataset.

4.4.2 Handwritten Character Recognition. We conduct this experiment using EMNIST as the labeled target dataset and MNIST [25] as the unlabeled source dataset. While EMNIST contains handwritten characters, MNIST is a standard dataset for handwritten digits containing 80,000 images. We use the original train/test split from both datasets in our experiments. Table 4 shows that our proposed method achieves performance very close to the supervised methods which indicates that our method can achieve comparable performance without requiring a single label from the source dataset. Note that the performance of Jigsaw baseline is only 50.90% which is much lower than that of training from scratch. We believe this is because the Jigsaw is optimized to work for 256×256 size images in training [33] and hence fails to learn efficient features with 28×28 small size images on this task.

4.5 Sentiment Recognition

Recently, sentiment analysis has garnered much interest in computer vision as there is more to images than their objective physical content: for example, advertisements are created to persuade a viewer to take a certain action. However, collecting manual labels for advertisement images are very difficult and costly compared to generic object labels. The goal of this experiment is to leverage unlabeled images for improving performance of a sentiment recognition classifier on advertisement images. We conduct this experiment using the Advertisement dataset [18] as the labeled target and Behance-Artistic-Media (BAM) [48] dataset as the unlabeled source dataset. While the Advertisement dataset contains 64,382 images across 30 sentiment categories, the BAM dataset consists of 2.5 million images across 4 sentiment categories.

Following are the observations from Table 5: (1) Our method consistently outperforms the other baselines to achieve the top accuracy of 28.87% on the Advertisement dataset. Jigsaw baseline is the most competitive with an accuracy of 28.38%. We observe that Jigsaw baseline is able to transfer the knowledge well only if the pretext task is trained on a very large-scale dataset containing millions of images. (2) Since our algorithm processes same amount of the source and the target data, it uses much lower number of unlabeled samples from the source dataset. However, by utilizing some fraction of unlabeled samples, our approach is only about 0.13% behind Finetuning baseline that uses all the unlabeled source samples along with their labels for training the source classifier.

5 ABLATION ANALYSIS

We perform the following ablation experiments to better understand the contributions of various components of our proposed approach.

5.1 Advantage of Feature Alignment

To better understand the contribution of feature alignment across target features from the classifier and source features from the pseudo-label generator, we perform an experiment where we jointly train the classifier with labeled target data and pseudo-labeled source data without any feature alignment. We observe that our approach without feature alignment produces inferior results, with target accuracy of 92.75% on CIFAR-10/CIFAR-100 and 67.10% on PASCAL-VOC/ImageNet experiments respectively. These performances are about 0.5% and 10% lower compared to our approach using feature alignment for CIFAR-10/CIFAR-100 and PASCAL-VOC/ImageNet experiments respectively. Thus, we conjecture that feature alignment is an essential step for predicting reliable pseudo-labels while transferring knowledge from the unlabeled data.

5.2 Different Layer for Alignment

We examine the performance of our approach by using adversarial loss across fc6 instead of fc7 features along with conv5 features and found that feature alignment across fc6 produces inferior results (92.85% vs 93.21%) on CIFAR-10/CIFAR-100 experiment. This suggests that fc7 features along with conv5 features are better suited for transfer learning as shown in many prior works.

5.3 Impact of Network Architecture

In the supervised setting, deeper architectures have shown higher performance on many tasks. To study the impact of choice of architecture we perform experiments on object recognition tasks using VGG16 and ResNet50 architectures. As expected, we observe that a deeper architectures VGG16 and ResNet50 leads to significant improvement as compared to AlexNet. From Table 6a we observe that on the CIFAR-10/CIFAR-100 task our proposed approach achieves 93.21% with VGG16 and 94.65% using ResNet50 as compared to 89.04% on AlexNet. Similar performance trend is observed on multi-label recognition (Table 6b), where we achieve improvement of 0.8% mAP and 2.2% mAP using VGG16 and ResNet50, respectively.

Table 6: Performance of proposed method with different architecture on Object Recognition tasks.

Target: CIFAR-10 and Source: CIFAR-100	
Architecture	Target Accuracy (%)
AlexNet [22]	89.04
VGG16 [44]	93.21
ResNet50 [16]	94.65

(a) Single-Label Object Recognition

Target: PASCAL-VOC and Source: ImageNet	
Architecture	Target mAP (%)
AlexNet [22]	77.4
VGG16 [44]	78.2
ResNet50 [16]	79.6

(b) Multi-Label Object Recognition

(a) Samples from class *aeroplane* from PASCAL-VOC experiment.

(b) Samples from class *bike* from PASCAL-VOC experiment.

Figure 3: Predicted pseudo-labels on ImageNet data. Top-row (blue) are the samples from the target PASCAL-VOC dataset. Middle-row (green) shows the samples from ImageNet dataset where pseudo labels correspond to correct labels. Bottom-row (red) shows the source samples where object in image does not match the correct label but has visual similarity to the target class. From the Middle-rows, we observe that our pseudo-label generator predicts the correct label of many unlabeled samples such as aeroplanes in (a) and bike in (b). Our approach classifies bird images (which resembles aeroplane) and fins of dolphins (which resembles wings of a aircraft) as aeroplanes as seen in bottom row of (a). Similarly, since target data in (b) have bike wheel as prominent feature, source samples with circular features are pseudo-labeled as bike. Best viewed in color.

5.4 Reliability of Pseudo Labels

From Figure 3a- 3b, we observe that the pseudo-label generator G is able to generate correct labels for many samples in the source dataset (middle row) and generates reasonable labels for visually similar classes (bottom row) as well. We further compute reliability score by comparing the correct labels and the pseudo-labels of about 700 randomly sampled unlabeled images from the ImageNet dataset and find that the reliability score is more than 85% for both of the classes as shown in Table 7.

5.5 MSE Loss vs Adversarial Loss

We investigate the importance of adversarial loss by comparing with L_2 loss for aligning the features on the CIFAR-10/CIFAR-100 task, and found that the later produces inferior results with a target accuracy of 92.66% compared to 93.21% by the adversarial loss.

5.6 Out-of-Distribution Source Data

Following [15], we perform an experiment using mini-ImageNet as the unlabeled source dataset and EuroSAT containing remote sensing images with no perspective distortion, as the labeled target dataset. Our proposed approach outperforms the Pseudo-Labels baseline by a margin of 0.72% (95.22% vs 94.5%) and is competitive with the supervised Fine-tuning baseline (95.22% vs 96.19%) using AlexNet as the network architecture.

Table 7: Top-3 pseudo label predictions on ImageNet classes.

ImageNet Class	Top-3 Pseudo Label	Score
Fig. 3a. warplane	**aeroplane**, bird, car	86.67%
Fig. 3b. bike	**bicycle**, motorbike, person	88.46%

6 CONCLUSION

We present an Adversarial Knowledge Transfer (**AKT**) approach for transferring knowledge from unlabeled data to the labeled data without requiring the unlabeled data to be from the same label space or data distribution as of the labeled data. The proposed adversarial learning jointly trains the classifier using both labeled target data and source data whose labels are predicted using a pseudo-label generator by aligning the feature space of unlabeled data with the labeled target data. Experiments show that our approach not only outperforms the unsupervised knowledge transfer alternatives but is also very competitive while comparing against supervised knowledge transfer methods.

ACKNOWLEDGMENTS

This work is partially supported by NSF grant 1724341, ONR grant N00014-18-1-2252 and gifts from Adobe. We thank Abhishek Aich for his assistance and feedback on the paper.

REFERENCES

[1] Rie K Ando and Tong Zhang. 2007. Learning on graph with Laplacian regularization. In *NIPS*. 25–32.

[2] Philip Bachman, Ouais Alsharif, and Doina Precup. 2014. Learning with pseudo-ensembles. In *Advances in Neural Information Processing Systems*. 3365–3373.

[3] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. 2006. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *JMLR* 7, Nov (2006), 2399–2434.

[4] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. 2009. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks* 20, 3 (2009), 542–542.

[5] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. 2017. EMNIST: an extension of MNIST to handwritten letters. *arXiv preprint arXiv:1702.05373* (2017).

[6] Wenyuan Dai, Qiang Yang, Gui-Rong Xue, and Yong Yu. 2008. Self-taught clustering. In *Proceedings of the 25th international conference on Machine learning*. ACM, 200–207.

[7] Teófilo Emídio De Campos, Bodla Rakesh Babu, Manik Varma, et al. 2009. Character recognition in natural images. *VISAPP (2)* 7 (2009).

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. (2009).

[9] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. 2010. The pascal visual object classes (voc) challenge. *International journal of computer vision* 88, 2 (2010), 303–338.

[10] Zeyu Feng, Chang Xu, and Dacheng Tao. 2019. Self-Supervised Representation Learning by Rotation Feature Decoupling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 10364–10374.

[11] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research* 17, 1 (2016), 2096–2030.

[12] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. 2018. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728* (2018).

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *NIPS*. 2672–2680.

[14] Yves Grandvalet and Yoshua Bengio. 2005. Semi-supervised learning by entropy minimization. In *Advances in neural information processing systems*. 529–536.

[15] Yunhui Guo, Noel CF Codella, Leonid Karlinsky, John R Smith, Tajana Rosing, and Rogerio Feris. 2019. A new benchmark for evaluation of cross-domain few-shot learning. *arXiv preprint arXiv:1912.07200* (2019).

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*. 770–778.

[17] Cheng-An Hou, Min-Chun Yang, and Yu-Chiang Frank Wang. 2014. Domain adaptive self-taught learning for heterogeneous face recognition. In *2014 22nd International Conference on Pattern Recognition*. IEEE, 3068–3073.

[18] Zaeem Hussain, Mingda Zhang, Xiaozhong Zhang, Keren Ye, Christopher Thomas, Zuha Agha, Nathan Ong, and Adriana Kovashka. 2017. Automatic understanding of image and video advertisements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1705–1715.

[19] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. 2019. Revisiting Self-Supervised Visual Representation Learning. *CVPR* (2019).

[20] Philipp Krähenbühl, Carl Doersch, Jeff Donahue, and Trevor Darrell. 2015. Data-dependent initializations of convolutional neural networks. *arXiv preprint arXiv:1511.06856* (2015).

[21] Alex Krizhevsky and Geoffrey Hinton. 2009. *Learning multiple layers of features from tiny images*. Technical Report. Citeseer.

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *NIPS*. 1097–1105.

[23] Samuli Laine and Timo Aila. 2016. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242* (2016).

[24] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. 2016. Learning representations for automatic colorization. In *European Conference on Computer Vision*. Springer, 577–593.

[25] Yann LeCun. 1998. The MNIST database of handwritten digits. *http://yann. lecun. com/exdb/mnist/* (1998).

[26] Dong-Hyun Lee. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, Vol. 3. 2.

[27] Honglak Lee, Rajat Raina, Alex Teichman, and Andrew Ng. 2009. Exponential family sparse coding with applications to self-taught learning. In *Twenty-First International Joint Conference on Artificial Intelligence*.

[28] Hong Liu, Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Qiang Yang. 2019. Separate to Adapt: Open Set Domain Adaptation via Progressive Separation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2927–2936.

[29] Mingsheng Long, Jianmin Wang, Guiguang Ding, Jiaguang Sun, and Philip S Yu. 2013. Transfer feature learning with joint distribution adaptation. In *Proceedings of the IEEE international conference on computer vision*. 2200–2207.

[30] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. 2016. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems*. 136–144.

[31] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. 2017. Deep transfer learning with joint adaptation networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2208–2217.

[32] Takeru Miyato, Shin-ichi Maeda, Shin Ishii, and Masanori Koyama. 2018. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence* (2018).

[33] Mehdi Noroozi and Paolo Favaro. 2016. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*. Springer, 69–84.

[34] Mehdi Noroozi, Ananth Vinjimoor, Paolo Favaro, and Hamed Pirsiavash. 2018. Boosting self-supervised learning via knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 9359–9367.

[35] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. 2014. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1717–1724.

[36] Pau Panareda Busto and Juergen Gall. 2017. Open set domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*. 754–763.

[37] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. (2017).

[38] Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. 2017. Learning features by watching objects move. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2701–2710.

[39] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. 2016. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2536–2544.

[40] Rajat Raina, Alexis Battle, Honglak Lee, Benjamin Packer, and Andrew Y Ng. 2007. Self-taught learning: transfer learning from unlabeled data. In *Proceedings of the 24th international conference on Machine learning*. ACM, 759–766.

[41] Kuniaki Saito, Shohei Yamamoto, Yoshitaka Ushiku, and Tatsuya Harada. 2018. Open set domain adaptation by backpropagation. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 153–168.

[42] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. 2016. Mutual exclusivity loss for semi-supervised deep learning. In *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 1908–1912.

[43] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. 2016. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *NIPS*. 1163–1171.

[44] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[45] Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems*. 1195–1204.

[46] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*. ACM, 1096–1103.

[47] Hua Wang, Feiping Nie, and Heng Huang. 2013. Robust and discriminative self-taught learning. In *International conference on machine learning*. 298–306.

[48] Michael J Wilber, Chen Fang, Hailin Jin, Aaron Hertzmann, John Collomosse, and Serge Belongie. 2017. Bam! the behance artistic media dataset for recognition beyond photography. In *Proceedings of the IEEE International Conference on Computer Vision*. 1202–1211.

[49] Yonghui Xu, Sinno Jialin Pan, Hui Xiong, Qingyao Wu, Ronghua Luo, Huaqing Min, and Hengjie Song. 2017. A unified framework for metric transfer learning. *IEEE Transactions on Knowledge and Data Engineering* 29, 6 (2017), 1158–1171.

[50] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks?. In *Advances in neural information processing systems*. 3320–3328.

[51] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2017. Understanding deep learning requires rethinking generalization. In *ICLR*.

[52] Richard Zhang, Phillip Isola, and Alexei A Efros. 2016. Colorful image colorization. In *European conference on computer vision*. Springer, 649–666.

[53] Richard Zhang, Phillip Isola, and Alexei A Efros. 2017. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1058–1067.

[54] Xiaojin Zhu and Zoubin Ghahramani. 2002. *Learning from labeled and unlabeled data with label propagation*. Technical Report. Citeseer.