# StFT: Style loss and Fourier Transformation for Domain Gap Reduction

Sarthak Srivastava
Amazon.com
sarthasr@amazon.com

Khaleeque Ansari
Amazon.com
akhaleeq@amazon.com

## ABSTRACT

This work explores the usage of Fourier Transform in conjunction with Triplet loss applied on image styles, for reduction of the domain gap between the Source (e.g. Product Images in natural setting) and Target domain (e.g. Product Images on Ecommerce store pages) towards solving the Domain Adaptation problem. Most Unsupervised Domain Adaptation (UDA) algorithms reduce the domain gap between labelled Source domain and the unlabelled Target domain by matching their marginal distribution. UDA is of special interest for several ecommerce applications. An example of this can be identification of live item image captured by a customer. Such identification can help in display of relevant selections available with the ecommerce stores. UDA algorithm performances degrade when the domain shift between the Source and Target domain is substantial. To improve the predictive performance of the existing single source single target UDA algorithms the proposed method **StFT** attempts to reduce the domain gap between the Source and Target domain via low-frequency component swapping and target style enforcement in the feature space upon training image via triplet loss. The proposed technique can be added on top of existing UDA methods. This leads to improvement in their performance without much increase in computational cost. We have evaluated the proposed method for Office-31 data set with the Amazon domain acting as either source or target domain.

## CCS CONCEPTS

• **Computing methodologies** → **Transfer learning**; • **Mathematics of computing** → *Computation of transforms.*

## KEYWORDS

Domain Adaptation, Fourier Transformation, Triplet Loss, Domain Gap

**ACM Reference Format:**
Sarthak Srivastava and Khaleeque Ansari. 2022. StFT: Style loss and Fourier Transformation for Domain Gap Reduction. In *Proceedings of First Content Understanding and Generation for E-commerce Workshop (SIGKDD 2022 Conference on Knowledge Discovery and Data Mining).* ACM, New York, NY, USA, 7 pages.

## 1 INTRODUCTION

In the last decade, computer vision research has been immensely revolutionized by advances in deep learning. This was kicked off by the introduction of AlexNet[11], the first deep convolutional neural network that was introduced to improve the results of ImageNet competition. A typical supervised deep learning model consumes a large amount of labelled samples to improve its predictive performance. A lot of state of the art deep learning architecture have been proposed in past decade. An underlying assumption in the working of most of such methods is existence of a common distribution between the train and test splits. Hence, the generalization power of these methods is not thoroughly evaluated as test is done on unseen test data having same distribution as the training data used for these models. When this assumption breaks, there is a significant reduction in the model performance. In most of the real world applications, a similar distribution might not exist between the training data and the inference data. The difference between the distribution of the training data viz. the Source domain, and the test/inference data viz. the Target domain, is termed as domain gap or domain shift. Domain Adaptation algorithms attempt to improve model performance by reducing this domain shift. Domain Adaptation problem where labels for target domain samples is not available is termed as Unsupervised Domain Adaptation ($UDA$) [26]. In UDA the target domain features show a covariate shift with respect to the source domain features, causing the model trained just using the source domain samples to under performs on target domain. Under this, the conditional distributions $D_s(y|x) = D_t(y'|x')$ but $D_s(x)! = D_t(x')$ where $D_s$, $D_t$ are the source and target domain distributions respectively, $x$ are the source domain features, $y$ are the source domain labels, $x'$ are the target domain features while the $y'$ are the target domain labels.

Being able to bridge this domain gap will enable development of applications that use the accessible source domain labelled samples for training, and are also able to perform well on the target domain samples. Such applications will be of special value to ecommerce industry where it can used to match customer captured live images of products with those available within the ecommerce store's selection. Another example might be live camera based verification of purchased product item delivered to the customer. Most of the domain adaptation methods try to reduce the domain gap by trying to align the marginal distributions[2, 20]. Han Zhao Et. Al[27] have demonstrated in their work that marginal alignment of the Source and Target distribution does not always ensure a joint alignment of distributions $D_s$ and $D_t$. Domain alignment becomes even more difficult as the domain shift between the source and target domain increases.

We propose Style loss and Fourier Transformation (StFT), an approach to close the domain gap between the source and target

*SIGKDD 2022 Conference on Knowledge Discovery and Data Mining,*
,

domain by coupling a Fourier transform based augmentation with an additional style focused triplet loss, applied to existing UDA algorithms. We make an assumption that there are two independent component to an image - **class discriminative** features and **style based** features. Out of the two, style based features are responsible for majority of the domain shift that we find in real world data. As per Fourier Domain Adaptation (FDA)[24], the transfer of target domain style to the source domain via Fourier Transformation reduces the domain gap and improves the performance for segmentation task. The first step in StFT involves **low frequency swapping**, where we apply Fourier transformation as described in FDA[24], for the classification purposes, in order to bridge the domain gap. The target domain samples to be used for the Fourier transformation are randomly selected. The low frequency components of the target domain images swapped to source domain during Fourier transformation represents just the smooth components of images. Hence, it is very possible that not all the style of the target domain gets transferred to the source domain image. To improve upon this, we add a style specific triplet loss component to the overall loss function to be optimized. This forces the features of source domain to converge towards the feature of target domain, in the context of encoded style. The Frequency Transformation step can be visualized in the fig.1.

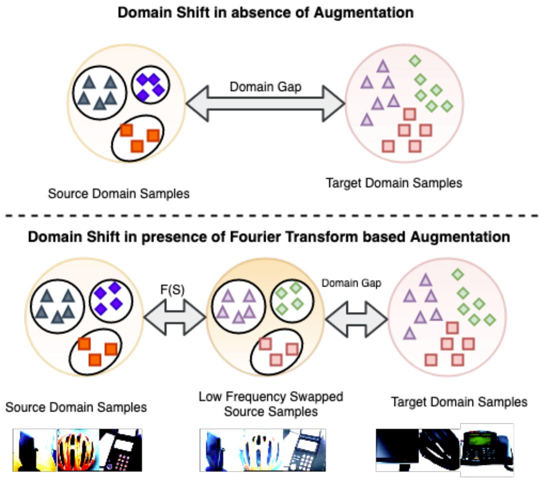

Domain Shift in absence of Augmentation
Domain Gap
Source Domain Samples
Target Domain Samples
Domain Shift in presence of Fourier Transform based Augmentation
F(S)
Domain Gap
Source Domain Samples
Low Frequency Swapped Source Samples
Target Domain Samples

**Figure 1: Frequency Transformation Overview. The *first row* above depicts a general UDA setting where domain shift between labelled source and unlabelled target is substantial. The *second row* shows how swapping of low frequency component in the source domain, with target domain help bridge the domain gap between the two. This is also demonstrated by the qualitative image below each distribution.**

To summarize briefly, the proposed method has following key components.

- The proposed method is essentially a combination of training sample augmentation and a new loss component, applied to existing state of the art UDA methods.
- As a first step, we apply computationally cheap Fourier Transformation based augmentation on source domain samples. This step swaps the low frequency component of the source domain training samples with that of the random target domain samples. This bridges the domain gap between the source and target domain to some extent.
- To further close the domain gap that exists between source and target domain, we add the triplet loss [7] to the overall loss function of UDA method under consideration. The triplet loss works in the feature space and the samples passed to it are selected in such a way so as to account for the contrast between the styles of domains.

## 2 RELATED WORKS

**The Unsupervised Domain Adaptation (UDA)** techniques bridge the domain gap through various means like divergence minimization, invariant feature learning via adversarial neural network training. Techniques for minimizing the divergence include Correlation Alignment (CORAL)[19], Maximum Mean Discrepancy (MMD)[5], Contrastive Domain Discrepancy (CDD)[10], etc. These methods target the statistics of the source and the target distribution, in order to minimize the domain gap between the two. Under adversarial network based methods[2, 12, 13], we pit two players in a zero sum game to optimize for domain gap minimization. The strategy used is similar to training of a vanilla GAN[4]. In RevGrad[2] a ResNet based feature extractor is used in combination with a domain discriminator and label classifier. The discriminator tries to differentiate between the source domain and target domain samples while classifier tries to identify correct class of the sample. A gradient reversal layer is used in the feature layer to obtain domain invariant features. DRCN[3] method uses reconstruction of image samples to apply domain adaptation. Through reconstruction, it tries to identify latent domain representation that can reconstruct both the target and source domain, along with encoding classification information in source domain.

**Image augmentation**. Another major tool used to tackle domain adaptation problem is sample image augmentation. Augmentation techniques like AdaIN[8], norm-VAE[23] learn functions to transfer style from target to source domain, in order to reduce the domain gap existing between the two. Such learning based techniques cost substantially in terms of time and computational power. Difficulty in the search of ideal hyperparameters to be used are another source of instability in the training process in such approaches. GAN based methods like CycleGAN pose substantial cost in terms of both time and computational power. Contrary to these, the proposed FDA based augmentation is a computationally cheap way to reduce the domain gap. The proposed triplet loss component also poses minimal additional computation cost.

## 3 PROPOSED METHOD

The proposed method StFT works on top of existing state of the art UDA methodologies. The proposed method follows 2 steps in an attempt to close down the domain gap between source and target domains. The first step involves swapping the low frequency component of the training source domain samples, with that of randomly sampled target domain samples. The second step involves application of an additional triplet loss component to the overall loss function of the underlying UDA algorithm.

## 3.1 Fourier Transformation

Since the low frequency component represents smooth features of an image, they help enable transfer of some style information from the target domain to the source domain. This helps bring the augmented source domain labelled samples closer to the target domain in the image space. Another purpose served by this Fourier transformation is to prepare component required for the second step. It does so by creating Fourier transformed source domain samples to be used as “anchor” in the triplet loss of second step.

For Fourier transformation, Fast Fourier transforms (FFT) for the Source domain and Target domain samples for the current batch are calculated. FFT algorithm efficiently implements the Discrete Fourier transform (DFT). DFT for two dimensional data is represented in equation (1) and its inverse representation is in equation (2) .

$$\mathcal{F} = F(p,q) = \frac{1}{RS}\sum_{r=0}^{R-1}\sum_{s=0}^{S-1} f[r,s]\exp\{-j2\pi(\frac{pr}{R}+\frac{qs}{S})\} \tag{1}$$

$$\mathcal{F}^{-1} = f(r,s) = \frac{1}{PQ}\sum_{p=0}^{P-1}\sum_{q=0}^{Q-1} F[p,q]\exp\{j2\pi(\frac{pr}{P}+\frac{qs}{Q})\} \tag{2}$$

Where $R$ and $S$ represent the number of rows and columns respectively, in the input 2-d image. $f[r,s]$ represents the pixel value at $r^{th}$ row and $s^{th}$ column in the input image. Likewise, the function $F[p,q]$ calculates the frequency value at $p^{th}$ row and $q^{th}$ column of the input image. $\mathcal{F}$ and $\mathcal{F}^{-1}$ are Fourier and inverse Fourier transform respectively.

To transfer the style specific information encoded in the low frequency component of the target domain, the FFT for both Source and Target domain samples are calculated. The low-frequency component of the Source domain samples is replaced with that of the Target domain samples as shown in Fig. 2(c). The transformed source image is called as Fourier transformed Source $\tilde{\mathcal{S}}$. It ensures a smaller domain shift between $\tilde{\mathcal{S}}$ and Target domain $\mathcal{T}$ in image space. Overall, Frequency transformation procedure can be summarized as follows:

$$\tilde{\mathcal{S}} = \mathcal{F}^{-1}[\tilde{\tau}\{\mathcal{F}(\mathcal{S}), \mathcal{F}(\mathcal{T})\}] \tag{3}$$

Where $\mathcal{F}$ and $\mathcal{F}^{-1}$ are Fourier transform and inverse Fourier transform respectively. Function $\tilde{\tau}$ represents the function to swap the low frequency component in the first argument, from the second argument. This is also demonstrated in Fig. 2(a)

## 3.2 Triplet Loss

In the second step, we apply an additional triplet loss along with the total overall loss of the underlying UDA method. Triplet loss is a loss function that compares current input sample, referred as the anchor, with a sample to be matched, called the positive, and with a sample to move away from, referred as the negative. The triplet loss tries to minimize the distance between the anchor and the positive, while it tries to maximize the distance between the anchor and the negative. If we are operating in euclidean space, triplet loss L can be defined as:

$$\mathcal{L}(p,a,n) = max(||F(a) - F(p)||^2 - ||F(a) - F(n)||^2 + \beta, 0) \tag{4}$$

where $p$ is the positive input, $a$ is the anchor input and $n$ is the negative input, function $F$ is the feature extractor used to generate embedding for an input image and $\beta$ is the margin to be enforced between positive sample and the negative sample. In our experiments, the $F$ will be the backbone of a ResNet 50 model.

The purpose of this loss is to add information encoding target style, to the Fourier transformed source samples. The effect of this step relies on the assumption that there are only style specific information and class specific information in any image data sample. In order to allow only style specific information to flow from target domain to the augmented source domain and in order to move augmented source domain style away from the source domain, we carry out randomization of samples in the original source domain samples batch, augmented source domain samples batch and to the unlabelled target domain samples batch. Randomization ensures that when triplet loss is applied over multiple sample triplets multiple epochs, the class specific information encoded in all three kind of samples get cancelled out, leaving the comparison only between the style specific components.

Let $\eta$ be a random shuffling function, $A$ be the set of training samples from Fourier Transformed source domain $\tilde{\mathcal{S}}$, acting as anchor in the loss function. $P$ be the set of positive target domain $\mathcal{T}$ and $N$ be the set of negative samples from original source domain $\mathcal{S}$ and $b$ be the batch size.

$$\mathcal{P} = \{p_1^{c_{p_1}}, p_2^{c_{p_2}}, ...., p_b^{c_{p_b}}\} \tag{5}$$

$$\mathcal{N} = \{n_1^{c_{n_1}}, n_2^{c_{n_2}}, ...., n_b^{c_{n_b}}\} \tag{6}$$

$$A = \{a_1^{c_{a_1}}, a_2^{c_{a_2}}, ..a_b^{c_{a_b}}\} = \eta(\mathcal{F}^{-1}[\tilde{\tau}\{\mathcal{F}(\mathcal{N}), \mathcal{F}(\mathcal{P})\}]) \tag{7}$$

Then the total triplet loss for each batch will become:

$$\mathcal{J} = \sum_{i=1}^{b} L(P_i, A_i, N_i) \tag{8}$$

As a result of this additional loss component in the overall loss, the ResNet generated feature used for classification encodes information about the target domain style as well. This results in further reduction in domain shift and a better UDA performance.

# 4 DOMAIN ADAPTATION

The Fourier transformation of source $\mathcal{S}$ brings the source domain closer to the target domain. As can be observed in fig1. the Fourier transformed samples are closer to the target domain, compared to the source domain, resulting in a reduction in the domain gap. This transformation only leads to a finite reduction in the domain gap between the source domain $\mathcal{S}$ and target domain $\mathcal{T}$. This transformation can also lead to development of artifacts in the image space during the inverse Fourier transformation, causing an imperfect style transfer. In order to counter this effect and to further supplement Fourier transformation, triplet loss is applied in the feature space. Triplet loss brings style information encoded in the feature of the augmented source domain $\tilde{\mathcal{S}}$ closer to that of the target domain $\mathcal{T}$ and away from that of the source domain samples $\mathcal{S}$. This also helps in further eliminating the small differences arising from the hardware device used to capture images in each domain.

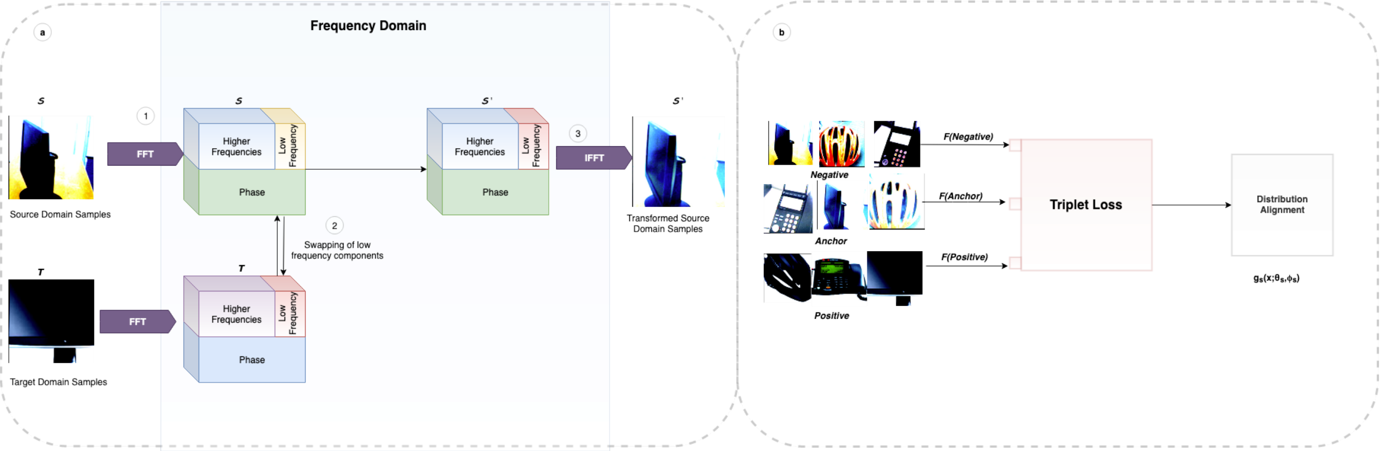


**Figure 2: Architecture of StFT. The architecture is divided into two stages. In Stage 1 the source images are transformed using low frequency component augmentation by swapping it with that of the target domain image samples. The augmented source domain is closer to the target domain. The Source $\mathcal{S}$, Transformed Source $\hat{\mathcal{S}}$ (with labels) and Target $\mathcal{T}$ images (unlabelled) are then passed through the underlying Domain Adaptation network. In stage 2, the randomly shuffled source domain samples (negative), augmented source domain samples (anchor) and the target domain samples (positive) are passed to the triplet loss function to bring augmented source closer to the target domain, in terms of the style.**

## 4.1 Algorithm

Algorithm of the proposed approach can be observed in Algorithm 1 with a graphical representation of the same in the fig.2

# 5 THEORETICAL INSIGHTS

Let H be a hypothesis space having VC dimension as $D$. Let $U_S$, $U_T$ be the labelled and unlabeled samples of size m each, drawn from the distributions $\mathcal{D}_S$ and $D_T$ respectively. Simplified form of theorem 2 from [1] can be written as

$$\epsilon_T(h) \leq \frac{1}{2}\hat{D}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_S, \mathcal{U}_T) + \mathcal{K} \tag{9}$$

where $\epsilon_T(h)$ is the classification error for Target domain samples for the given hypothesis $h$. $\hat{D}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_S, \mathcal{U}_T)$ is the divergence value between the target and source domains while $\mathcal{K}$ is a constant value. To obtain an even stricter upper bound, UDA algorithms should aim for a smaller divergence $\hat{D}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_S, \mathcal{U}_T)$. Domain adaptation algorithm that minimizes the domain divergence reduces the Target error $\epsilon_T(h)$. The method proposed in this work explicitly tries to minimize the divergence between the domains. It does so by swapping the low frequency component of source domain by that of target domain samples as well as applying a domain style specific triplet loss such that $\hat{D}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_{\hat{S}}, \mathcal{U}_T) < \hat{D}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_S, \mathcal{U}_T)$ where $\mathcal{U}_{\hat{S}}$ is unlabelled transformed source sample. Hence, the proposed method will have a stricter target classification error upper bound. Hence, applying domain adaptation algorithm on reduced divergence should further minimize the target classification error value $\epsilon_T(h)$ resulting in improved domain adaptation model performance.

# 6 EXPERIMENTS

The model agnostic nature of the proposed method makes it possible to use it to improve the performance of any existing method. In this work, the proposed method has been applied to two popular methods - Minimum Class Confusion (MCC)[9] and RevGrad using Office31 dataset. We compare the proposed method with existing UDA baseline using test accuracies on target domain samples. The loss values of training and testing of different methods has not been compared in this work owing to differences in loss functions used.

## 6.1 Dataset

**Office-31**: This dataset consists of 3 domains with samples corresponding to 31 classes, in each of the 3 domains. There are 4110 image samples in total. The 3 domains in the dataset are Amazon ($A$), DSLR ($D$) and Webcam ($W$). The Amazon dataset can be downloaded from amazon.com. It consists of objects corresponding to the 31 classes, against a white background. The images in DSLR and Webcam domain have been captured in an office setting. A major source of domain gap between the DSLR and Webcam domain is the difference in image resolution. The DSLR domain images have a higher resolution compared to the Webcam images. There are total 6 possible domain adaptation cases for Office31 dataset. These cases are $D \longrightarrow A$, $W \longrightarrow A$, $A \longrightarrow W$, $A \longrightarrow D$, $D \longrightarrow W$, and $W \longrightarrow D$. Since our focus is on ecommerce application of Domain Adaptation, the cases covered in this work are $D \longrightarrow A$, $W \longrightarrow A$, $A \longrightarrow W$, $A \longrightarrow D$.

## 6.2 Setup

Sagemaker instance ml.p3.2xlarge having 1 Tesla V100 has been used for all the experiments. ResNet 50 has been used as the feature extractor in all of the experiments. Minibatch gradient descent with 0.9 as the momentum has been used as the optimization algorithm. The learning rate used has been defined as:

$$\mu_p = \frac{\mu_0}{(1 + 9p)} \tag{10}$$

$$p = \frac{epoch}{total\ number\ of\ epochs} \tag{11}$$

**Algorithm 1:** StFT: Style loss and Fourier Transformation for Domain Gap Reduction

```
Stage 1: Frequency Transformation
Input: $\{x_i^s, y_i^s\}_{i=1}^B \sim \mathcal{D}_s$ ▷ Randomly sample a batch of source image
$\{x_i^t\}_{i=1}^B \sim \mathcal{D}_t$ ▷ Randomly sample a batch of Target domain
Find the Fourier transform of source sample batch: $\mathcal{F}(\mathcal{S}) = \mathcal{F}(x_i^s)$
Find Fourier transform of target sample batch: $\mathcal{F}(\mathcal{T}) = \mathcal{F}(x_i^t)$
Swap source low frequency with that of target: $\tilde{f}\{\mathcal{F}(\mathcal{S}), \mathcal{F}(\mathcal{T})\}$ ▷ Figure 2(a)
Apply inverse Fourier Transform and extract the transformed source: $\hat{\mathcal{S}} = \mathcal{F}^{-1}[\tilde{f}\{\mathcal{F}(\mathcal{S}), \mathcal{F}(\mathcal{T})\}]$ ▷ Refer Eq. 3
Shuffle the order of transformed source images in $\tilde{\mathcal{S}}$
Stage 2: Domain Adaptation
Input: original Source $\mathcal{S}$; Transformed source $\hat{\mathcal{S}}$; Unlabelled Target $\mathcal{T}$; Adaptation Network Parameter $g_s(x; \theta_s, \phi_s)$
while epoch < $E^{max}$ do
    Triplet Loss $\mathcal{J} = \sum L(\mathcal{T}, \hat{\mathcal{S}}, \mathcal{S})$ ▷ Refer Eq. 8
    Update UDA model network parameters $\theta_s, \phi_s$ using adaptation loss + Style Specific Triplet loss $\mathcal{J}$
end
Repeat Stage-1 to 2 for all epochs with updated model parameter
Inference Using Trained Model
$\{x_i\}_{i=1}^B \sim \mathcal{D}_t$ ▷ Sample a batch of target images for inference
$\{y_i^{infer}\}_{i=1}^B = argmax\{g_t(x_i^t; \theta_s, \phi_s)\}_{i=1}^B$ ▷ obtain the target label predicted by the model
```

where $p \in [0, 1]$ and $\mu_0$ as 0.01.

The square window size for selecting low-frequency component is $2L * \min(Width, Height)$ and $L$ was chosen to be 0.01 for RevGrad and 0.005 for MCC. The margin value used for the triplet loss was 0. The model's test accuracy for target domain prediction has been used for evaluating the proposed approach.

## 6.3 Results

**MCC**: As can be observed from the experiments in table 1 the proposed method leads to additional performance in each of the four cases concerning adaptation from or to **A**. The overall improvement in the aggregate accuracy was **1.3%**. In Fig. 3 it can be observed that upon convergence, the differences in performance are maintained for further iterations as well. In Fig 6 a better alignment between source and target domain features can be observed in proposed method compared to the existing. This can be quantified via A Distance which is 1.67 in case of the proposed method compared to 1.87 in case of existing method. This helps us to infer a better domain adaptation taking place between the source and target domains when the proposed method is used.

**RevGrad**: As can be observed from the experiments in table 1 the proposed method leads to additional performance in each of the four cases concerning adaptation from or to **A**. The overall improvement in the aggregate accuracy was **1.4%**. In Fig. 4 it can be observed that upon convergence, the differences in performance are maintained for further iterations as well. In Fig 5 a better alignment between source and target domain features can be observed in proposed method compared to the existing. This can be quantified via A Distance which is 1.23 in case of the proposed method compared to 1.63 in case of existing method. This helps us to infer that the proposed method leads to a better domain adaptation between the source and target domains.

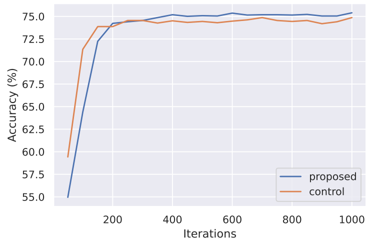


**Figure 3: Accuracy Comparison for $W \rightarrow A$ for MCC As can be observed, as the model training converges, the proposed solution outperforms the underlying UDA algorithm**

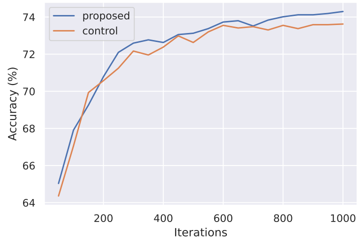


**Figure 4: Accuracy Comparison for $W \rightarrow A$ for RevGrad As can be observed, as the model training converges, the proposed solution outperforms the underlying UDA algorithm**

## 7 E-COMMERCE APPLICATION

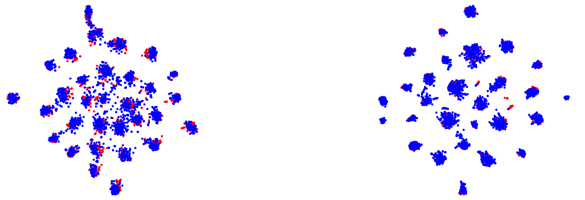
The improvement due to the proposed approach can also have positive implications for ecommerce application of Domain Adaptation algorithms. Improvement in image based match between images captured by customers and selections available with ecommerce stores can be one of such application (W ⟶ A or D⟶ A) . Such

**Table 1: Accuracy (%) on Office-31 for unsupervised domain adaptation (ResNet50).**

| Method | A → W | A → D | D → A | W → A | Avg |
|---|---|---|---|---|---|
| ResNet [6] | 68.4 | 68.9 | 62.5 | 60.7 | 65.1 |
| DDC [22] | 75.8 | 77.5 | 67.4 | 64.0 | 71.1 |
| DAN [14] | 83.8 | 78.4 | 66.7 | 62.7 | 72.9 |
| ADDA [21] | 86.2 | 77.8 | 69.5 | 68.9 | 75.6 |
| JAN [16] | 85.4 | 84.7 | 68.6 | 70.0 | 77.1 |
| MADA [17] | 90.0 | 87.8 | 70.3 | 66.4 | 78.6 |
| GTA [18] | 89.5 | 87.7 | 72.8 | 71.4 | 80.3 |
| CAN [25] | 81.5 | 85.5 | 65.9 | 63.4 | 74.0 |
| iCAN [25] | 92.5 | 90.1 | 72.1 | 69.9 | 84.1 |
| CDAN [15] | 93.1 | 89.8 | 70.1 | 68.0 | 80.2 |
| CDAN+E [15] | **94.1** | 92.9 | 71.0 | 69.3 | 81.8 |
| DSAN [28] | 93.6 | 90.2 | 73.5 | 74.8 | 83.0 |
| MCC [9] | 93.1 | 91.9 | 75.3 | 74.9 | 83.8 |
| MCC+StFT (proposed) | 93.6 | **93.7** | **77.8** | **75.4** | **85.1** |

**Table 2: Accuracy on Office-31**

| Method | $D \rightarrow A$ | $W \rightarrow A$ | $A \rightarrow D$ | $A \rightarrow W$ | Avg |
|---|---|---|---|---|---|
| ResNet [6] | 62.5 | 60.7 | 68.9 | 68.4 | 65.1 |
| RevGrad [2] | 71.5 | 73.6 | 84.3 | 88.9 | 79.5 |
| RevGrad+StFT (proposed) | **73.1** | **74.3** | **84.9** | **91.3** | **80.9** |

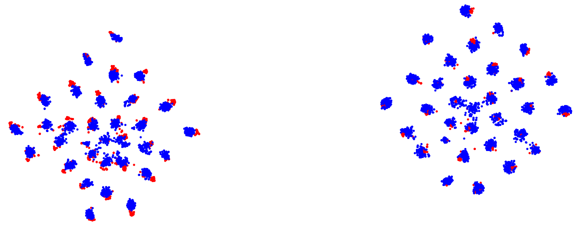


**(a) T-SNE Plot for features generated by RevGrad for $W \rightarrow A$ adaptation. A-Dist = 1.6318**

**(b) T-SNE Plot for features generated by RevGrad with StFT for $W \rightarrow A$ adaptation. A-Dist = 1.2357**

**Figure 5: Dimensionally Reduced T-SNE plots for RevGrad Features generated in Experiment and Control. As can be observed from (a) and (b), features in experiment are more aligned compared to control**

improvement can also form the basis of image based verification of correct product items being delivered by ecommerce stores to their customers (A → W or A→ D). Another innovative application can be mapping of products to those trending on social media or those available with local shops. Such maps can form basis of Advertisement content and product recommendations.



**(a) T-SNE Plot for features generated by MCC for $W \rightarrow A$ adaptation. A-Dist = 1.8778**

**(b) T-SNE Plot for features generated by MCC with StFT for $W \rightarrow A$ adaptation. A-Dist = 1.6722**

**Figure 6: Dimensionally Reduced T-SNE plots for MCC Features generated in Experiment and Control. As can be observed from (a) and (b), features in experiment are more aligned compared to control**

## 8 CONCLUSION

In this work the domain gap has been explicitly reduced in image space using low frequency swapping and in feature space via style specific triplet loss. We have measured the effectiveness of the proposed method using publicly available Office31 dataset consisting of samples from 3 domains. Apart from stating the theoretical insights, we have also experimentally evaluated our proposed approach. The proposed method yields better or comparable results against the existing unsupervised domain adaptation baseline methods. StFT is independent of the underlying UDA algorithm and works both before as well as during the adaptation steps. It can be easily used with any existing UDA method to enhance their current performance. Unlike GAN based style transfer UDA algorithms the proposed method step is relatively computationally inexpensive and doesn't consists of many trainable hyperparameters.

**Limitations** The Fourier transformation stage consisting of low frequency component swap in the source domain images can lead to loss of class discriminative features in the source domain. Also, the proposed method doesn't account for differences in the perspective of image samples from different domains.

**Future Works** For the future work, we will investigate the effectiveness of the proposed approach for other tasks such as semantic segmentation, keypoint recognition and image regression. Further work can be done on the robustness and scalability of the proposed method across different cases of domain adaptation such as multi-source domain adaptation and multi-target domain adaptation.

## REFERENCES

[1] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. *Machine learning* 79, 1 (2010), 151–175.

[2] Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*. PMLR, 1180–1189.

[3] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, David Balduzzi, and Wen Li. 2016. Deep reconstruction-classification networks for unsupervised domain adaptation. In *European Conference on Computer Vision*. Springer, 597–613.

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems* 27 (2014), 2672–2680.

[5] Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex Smola. 2006. A kernel method for the two-sample-problem. *Advances in neural information processing systems* 19 (2006), 513–520.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[7] Elad Hoffer and Nir Ailon. 2015. Deep metric learning using triplet network. In *International workshop on similarity-based pattern recognition*. Springer, 84–92.

[8] Xun Huang and Serge Belongie. 2017. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*. 1501–1510.

[9] Ying Jin, Ximei Wang, Mingsheng Long, and Jianmin Wang. 2020. Minimum Class Confusion for Versatile Domain Adaptation.

[10] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. 2019. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4893–4902.

[11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (Eds.), Vol. 25. Curran Associates, Inc., 1097–1105. https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf

[12] Jogendra Nath Kundu, Nishank Lakkakula, and R Venkatesh Babu. 2019. Umadapt: Unsupervised multi-task adaptation using adversarial cross-task distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 1436–1445.

[13] Jogendra Nath Kundu, Phani Krishna Uppala, Anuj Pahuja, and R Venkatesh Babu. 2018. Adadepth: Unsupervised content congruent adaptation for depth estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2656–2665.

[14] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. 2015. Learning transferable features with deep adaptation networks. In *International conference on machine learning*. PMLR, 97–105.

[15] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. 2017. Conditional adversarial domain adaptation. *arXiv preprint arXiv:1705.10667* (2017).

[16] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. 2017. Deep transfer learning with joint adaptation networks. In *International conference on machine learning*. PMLR, 2208–2217.

[17] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. 2018. Multi-adversarial domain adaptation. In *Thirty-second AAAI conference on artificial intelligence*.

[18] Swami Sankaranarayanan, Yogesh Balaji, Carlos D Castillo, and Rama Chellappa. 2018. Generate to adapt: Aligning domains using generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 8503–8512.

[19] Baochen Sun, Jiashi Feng, and Kate Saenko. 2016. Return of frustratingly easy domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30.

[20] Baochen Sun and Kate Saenko. 2016. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*. Springer, 443–450.

[21] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. 2017. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7167–7176.

[22] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. 2014. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474* (2014).

[23] Qian Wang, Fanlin Meng, and Toby P Breckon. 2020. Data augmentation with norm-VAE for unsupervised domain adaptation. *arXiv preprint arXiv:2012.00848* (2020).

[24] Yanchao Yang and Stefano Soatto. 2020. Fda: Fourier domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4085–4095.

[25] Weichen Zhang, Wanli Ouyang, Wen Li, and Dong Xu. 2018. Collaborative and adversarial network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3801–3809.

[26] Youshan Zhang. 2021. A Survey of Unsupervised Domain Adaptation for Visual Recognition. *CoRR* abs/2112.06745 (2021). arXiv:2112.06745 https://arxiv.org/abs/2112.06745

[27] Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. 2019. On learning invariant representations for domain adaptation. In *International Conference on Machine Learning*. PMLR, 7523–7532.

[28] Yongchun Zhu, Fuzhen Zhuang, Jindong Wang, Guolin Ke, Jingwu Chen, Jiang Bian, Hui Xiong, and Qing He. 2020. Deep Subdomain Adaptation Network for Image Classification. *IEEE Transactions on Neural Networks and Learning Systems* (2020).