Multidomain Adaptive Semantic Communications

Dongwook Won, Quang Tuan Do, Thwe Thwe Win, Donghyun Lee,
Junsuk Oh and Sungrae Cho, *Memeber, IEEE*

***Abstract*—The domain adaptation issues in semantic communications become critical when transmitter and receiver operate across different multiple domains or when input data during inference have different distributional characteristics than the data used to train semantic encoders and decoders. In this paper, we introduce the Multi-Domain Adaptive Deep Semantic Communication (MA-DeepSC) framework, designed to enhance semantic communications across multiple domains. Our framework consists of two core components: the Multi-Domain Adaptive Semantic Coding Network (MASCN), inherently designed to adapt semantic encoding and decoding across multiple domains, and the multidomain data adaptation network (MDAN), which transforms actual observable data into the data on which the system was initially trained, thus obviating the need for retraining the existing pre-trained semantic coding network. We validate our approach through experiments on digit datasets and CelebA, observing significant outperformance over existing techniques. In addition, we analyze the strategic benefits and drawbacks of both MASC and MDAN, assessing their applicability under various scenarios. The source code for MA-DeepSC is available at https://github.com/wongdongwook/JSAC_MA-DeepSC**

***Index Terms*—Semantic communications, semantic coding, domain adaptation.**

I. Introduction

WITH the evolution of wireless communication and the development of deep learning, a wide range of intelligent applications has emerged, demanding extensive connectivity within the limited spectrum of wireless resources. This poses significant challenges for Shannon-based traditional communication systems [1]. The rise of semantic communication offers a promising solution to this challenge. Semantic communications are characterized by being content-aware, task-oriented, and semantic-related, facilitating the extraction of only useful information from a large amount of data for delivery to designated destinations [2]. Unlike conventional methods that transmit all data regardless of semantic relevance for task, semantic communication significantly enhances system performance and spectrum efficiency [3]. Recognized for its potential to transform the landscape of future networks, including the upcoming 6G networks, semantic communication is anticipated to play a pivotal role in the advancement of next-generation network technologies. By enabling more efficient and context-aware data transmission, it holds the promise of significantly enhancing the responsiveness and reliability of critical applications such as autonomous driving, unmanned aerial vehicles and emergency medical services. Employing deep learning (DL), various practical semantic communication systems have been designed for the transmission of text [4], vocal speech [5], image [6], and video [7], significantly enhancing the reliability and effectiveness of communication compared to traditional systems under the same transmission rates.

However, domain adaptation poses significant challenges in semantic communication systems. The term ‘domain’ refers to a specific set of data distributions associated with particular categories under which data is collected. The semantic coding network, consisting of a semantic encoder and decoder, is typically trained using dataset from specific domains, which can lead to the Out of Distribution (OoD). The OoD problem arises when the data distribution encountered during inference differs significantly from the data distribution utilized to train the semantic coding network. Xie *et al*. [4] addressed the OoD by adapting data through transfer learning techniques, enabling the utilization of data from varying distributions encountered during semantic inference. Additionally, Zhang *et al*. [8] introduced a CycleGAN-based data adaptation network. This network transforms the data in the source domains to match the data in the observed domain during inference, facilitating domain adaptation without the need to retrain the entire semantic coding network. Only the data adaptation network is trained, ensuring efficiency without significantly degrading the performance of the pragmatic task. However, the data adaptation network proposed by Zhang *et al*. primarily focus on single

D. Won, D. Q. Tuan, T. T. Win, D. Lee, J. Oh, and S. Cho are with the School of Computer Science and Engineering, Chung-Ang University, Seoul 06974, South Korea (E-mail: dwwon@uclab.re.kr; dqtuan@uclab.re.kr; ttwin@uclab.re.kr; dhlee@uclab.re.kr; jsoh@uclab.re.kr; srcho@cau.ac.kr).

domain adaptation, faces scalability limitations when dealing with multidomain adaptation issues. To learn all mapping among k domains, $k(k-1)$ generators have to be trained for the accommodation of multiple k domains. Therefore, there is a pressing need for a scalable data adaptation network design from a multidomain adaptation perspective.

Furthermore, the implementation of data adaptation network precedes that of existing semantic coding network, thereby imposing substantial computational overhead and slowing down operational speed. Consequently, data adaptation network provide a short-term solution that lacks sustainability over long periods or temporary solution for frequent domain change of data source. From a long-term perspective, it is advantageous to design semantic coding networks that inherently adapt multiple domains because it allows for fast and efficient adaptation to new domains without the extensive overhead associated with existing data adaptation network. However, so far there has been no research paper dealed with the design of multidomain adaptive semantic coding network.

The unanswered questions in the context of semantic communication with multidomain adaptation issue include:

- Q1. How can the data adaptation network be designed to efficiently handle scalable adaptation of dynamic source data across multiple domains?
- Q2. How can the semantic coding network effectively extract and transmit semantic information behind the bits across multiple domains?

Our research contributes to the advancement of semantic communication systems with a focus on multidomain adaptation, offering solutions to significant challenges:

- For answering Q1, we propose a multidomain data adaptation network (MDAN) applying a StarGAN [9]. StarGAN addresses scalability issues by requiring only one generator for adaptation across multiple domains, rather than multiple generators required by existing method [8]. By employing a single generator capable of handling multiple domains, our approach significantly enhances scalability and operational efficiency.
- For answering Q2, we propose a StarGAN-based Multidomain Adaptive Semantic Coding Network (MASCN) that inherently supports adaptation across multiple domains. We have developed a semantic distortion function in a multidomain context. Utilizing the proposed training algorithm, MASCN effectively minimizes semantic distortion, enabling accurate extraction and transmission of semantic information across multidomain.
- Additionally, our research provides a comparative analysis of MASCN and MDAN, specifically highlighting in which situations each approach is preferable. This analysis helps identify the distinct strengths and weaknesses of both strategies within the context of multidomain adaptation. Our insights are aimed at guiding the development of more tailored and effective solutions for multidomain adaptive semantic communication systems.

II. Related Works

1) Domain Adaptation in DL: Domain adaptation techniques in DL have evolved significantly, catering to both single and multidomain challenges. Early works primarily focused on leveraging data from the target domain directly, assuming a direct correspondence between source and target domain samples. Recent advancements have introduced generative models like Generative Adversarial Networks (GANs) that enhance the capability of domain adaptation by generating indistinguishable fake samples from real ones [10]. A notable example includes the Pix2Pix framework by Isola *et al.*, which utilizes conditional GANs to facilitate effective image-to-image translation tasks under paired sample conditions [11]. In contrast, models like CycleGAN, introduced by Zhu *et al.*, and similar frameworks like DTN by Taigman *et al.* and DiscoGAN by Kim *et al.*, have pioneered the unpaired image-to-image translation domain [12]–[14]. These models do not rely on paired samples but use dual generators to enable bidirectional translation across domains. Thus, above papers are not scalable to the increasing number of domains. For addressing the scalability issues inherent in previous models, StarGAN represents a significant leap forward by employing a single generator to map between all available domains using a single generator in [9]. This approach not only simplifies the model architecture but also increases efficiency, making it possible to scale up to a larger number of domains without a proportional increase in complexity.

2) Domain Adaptation in Semantic Communication: In the realm of semantic communication, domain adaptation is a critical area of research, with a focus predominantly on single-domain adaptation methodologies [8]. Semantic communication systems generally operate in two distinct phases: the preparation and working stages [4], [8]. During the preparation stage, the transmitter and receiver collaboratively train the semantic coding network by leveraging a shared background knowledge library, which includes empirical data and associated semantic information. This foundational training prepares the system to handle data similar to what was encountered during this initial phase. However, during the working stage, challenges arise when the actual observable data at the transmitter diverges in distribution from the empirical data. This disparity can lead to the OoD problem, where a model trained on one dataset (e.g., MNIST) may perform poorly on another with a different distribution (e.g., SVHN). To address this issue, Zhang *et al.* introduced a CycleGAN-based data adaptation network designed to convert observed data into a form similar to the empirical data, enabling the semantic coding network to function effectively without the need for retraining [8]. However, this solution is limited in scalability and robustness, as it requires separate data adaptation networks for each pair of domains, complicating the adaptation process and increasing resource demands.

III. Proposed Framework

Our multidomain adaptive deep learning-enabled semantic communication (MA-DeepSC) framework focuses on addressing the challenges of multidomain adaptation. As shown in Fig. 1, our MA-DeepSC framework consists of MASCN and MDAN. In this section, we will formally present our proposed MA-DeepSC framework: Section III-A will detail the MASCN, and Section III-B will discuss the MDAN.

A. Multidomain Adaptive Semantic Coding Network

MASCN is based on the a SNR (Signal-to-Noise Ratio) adaptive DeepJSCC system for a point-to-point image transmission scenario with SNR feedback [15], [16]. In this system, the SNR, denoted as $\gamma \in \mathbb{R}$, can be estimated at the joint semantic-channel decoder of the receiver and then fed back to the joint semantic-channel encoder of the transmitter. As a result, the channel SNR is known at both ends of the semantic communication system. The system model for the semantic coding network comprises a trainable semantic encoder f_θ, a non-trainable physical channel, and a trainable semantic decoder g_ϕ, where θ and ϕ denote the parameters of the semantic encoder and decoder, respectively.

1) Semantic Transmitter: The semantic transmitter processes domain-specific image data, which is represented as an n-dimensional vector $\mathbf{x}_d \in \mathbb{R}^n$, where $n = \mathrm{H} \times \mathrm{W} \times \mathrm{C}$ for image dimensions—height H, width W, and depth C. The domain label $d \in \mathcal{D}$ indicates the specific domain with domain set $\mathcal{D}$ to which $\mathbf{x}$ belongs, represented within a real number to represent a variety of domains. The domain information is encoded using a label, typically a one-hot vector for categorical attributes. The semantic encoder f_θ receives image data $\mathbf{x}_d$ and the target domain label $c \in \mathcal{D}$, transforming into the input image data corresponding to the specific target domain, ensuring the encoder to adapt across multiple domains. This adaptability is crucial for applications that require robust performance in various domains. The output of the semantic encoder, denoted by $\mathbf{z}_c \in \mathbb{C}^k$, comprises a vector of semantic codes. The semantic encoding function, $f_\theta(\mathbf{x}_d, \gamma, r, c)$, prioritizes features based on their semantic importance, especially under constraints such as the compression ratio (CR), where a lower CR signifies a higher semantic priority. Consequently, under given network conditions such as SNR γ, CR r, the semantic encoder transmits the prioritized semantic features, with the highest priority of semantic features transmitted first, thus improving the efficiency and relevance of the information transmitted. The operation of multidomain adaptive semantic encoder is formalized as follows.

$$\mathbf{z}_c = f_\theta(\mathbf{x}_d, \gamma, r, c) \in \mathbb{C}^k \tag{1}$$

where $r = k/n$ denotes the CR of the input image data [17], [18]. The size of the image n denotes the source bandwidth, and the size of the channel input k is the channel bandwidth, with r being the bandwidth ratio. To satisfy the average power constraints at the transmitter, the semantic code $\mathbf{z}_c$ must satisfy the condition $\frac{1}{k}\mathbb{E}\left[\mathbf{z}_c^* \mathbf{z}_c\right] \leq P_{\mathbf{z}_c}$, where $\mathbf{z}_c^*$ denotes the conjunct transpose of $\mathbf{z}_c$, and $P_{\mathbf{z}_c}$ denotes the average power of $\mathbf{z}_c$.

2) Transmission Channel: The semantic code $\mathbf{z}_c$ is transmitted over a physical noisy channel represented by the function $\eta : \mathbb{C}^k \rightarrow \mathbb{C}^k$. An additive white Gaussian noise (AWGN) channel is considered in this work. In general, the SNR γ is determined by the noise power N_0^2, which is mainly caused by the natural noise and interference in the physical channel [18].

$$\gamma = 10 \log_{10} \frac{P_z}{N_0^2} \tag{2}$$

In general, a physical channel with a lower SNR indicates a higher physical noise power compared with the $\mathbf{z}_c$ power, and the quality of the final recovered result is lower than a higher SNR channel. After passing through the physical channel, the distorted semantic code $\hat{\mathbf{z}}_c \in \mathbb{C}^k$ by AWGN ω is transmitted to the receiver, as given by

$$\hat{\mathbf{z}}_c = \eta(\mathbf{z}_c) = \mathbf{z}_c + \omega \tag{3}$$

where the vector $\omega \in \mathbb{C}^k$ consists of independent and identically distributed samples with the distribution $\mathcal{CN}(0, N_0^2\mathbf{I})$. In addition, $\mathcal{CN}(\cdot,\cdot)$ denotes a circularly symmetric complex Gaussian distribution.

3) Semantic Receiver: On the semantic receiver side, the semantic decoder g_ϕ plays a role in reconstructing the image from transmitted semantic features, which are

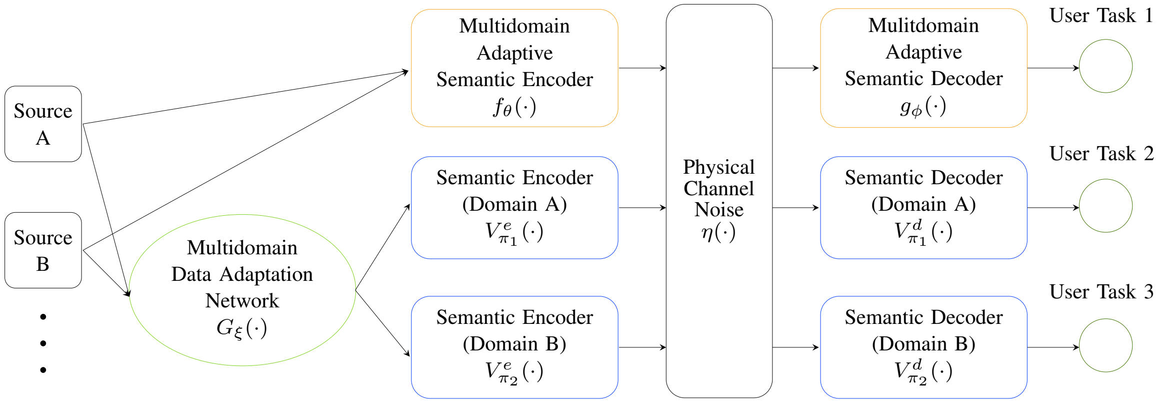

Source A
Source B
Multidomain Data Adaptation Network $G_\xi(\cdot)$
Multidomain Adaptive Semantic Encoder $f_\theta(\cdot)$
Semantic Encoder (Domain A) $V^e_{\pi_1}(\cdot)$
Semantic Encoder (Domain B) $V^e_{\pi_2}(\cdot)$
Physical Channel Noise $\eta(\cdot)$
Mulitdomain Adaptive Semantic Decoder $g_\phi(\cdot)$
Semantic Decoder (Domain A) $V^d_{\pi_1}(\cdot)$
Semantic Decoder (Domain B) $V^d_{\pi_2}(\cdot)$
User Task 1
User Task 2
User Task 3

Fig. 1: Illustration of our MA-DeepSC Framework.

transformed into images corresponding to specific domain c. This transformation is defined by the mapping $\mathbb{C}^k \times \mathbb{R} \rightarrow \mathbb{R}^n$, effectively processing the input $\hat{\mathbf{z}}_c$, γ, and r as follows:

$$\hat{\mathbf{s}}_c = g_\phi(\hat{\mathbf{z}}_c, \gamma, r, c) = M_\alpha(\mathbf{x}_d, \gamma, r, c) \in \mathbb{R}^n \quad (4)$$

where $\hat{\mathbf{s}}_c \in \mathbb{R}^n$ represents the semantic feature vector outputted by the decoder. This decoded semantic feature vector encapsulates the reconstructed image at the semantic level. The entirety of the MASCN system, comprising the encoder, channel, and decoder, is represented as M_α. Here, α denotes the collective parameters ϕ and θ.

In the context of image transmission, potential pragmatic tasks could be image classification, segmentation, among others. For the sake of simplicity, we will assume that the primary pragmatic task is image classification and is modeled by a trainable function h_ψ, which is designed to process the transformed image $\hat{\mathbf{s}}_c$ to classify the image into predefined categories. The operation of classifier h_ψ is formalized as:

$$\hat{l} = h_\psi(\hat{\mathbf{s}}_c) \quad (5)$$

where, $\hat{\mathbf{s}}_c$, the reconstructed image, serves as the input to the classifier h_ψ function which categorizes the image $\hat{\mathbf{s}}_c$ and the output is the predicted label $\hat{l}$.

B. Multidomain Data Adaptation Network

The MDAN precedes the semantic coding network and align incoming source data with the domain of pre-trained semantic coding network. It employs a conversion function $G_\xi : \mathbb{C}^k \times \mathbb{R} \rightarrow \mathbb{R}^n$ to map original domain image $\mathbf{x}_n$ original domain $c \in \mathcal{D}$, target domain $d \in \mathcal{D}$ as follows:

$$\mathbf{x}_c = G_\xi(\mathbf{x}_d, c) \quad (6)$$

where $\mathbf{x}_c \in \mathbb{R}^n$ represents the image after conversion to the target domain, serving as the output of the domain data adaptation module. This transformation is vital for allowing the domain-specific components of the semantic encoder and decoder to process data effectively across diverse domains.

Subsequently, we utilizes the DeepJSCC-V [18] framework V_π as semantic coding network, which is designed with semantic encoder V^e_π and decoder V^d_π for specific domain. The data $\mathbf{x}_c$ serves as input to the DeepJSCC-V process V_π, which is formalized as:

$$\hat{\mathbf{s}}_c = V_\pi(\mathbf{x}_c) \quad (7)$$

where the output $\hat{\mathbf{s}}_c$ becomes the input for pragmatic tasks. This data adaptation approach not only enhances the system's flexibility and applicability across different domains but also ensures that the semantic communication framework remains robust and effective, even in the face of OoD challenges.

IV. Multidomain Adaptive Semantic Coding

In this section, we explore the MASCN, discussing the workflow and architecture in Section IV-A, the semantic distortion function in Section IV-B, problem formulation in Section IV-C, and the training algorithm in Section IV-D.

A. WorkFlow and Architecture

1) Components Description: Based on StarGAN, which can perform the multidomain translation of images, this paper designs a new semantic coding framework for multidomain adaptive semantic communication. As shown in Fig. 2, the framework consists of MASCN and discriminator and pragmatic task. The MASCN M_α is responsible for transforming images into images of different domains, composed of semantic

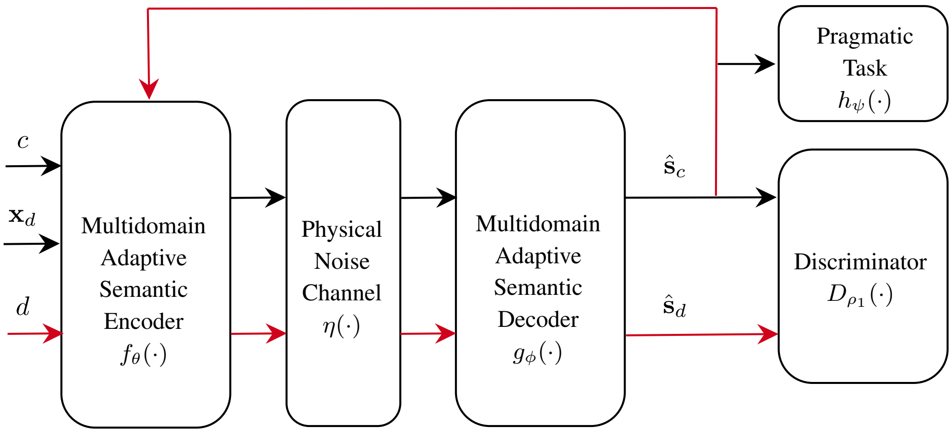


Fig. 2: Illustration of our MASCN framework.

encoder and decoder. The output of M_α is fed into the discriminator and pragmatic task. The discriminator, as denoted D_{ρ_1}, not only distinguishes between real and generated images but also classifies them according to domain labels. We assume that pragmatic task is iamge classification. The pragmatic task h_ψ classifies the reconstructed image.

2) Workflow Description: We divides the whole model of work flow into three parts: original-to-target domain, target-to-original domain, and discriminator domain. To guide the training of the MASCN framework, we employ a set of loss functions, including cycle consistency loss for image reconstruction [14], [19], domain label classification loss [20], and pragmatic classifier loss.

- Original-to-target domain: The input image $\mathbf{x}_d$ is associated with the target domain label c provided by the receiver and fed into M_α to generate $\hat{\mathbf{s}}_c$. The artificially generated fake image subsequently fed into a pragmatic task, where a pragmatic task loss is computed.
- Target-to-original domain: The fake image generated $\hat{\mathbf{s}}_c$ is combined with the original image label d and fed back into M_α to produce the translated image $M_\alpha(\hat{\mathbf{s}}_c, \gamma, r, d) = \hat{\mathbf{s}}_d$. We apply cycle consistency loss between $\hat{\mathbf{s}}_d$ and $\mathbf{x}_d$, and this reconstructed image is further subjected to a pragmatic task loss. The training process makes M generate images that are indistinguishable from the real images.
- Discriminator domain: The training process makes M_α generate images that are indistinguishable from the real images. For $\mathbf{x}_d$, the discriminator identifies it as real and classifies it as original image domain d. Conversely, for $\hat{\mathbf{s}}_c$, the discriminator identifies it as fake and classifies it as target domain c.

B. Semantic Distortion in Multidomain Adaptation

This section aims to minimize semantic distortion by extracting task-oriented semantic information acrross mutli domain.

1) Adversarial Loss: The adversarial loss function, as originally formulated in GANs, quantifies the indistinguishability of fake images from real ones [10]. StarGAN modifies this loss function by conditioning it on both the input image and the target domain label, allowing the generated images to closely replicate the attributes of the desired target domain. We further extend StarGAN's adversarial loss for a multidomain adaptive semantic communication distortion, which is defined according to the given compression ratio r.

$$\mathcal{L}_{adv} = \mathbb{E}_x \left[\log D_{src}(\mathbf{x})\right] + \mathbb{E}_{x,c} \left[\log\left(1 - D_{src}(\hat{\mathbf{s}}_c)\right)\right], \tag{8}$$

where M_α makes an image $\hat{\mathbf{s}}_c$ conditioned on both the input image $\mathbf{x}_d$ and the target domain label c, while discriminator D_{ρ_1} distinguishes between real and fake images. The discriminator D_{ρ_1} evaluates the output from the MASCN, calculating probability distributions over both the source data, domain labels and iamge lables, operating as $D_{\rho_1} : \mathbf{x}_d \rightarrow \{D_{src}(\mathbf{x}_d), D_{cls}(\mathbf{x}_d), D_{prg}(\mathbf{x}_d)\}$. It differentiates whether images are genuine $D_{src}(\mathbf{x}_d)$ or counterfeit $D_{cls}(\mathbf{x}_d)$, and categorizes the actual image labels $D_{prg}(\mathbf{x}_d)$. The image $\hat{\mathbf{s}}_c$ is the result of prioritized transmission based on the given compression ratio r from a multidomain perspective, ensuring that semantic-aware priorities are considered. The model M_α seeks to minimize $\mathcal{L}_{adv}$, whereas D_{ρ_1} strives to maximize it, thus creating a competitive environment that improves the quality of generated images.

2) Domain Classification Loss: While adversarial loss ensures the generated images are realistic, domain classification loss ensures that they are not only look real but also correctly represent the target domain. This aspect is crucial in controlling specific attributes like hair color, age, or style. Given any input image $\mathbf{x}_d$ and a target domain label c, the objective is to transform $\mathbf{x}_d$ into an output image $\hat{\mathbf{s}}_c$, which can be classified accurately to the target domain c. An auxiliary classifier on top of D_{ρ_1} to impose the domain classification loss when optimizing both D_{ρ_1} and M_α. Specifically, the objective is broken down into two terms: a domain classification loss for real images, used to optimize D, and a domain classification loss of fake images used to optimize M_α. In detail, the former is defined as

$$\mathcal{L}_{cls}^{r} = \mathbb{E}_{x,d}\left[-\log D_{cls}\left(d \mid \mathbf{x}_d\right)\right] \tag{9}$$

where the term $D_{cls}\left(d \mid \mathbf{x}_d\right)$ represents a probability distribution over domain labels computed by D_{ρ_1}. By minimizing this objective, D_{ρ_1} learns to classify a real image $\mathbf{x}_d$ to its corresponding original domain d. We assume that the input image and domain label pair $(\mathbf{x}_d, d)$ is given by the training data.

On the other hand, the loss function for the domain classification of fake images is defined as

$$\mathcal{L}_{cls}^{f} = \mathbb{E}_{x,c}\left[-\log D_{cls}(c \mid \hat{\mathbf{s}}_c)\right] \tag{10}$$

In other words, M_α tries to minimize this objective to generate images that can be classified as the target domain c.

3) Reconstruction Loss: By minimizing the adversarial and classification losses, M_α is trained to generate images that can be accurately classified into their correct target domains. However, minimizing these losses alone does not ensure that the translated images retain the content of the input images while modifying only the domain-specific features. To address this limitation, a cycle consistency loss function [12], [14] is incorporated into M_α to minimize semantic distortion. This loss function is designed to maintain the original content while altering only the domain-related aspects of the inputs. The reconstruction loss is defined as:

$$\mathcal{L}_{rec} = \mathbb{E}_{\mathbf{x}_d,c,d}\left[\|\mathbf{x}_d - \hat{\mathbf{s}}_d\|_1\right] \tag{11}$$

where M_α takes in the translated image $\hat{\mathbf{s}}_c$ and the original domain label d as input, attempting to reconstruct the original image $\mathbf{x}_d$. We adopt the L_1 norm to measure the reconstruction loss. M_α is used twice: first to translate the original image into an image in the target domain, and then to reconstruct the original image from the translated image. This process ensures that the reconstructed image $\hat{\mathbf{s}}_d$ remains semantically consistent with the original input $\mathbf{x}_d$.

4) Pragmatic Task Loss: In contrast to StarGAN's original approach, the pragmatic task loss is incorporated to ensure accurate classification of images within specific target domains. This involves both MASCN and discriminator working together to produce realistic, semantically accurate images while correctly identifying these generated images and their associated labels. This strategy is particularly useful for generating task-oriented images that retain key features of the original domain.

The discriminator is tasked with correctly classifying real images to their original labels. The pragmatic classifcation loss for discrimiantor penalizes the discriminator when real images are misclassified:

$$L_{prg}^{D} = -\mathbb{E}_{x,l}[\log h_\psi(l|\mathbf{x}_d)] \tag{12}$$

The MASCN is penalized when it fails to generate images that match the target labels. The pragmatic classification loss for MASCN is defined as:

$$L_{prg}^{M} = -\mathbb{E}_{x,l}[\log h_\psi(l|\hat{\mathbf{s}}_c)] \tag{13}$$

By utilizing these pragmatic task losses into discriminator and MASCN, the model can enhance the accuracy of image classification within semantic communication, leading to a more effective transmission of task-relevant features across different domains.

C. Problem Formulation

The optimization strategy for both the MASCN and the discriminator involves minimizing the generator's loss while maximizing that of the discriminator. The loss functions for the discriminator $\mathcal{L}_{D_{\rho_1}}$ and MASCN $\mathcal{L}_{M_\alpha}$ are as follows:

$$\mathcal{L}_{D_{\rho_1}} = -\mathcal{L}_{adv} + \lambda_{cls}\mathcal{L}_{cls}^{r} + +\lambda_{prg}L_{prg}^{r} \tag{14}$$

$$\mathcal{L}_{M_\alpha} = \mathcal{L}_{adv} + \lambda_{cls}\mathcal{L}_{cls}^{f} + \lambda_{rec}\mathcal{L}_{rec} + \lambda_{prg}L_{prg}^{f} \tag{15}$$

where the coefficients λ_{cls}, λ_{rec}, and λ_{acc} are hyperparameters that balance the relative importance of domain classification, reconstruction losses, and the pragmatic task loss, respectively. These settings ensure a focused optimization on not just creating visually plausible images but also on enhancing their domain-specific accuracy, content fidelity and performance of pragmatic task.

The problem formulation is structured as follows, considering given SNR γ and CR r constraints:

$$\begin{aligned} \min_{M_\alpha} \max_{D_{\rho_1}} \quad & \mathcal{L}_{D_{\rho_1}} + \mathcal{L}_{M} \\ \text{s.t.} \quad & CR = r \\ & SNR = \gamma \end{aligned} \tag{16}$$

This constraint-based optimization ensures that the system adheres to a specified compression ratio CR reflecting the necessary balance between image quality and bandwidth efficiency. The inclusion of the compression ratio directly impacts how the semantic content is prioritized during transmission, making it a critical factor in the system's overall performance.

D. Training Algorithm

The training algorithm for the MASCN model begins by initializing a training batch $\mathcal{T}$, which contains real images, their original labels, and SNR and CR. In each iteration, real images and labels are fetched, and these labels are shuffled to create target domain labels for the MASCN. A random SNR γ and CR r are generated to simulate real-world variations. The discriminator is trained first. It evaluates real images, computing classification losses for domain labels. The MASCN creates fake images with randomly generated target labels, γ, and r, which are used to compute adversarial and pragmatic losses. The adversarial loss ensures the discriminator cannot distinguish between real and generated images, while the pragmatic loss measures its ability to classify labels accurately. The MASCN is trained periodically. It generates fake images using target labels, γ, and r. Adversarial loss is calculated to evaluate whether the MASCN can fool the discriminator. Domain classification loss checks if the MASCN can accurately recreate domain-specific features, and reconstruction loss ensures semantic consistency. The pargmatic task loss and total MASCN loss is calculated and the MASCN's parameters are updated. This comprehensive training approach ensures that both the discriminator and MASCN are optimized concurrently, enabling the MASCN system to handle multidomain image translation tasks efficiently.

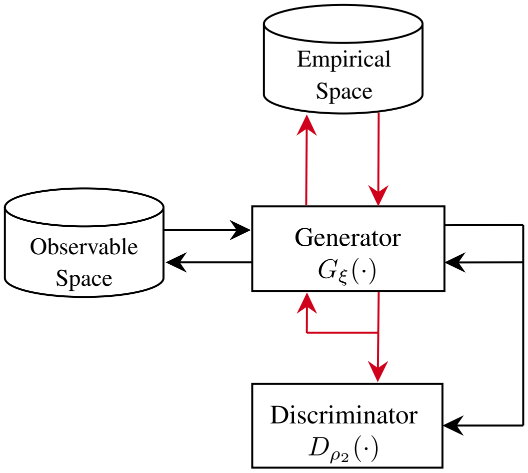


Fig. 3: Illustration of our MDAN framework.

Algorithm 1 Training Algorithm for MASCN

```
Input: Training batch $\mathcal{T}$, learning rate $\eta$;
Output: Updated model parameters $(\alpha, \rho_1)$ $D_{\rho_1}$;
for each iteration over the training batch $\mathcal{T}$ do
    Fetch real images and original labels from $\mathcal{T}$;
    Shuffle labels to create target domain labels;
    Randomly generate channel SNR $\gamma \in U(0, 27)$;
    Randomly generate CR $r \in U(0.1, 1)$;
    Calculate classification loss by (9);
    Generate fake images using $M_\alpha$ with target labels, $\gamma$, and $r$;
    Calculate adversarial loss by (8);
    Classifies fake image by eq. (5);
    Calculate pragmatic task loss by (12);
    Compute total discriminator loss by (14);
    Update $D_{\rho_1}$ using its optimizer;
    if current step % training frequency == 0 then
        Generate fake images using $M_\alpha$ with target labels, $\gamma$, and $r$;
        Calculate adversarial loss by (8);
        Calculate classification loss by (10);
        Generate reconstructed images with original domain labels, $\gamma$, and $r$;
        Calculate reconstruction loss by (11);
        Compute pragmatic task loss by (13);
        Calculate total MASCN loss by (15);
        Update $M_\alpha$ using its optimizer;
    end if
end for
```

V. Multidomain Data Adaptation

In this section V, we discuss the architecture, the loss functions of MDAN in section V-A, and the training algorithm in section V-B.

A. Multidomain Data Adaptation Architecture

The architecture of our MDAN network, illustrated in Fig. 3, is composed of the entire training data space, which is divided into two key data spaces: the empirical space and the observable space. The empirical space houses data that has been previously learned and serves as the foundation of the trained model's knowledge base. The observable space represents new data distributions encountered in real-world scenarios. The primary objective of the MDAN is to adapt new data to align with the empirical data space, ensuring the network maintains consistent performance across different environments. The network's primary training processes involve transforming data from the empirical space to fit the observable space, and vice versa. This bidirectional transformation is crucial for the network's ability to effectively manage and accommodate new information and changes within its operating environment.

The multidomain data adaptation network consists of a generator G_ξ and a discriminator D_{ρ_2}, which are trained concurrently in adversarial ways. The generator G_ξ receives input data $\mathbf{x}_d$, along with a target domain label c, then generates an output $G_\xi(\mathbf{x}_d, c)$. This output represents the data as it would appear in the target domain, effectively transforming the observable data to align with the characteristics of the target domain. The discriminator D_{ρ_2} evaluates the output of the generator, calculating probability distributions over both the source data and domain labels, operating as $D_{\rho_2} : \mathbf{x}_d \to \{D_{src}(\mathbf{x}_d), D_{cls}(\mathbf{x}_d))\}$. It differentiates whether images are genuine $D_{src}(\mathbf{x}_d)$ or counterfeit $D_{cls}(\mathbf{x}_d)$, and categorizes the actual image labels $D_{prg}(\mathbf{x}_d)$. The adversarial loss, defined as follows, refines the generator's ability to create images indistinguishable from genuine ones:

$$\begin{aligned}\mathcal{L}_{adv} &= \mathbb{E}_{\mathbf{x}_d}\left[\log D_{src}(\mathbf{x}_d)\right] \\ &\quad + \mathbb{E}_{\mathbf{x}_d, c}\left[\log\left(1 - D_{src}(G_\xi(\mathbf{x}_d, c))\right)\right] \qquad (17)\end{aligned}$$

where $\hat{x}_d$ represents the straight line uniform sample between original image and generated image. Secondly, the classification loss of the discriminator on the original dataset and the generated dataset are defined as follows

$$\mathcal{L}_{cls}^{r} = \mathbb{E}_{\mathbf{x}_d, d}\left[-\log D_{cls}\left(d \mid \mathbf{x}_d\right)\right] \qquad (18)$$

$$\mathcal{L}_{cls}^{f} = \mathbb{E}_{\mathbf{x}_c, c}\left[-\log D_{cls}(c \mid G_\xi(\mathbf{x}_d, c))\right] \qquad (19)$$

The reconstruction loss measures the fidelity of the reconstructed images to the original ones, ensuring consistency across multidomain transformations:

$$\mathcal{L}_{rec} = \mathbb{E}_{\mathbf{x}_d, c, d}\left[\left\|\mathbf{x}_d - G_\xi(G_\xi(\mathbf{x}_d, c), d)\right\|_1\right], \qquad (20)$$

Finally, the overall objective functions to optimize G_ξ and D_{ρ_2} are formulated as follows:

$$\mathcal{L}_{D_{\rho_2}} = -\mathcal{L}_{adv} + \lambda_{cls}\mathcal{L}_{cls}^{r} \qquad (21)$$

$$\mathcal{L}_{G_\xi} = \mathcal{L}_{adv} + \lambda_{cls}\mathcal{L}_{cls}^{f} + \lambda_{rec}\mathcal{L}_{rec} \qquad (22)$$

where λ_{cls} and λ_{rec} are important factors in controlling domain classification loss and reconstruction loss. In this min-max optimization, the generator strives to minimize its objective by producing convincing, domain-appropriate images, while the discriminator aims to maximize its own objective by distinguishing between real and generated images. This adversarial process ultimately improves the generator's capability to create authentic and semantically accurate images.

B. Training Algorithm

The training algorithm for MDAN optimizes the generator G_ξ and discriminator D_{ρ_2} within a StarGAN-based architecture. Starting with a training batch $\mathcal{T}$ and a learning rate η, the algorithm refines the parameters (ξ, ρ_2) for the generator and discriminator. Each iteration begins with extracting real images and their original labels from $\mathcal{T}$. The labels are shuffled to create target labels for domain translation. The discriminator is first trained by calculating the classification loss for real images, generating fake images using G_ξ, and computing adversarial loss and gradient penalty using a mix of real and fake images. The total discriminator loss is then used to update D_{ρ_2}. The generator G_ξ is trained less frequently. It regenerates fake images to recalibrate adversarial and classification losses, and generates reconstructed images to maintain cycle consistency. The generator's total loss, which includes adversarial, classification, and reconstruction losses, guides the parameter updates for G_ξ.

Algorithm 2 Training Algorithm for MDAN

```
Input: Training batch $\mathcal{T}$, learning rate $\eta$;
Output: Updated model parameters $(\xi, \rho)$;
for each iteration over the training batch $\mathcal{T}$ do
    Fetch real images and original labels from $\mathcal{T}$;
    Shuffle labels to create target domain labels;
    Calculate classification loss by (18);
    Generate fake images using $G_\xi$ with target labels;
    Calculate adversarial loss by (17);
    Compute total discriminator loss by (21);
    Update discriminator $D_{\rho_2}$ using its optimizer;
    if current step % training frequency == 0 then
        Generate fake images $G_\xi$ with target labels;
        Calculate adversarial loss by (17);
        Calculate classification loss by (19);
        Generate reconstructed images;
        Calculate reconstruction loss by (20);
        Total generator loss by (22);
        Update generator $G_\xi$ using its optimizer;
    end if
end for
```

VI. Experiments

This section details our experimental setup and evaluations. Initially, we focus on the digit datasets to benchmark our method against recent techniques in image recognition tasks. Furthermore, we extend our comparisons to the domain of facial attribute transfer. All experiments are conducted using model outputs derived from images that were not seen during the training phase, ensuring that our results reflect the model's ability to generalize to new, unseen data.

A. Comparison Models

As our comparison models, we adopt the DeepJSCC-V [18] and CycleGAN based data adaptation (DA) network [8].

- DeepJSCC-V (Retrained): This model has no domain adaptation capacity and is retrained on domain faced with actual observable data distribution so that the overhead is significant and lack of scalability.
- CycleGAN (DA): Utilizes a CycleGAN-based approach for data adaptation (DA), transforming actual observable data from a new domain into the format of the originally trained domain without retraining the semantic coding network. This method leverages the existing DeepJSCC-V as a semantic coding network.
- DeepJSCC-V (no DA): This model is trained only on the original domain and does not undergo retraining to adapt to different actual observable data distributions, serving as a baseline to assess the necessity and impact of domain adaptation.

B. Dataset and Training

In our experiments, we used two main benchmark datasets: digit datasets and CelebA. The digit datasets include MNIST, MNIST-M, SYN, and USPS. These datasets present various challenges in domain adaptation and image recognition. MNIST consists of handwritten digits, MNIST-M has mixed backgrounds, SYN simulates real-world scenarios with different styles and distortions, and USPS features handwritten digits from U.S. Postal Service mail. All images are resized to 32×32 pixels for neural network processing. These datasets help test the model's capability to handle domain shifts, with the task being digit classification from 0 to 9.

The CelebA dataset [21] comprises 202,599 face images of celebrities, annotated with 40 distinct binary attributes. This extensive dataset is utilized to train on specific attributes such as hair color (black, blonde, brown), gender (male), and age (young). For preprocessing, the original images, sized 178 x 218, are cropped to 178 x 178 and resized to 128 x 128 to standardize the input size. To evaluate performance, 2,000 images are randomly selected as the test set, with the remainder serving as the training set. The CelebA dataset allows for the construction of seven distinct domains based on combinations of hair color, gender, and age.

In our experiments, all models are trained using the Adam optimizer [22] with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. To balance training dynamics, we update MASCN and MDAN once for every five updates of the discriminator, following the strategy recommended in [23]. For the CelebA dataset, we utilize a batch size of 16, while for digit datasets, the batch size is increased to 128. All models are initially trained with a learning rate of 0.0001 for the first 10 epochs, after which the rate is linearly decayed to zero over the subsequent 10 epochs to ensure smooth convergence. Hyperparameters are set with $\lambda_{cls} = 1$, $\lambda_{rec} = 10$, and $\lambda_{prg} = 1$.

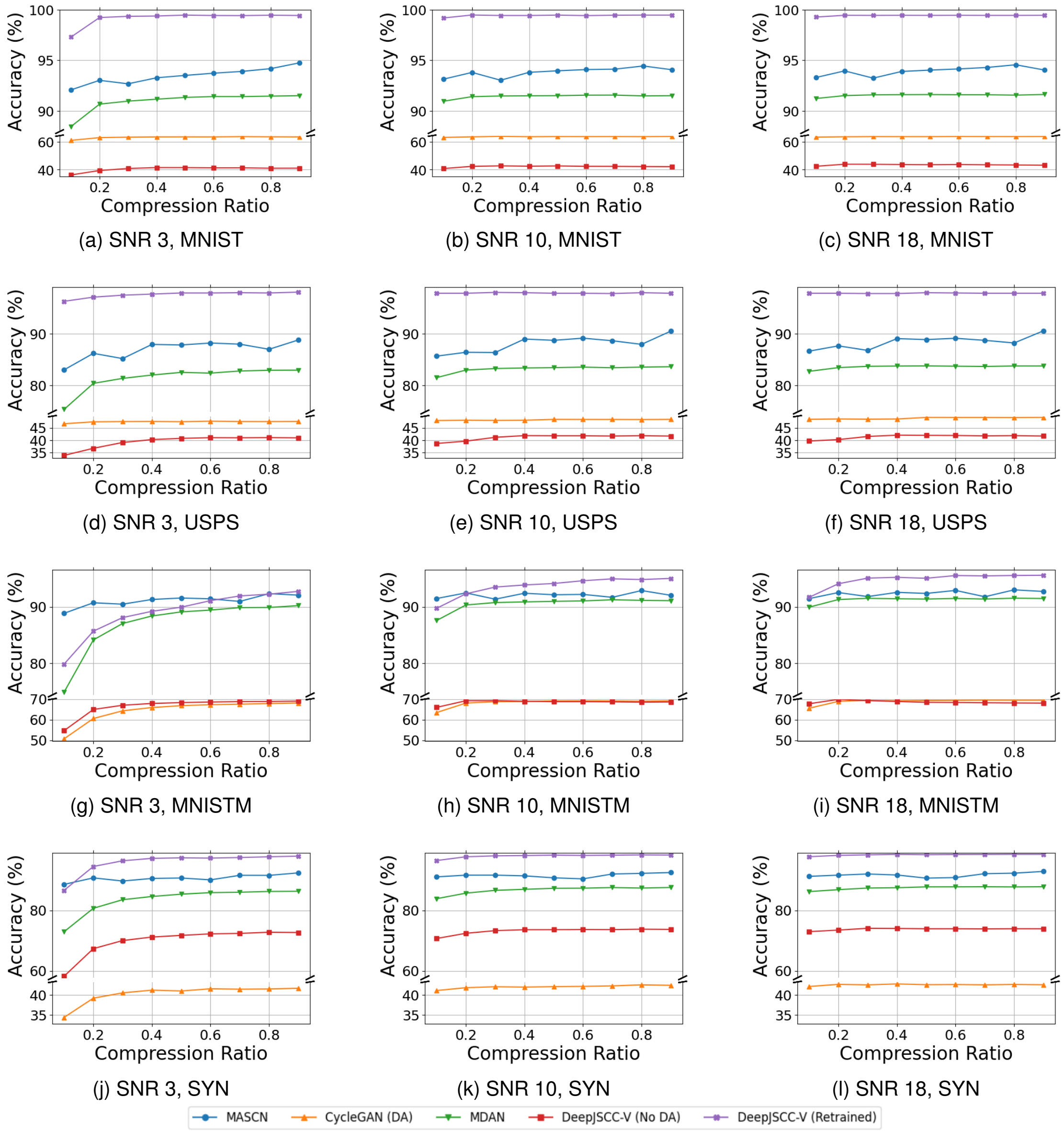
In evaluating our framework on the digit and CelebA datasets, we conducted both quantitative and qualitative analyses. For the quantitative analysis, we tested on different CRs ranging from 0.1 to 0.9 and SNRs at 3, 10, and 18. For the qualitative analysis, we examined CRs of 0.2 and 0.7 and an SNR of 3. We constructed a balanced random test dataset composed of equal sizes for each domain and assessed the classification accuracy following specific domain transformations.

C. Experiment Results on Digit Datasets

Fig. 4 (a) to (c) illustrates the MNIST experiment outcomes across different SNRs. Retrained DeepJSCC-V set a high benchmark, achieving nearly 99% accuracy across SNRs. MASCN and MDAN substantially outperformed CycleGAN-based DA, with accuracies improving by approximately 27% to 30% in all SNR settings. Specifically, MASCN reached accuracies up to 93.93%, while MDAN achieved up to 91.53%, significantly surpassing the CycleGAN-based DA and DeepJSSC-V without DA, which scored lowest at each SNR.

Fig. 4 (d) through (f) show the USPS dataset outcomes at different SNRs. MASCN and MDAN outshined the CycleGAN-based DA across all settings, with MASCN achieving up to 88.41% and MDAN 83.60%. Retrained DeepJSSC-V consistently led with nearly 98% accuracy, while DeepJSSC-V with no DA lagged behind, never surpassing 41.44%.

As shown in Fig. 4 (g) to (i) for MNISTM, MASCN demonstrated consistent improvement across SNRs, achieving accuracies of 91.10%, 92.09%, and 92.37%, respectively. MDAN also showed strong performance with accuracies reaching up to 91.29%. In contrast, CycleGAN-based DA and DeepJSSC-V with no DAlagged significantly, with the highest performances being 68.95% and 68.58%, respectively. No-



(a) SNR 3, MNIST

(b) SNR 10, MNIST

(c) SNR 18, MNIST

(d) SNR 3, USPS

(e) SNR 10, USPS

(f) SNR 18, USPS

(g) SNR 3, MNISTM

(h) SNR 10, MNISTM

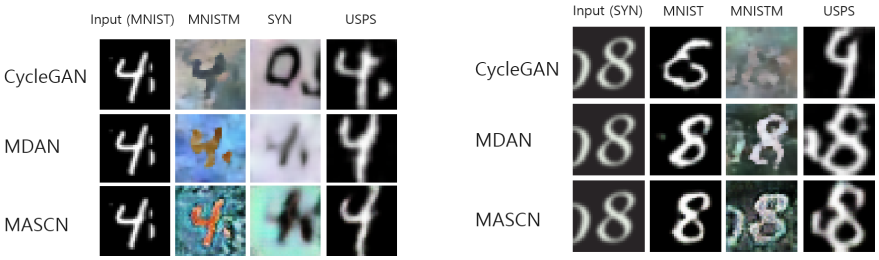
(i) SNR 18, MNISTM

(j) SNR 3, SYN

(k) SNR 10, SYN

(l) SNR 18, SYN

Fig. 4: Classification accuracy comparison of digit datasets for different CR and SNR.



(a) CR = 0.2, SNR = 3

(b) CR = 0.7, SNR = 3

Fig. 5: Qualitative Comparison of digit datasets for different CR and SNR.

tably, the retrained DeepJSSC-V exhibited robust results, closely matching MASCN with a peak accuracy of 94.84% at the highest SNR.

For the SYN dataset, evaluations across compression ratios from 0.1 to 0.9 and SNRs of 3, 10, and 18 demonstrated that MASCN surpassed CycleGAN-based DA by 50.41%, 49.52%, and 49.21%, respectively. Similarly, MDAN outperformed CycleGAN-based DA by 43.25%, 44.69%, and 44.92% at these SNRs. MASCN also consistently exceeded MDAN's performance by 7.16%, 4.83%, and 4.29%, highlighting its superior adaptability in diverse domain scenarios. These findings, detailed in Fig. 4 (j) to (l), affirm the robustness of our models under various operational conditions.

Fig. 5 presents the visual results for the CR of 0.2 and 0.7. In the scenarios where the transformation targeted SYN at CR=0.2 and MNIST, at CR=0.7, MNIST-M and USPS, CycleGAN-based DA doesn't perform accurate transformations, altering the semantic information. In contrast, MDAN and MASCN effectively preserved the original semantic content, ensuring the images were sufficiently restored for accurate classification.

D. Experiment Results on CelebA

Fig. 6 (a) to (c) highlights CelebA dataset transformations to blond hair across SNRs of 3, 10, and 18. MASCN significantly outperformed CycleGAN-based DA with improvements of 48.99%, 50.41%, and 26.45% respectively. MDAN also showed substantial gains, enhancing performance by 46.06%, 47.95%, and 24.98% across these SNRs. MASCN consistently held modest advantages over MDAN, solidifying its adaptability in domain transformations.

Transformations to black hair are depicted in Fig. 6 (d) to (f), where MASCN's improvements of 26.45%, 27.95%, and 29.21% at each SNR distinctly outpaced CycleGAN-based DA. MDAN's enhancements at 24.98%, 25.85%, and 26.21% also surpassed CycleGAN-based DA, with MASCN maintaining leads over MDAN, emphasizing its robust adaptation capabilities.

For gender transformations, shown in Fig. 6 (g) to (i), MASCN advanced with improvements of 44.03%, 42.87%, and 42.38%, significantly over CycleGAN-based DA. MDAN's performance improvements at 33.70%, 36.48%, and 36.74% also highlighted its effectiveness, with MASCN outperforming MDAN by notable margins.

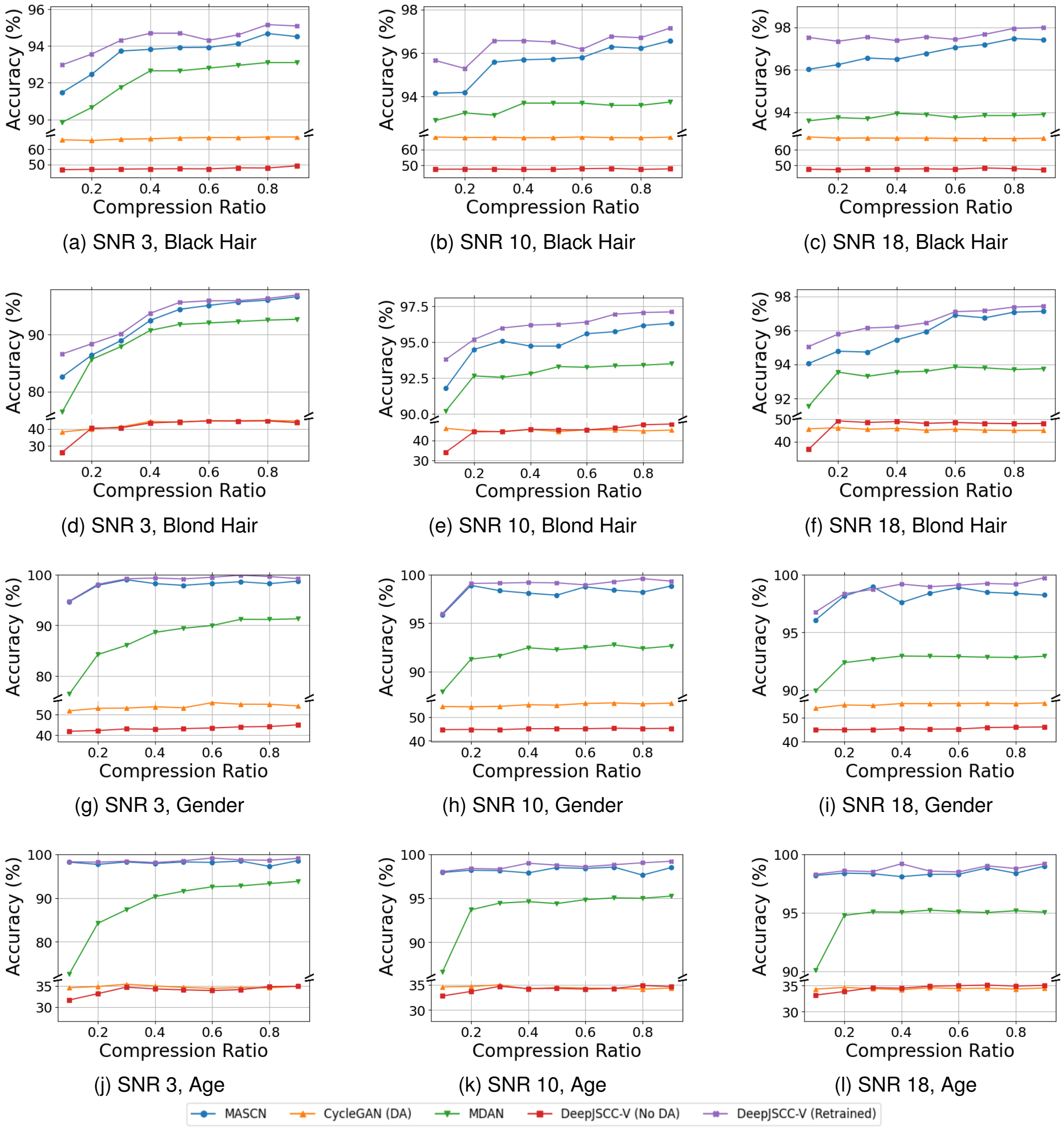
Lastly, age transformations in Fig. 6 (j) to (l) illustrate MASCN's dominance with gains of 63.25%, 63.72%, and 63.95%, outstripping CycleGAN-based DA. MDAN's performance also surged with increments of 53.91%, 59.31%, and 60.04%, confirming MASCN's superior capability in adapting to age attributes under various conditions.

Fig. 7 presents the visual outcomes at CR of 0.2 and 0.7. As CR decreases, the transformed images exhibit increased blurriness. Nevertheless, even at a lower CR of 0.2 and with reduced SNR, MDAN and MASCN managed to perform transformations that allowed for accurate classification. In contrast, CycleGAN based DA sometimes incorrectly translated gender attributes at CR=0.2, highlighting the robustness of MDAN and MASCN in maintaining semantic accuracy under stringent conditions.

VII. Discussion

In this section, we examine the strategic advantages of MDAN compared to MASCN. MDAN can be trained significantly faster than MASCN because they reuse existing semantic coding frameworks. This makes MDANs highly advantageous in environments characterized by frequent domain shifts, facilitating fast adaptation to new domains. While MDAN offer quick adaptability, this comes at the cost of increased computational overhead from additional data adaptation network. Conversely, MASCN is inherently designed to adapt across multiple domains efficiently. This capability is crucial, especially when the transmitter and receiver operate in distinctly different domains, or when there is a prior understanding of multidomain attributes in the source data. MASCN leverages this pre-awareness to ensure that the system is well-prepared to adapt for each domain.

Moreover, the integration of MDAN and MASCN offers a complementary strategy, for dynamic environments with frequent domain shifts, deploying a MDAN can provide an immediate, albeit temporary, solution while MASCN is being trained in the backend. Once MASCN training is complete, it can replace the MDAN, combining the benefits of rapid initial adaptability with robust, long-term stability. This dual approach ensures that the system benefits from the quick adaptability of MDAN initially, followed by the long-term stability of MASCN.



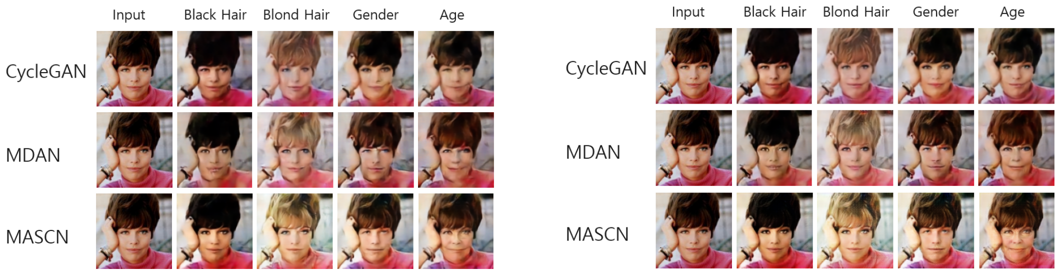
(a) SNR 3, Black Hair (b) SNR 10, Black Hair (c) SNR 18, Black Hair

(d) SNR 3, Blond Hair (e) SNR 10, Blond Hair (f) SNR 18, Blond Hair

(g) SNR 3, Gender (h) SNR 10, Gender (i) SNR 18, Gender

(j) SNR 3, Age (k) SNR 10, Age (l) SNR 18, Age

Fig. 6: Quantitative comparison of CelebA for different CR and SNR.



(a) CR = 0.2, SNR = 3 (b) CR = 0.7, SNR = 3

Fig. 7: Qualitative Comparison of digit datasets for different CR and SNR.

VIII. Conclusion

This paper introduced the MA-DeepSC framework, comprising MASCN and MDAN. This framework not only enhances scalability but also retains high levels of classification accuracy in various SNR compared to existing methods. Our experimental results demonstrate that MA-DeepSC achieves performance comparable to that of the retrained DeepJSCC-V and significantly outperforms the CycleGAN-based DA approach in multidomain scenarios. Unlike DeepJSCC-V, which requires a separate trained model for each domain, MA-DeepSC adapts to multiple domains with a single model, greatly simplifying deployment and reducing overhead. In conclusion, MA-DeepSC advances the field of semantic communications by addressing the critical need for systems that operate effectively across multidomain. We envision that our findings will contribute to the ongoing development of practical and efficient semantic communication systems, potentially influencing future standards and implementations in the industry.

References

[1] M. A. Amanullah, R. A. A. Habeeb, F. H. Nasaruddin, A. Gani, E. Ahmed, A. S. M. Nainar, N. M. Akim, and M. Imran, "Deep learning and big data technologies for iot security," *Computer Communications*, vol. 151, pp. 495–517, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0140366419315361

[2] H. Xie, Z. Qin, X. Tao, and K. B. Letaief, "Task-oriented multi-user semantic communications," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 9, pp. 2584–2597, 2022.

[3] Z. Qin, X. Tao, J. Lu, W. Tong, and G. Y. Li, "Semantic communications: Principles and challenges," *arXiv preprint arXiv:2201.01389*, 2021.

[4] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, "Deep learning enabled semantic communication systems," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2663–2675, 2021.

[5] Z. Weng, Z. Qin, X. Tao, C. Pan, G. Liu, and G. Y. Li, "Deep learning enabled semantic communications with speech recognition and synthesis," *IEEE Transactions on Wireless Communications*, 2023.

[6] D. Huang, F. Gao, X. Tao, Q. Du, and J. Lu, "Toward semantic communications: Deep learning-based image semantic coding," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 1, pp. 55–71, 2022.

[7] T.-Y. Tung and D. Gündüz, "Deepwive: Deep-learning-aided wireless video transmission," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 9, pp. 2570–2583, 2022.

[8] H. Zhang, S. Shao, M. Tao, X. Bi, and K. B. Letaief, "Deep learning-enabled semantic communication systems with task-unaware transmitter and dynamic data," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 1, pp. 170–185, 2023.

[9] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8789–8797.

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[11] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[12] 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*, 2017, pp. 2223–2232.

[13] Y. Taigman, A. Polyak, and L. Wolf, "Unsupervised cross-domain image generation," *arXiv preprint arXiv:1611.02200*, 2016.

[14] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, "Learning to discover cross-domain relations with generative adversarial networks," in *International conference on machine learning*. PMLR, 2017, pp. 1857–1865.

[15] D. B. Kurka and D. Gündüz, "Deepjscc-f: Deep joint source-channel coding of images with feedback," *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 178–193, 2020.

[16] J. Xu, B. Ai, W. Chen, A. Yang, P. Sun, and M. Rodrigues, "Wireless image transmission using deep source channel coding with attention modules," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 4, pp. 2315–2328, 2021.

[17] E. Bourtsoulatze, D. B. Kurka, and D. Gündüz, "Deep joint source-channel coding for wireless image transmission," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 3, pp. 567–579, 2019.

[18] W. Zhang, H. Zhang, H. Ma, H. Shao, N. Wang, and V. C. Leung, "Predictive and adaptive deep coding for wireless image transmission in semantic communication," *IEEE Transactions on Wireless Communications*, 2023.

[19] J.-Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman, "Toward multimodal image-to-image translation," *Advances in neural information processing systems*, vol. 30, 2017.

[20] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *International conference on machine learning*. PMLR, 2017, pp. 2642–2651.

[21] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3730–3738.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[23] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," *Advances in neural information processing systems*, vol. 30, 2017.