Data Adaptive Semantic Communication Systems for Intelligent Tasks and Image Transmission

Zhiqiang Feng, Donghong Cai(✉), Zhiquan Liu(✉), Jiahao Shan, and Wei Wang

College of Information Science and Technology, Jinan University, Guangzhou 510632, People's Republic of China
{63064fzq,shanjh,walle}@stu2022.jnu.edu.cn, {dhcai,zqliu}@jnu.edu.cn

Abstract. In this paper, we consider an end-to-end semantic communication system for compressed image wireless transmission. To economize the communication bandwidth, and enhance the communication reliability as well as the intelligence of the receiver, two networks are designed for image transmission and intelligent tasks. In particular, two simple linear layers are used to extract the feature of the image and recover the image, taking into account the sparsity of the image in a specific linear space. The trained encoder is deployed at the transmitter, while the trained decoder and the classifier are deployed at the receiver. To adapt to new communication data, the proposed networks are trained in a meta-learning framework. A few samples of new data are fed into the trained network to calculate new model parameters, which are fed back to the transmitter for updating the network of the encoder. Experimental results show that the proposed system has superior performance in terms of image compression transmission over fading channels compared with the existing semantic communication systems.

Keywords: Semantic communication · image transmission · Meta-learning

1 Introduction

With the development of communication techniques and the applications of Metaverse and virtual reality, communication systems are gradually shifting from traditional bit information transmission to systems with more powerful semantic understanding [1–3]. The semantic communication systems are not only able to convey information to the recipient but also to make deep understanding of the information, enabling a wider range of applications [4], such as autonomous vehicles. By using the deep learning techniques to design the transmitter and the receiver, the semantic communication system can compress the source information for effective transmission. Meanwhile, the receiver design based on deep learning can improve the intelligence of communication.

F. Zhao and D. Miao (Eds.): AIGC 2023, CCIS 1946, pp. 105–117, 2024.
https://doi.org/10.1007/978-981-99-7587-7_9

Traditional compression methods, such as JPEG2000 [5] and H.264 [6], are commonly used for image and video compression. Especially, JPEG2000 uses Discrete Cosine Transform (DCT) to transform the image in the frequency domain, and then uses quantization and entropy coding to achieve data compression. H.264 is a widely used standard for video compression, which uses motion estimation, transform coding and entropy coding to improve the efficiency of video compression. These traditional compression methods can reduce the size of data effectively, but they may have some limitations for wireless communication. For example, compressed information is difficult to decode correctly due to channel fading and noise. Channel coding should be considered to protect the compressed information. Compared with traditional compression schemes, semantic communication encodes information in a structured, simplified and flexible way, and provides a new perspective for joint source and channel coding [7]. However, the accuracy and robustness of semantic understanding remains a critical issue. Since the communication data includes structured data and unstructured data, such as text, picture, and video, it is difficult to use a scope-limited knowledge base. In addition, data scarcity is a problem that cannot be ignored in current semantic systems. Semantic understanding of specific tasks usually requires large amounts of annotated data to train models. However in real scenarios, it is very difficult and expensive to obtain large-scale annotated data [8–10]. Existing semantic communication methods face the challenge of efficiency and scalability when dealing with large and diverse data. Besides, the effective knowledge transfer and representation is also an urgent problem.

Recently, Huiqiang Xie et al. [11] proposed that a deep neural network enabled semantic communication system, named MU-DeepSC, to execute the visual question answering (VQA) task. Through joint design and optimization of transceivers, the most relevant data features are extracted to achieve task-oriented transmission. However, deep learning-based semantic communication systems may suffer from problems such as data scarcity and labeling difficulties during the learning process. In order to solve the problem that the actual observation data at the transmitter may have inconsistent distribution with the empirical data in the shared background knowledge base, Hongwei Zhang et al. [12] proposed a new semantic communication system for image transmission based on neural network. By using the domain adaptation technique of transfer learning, the data adaptation network is designed to learn how to transform observed data into similar forms of empirical data that semantic coding networks can process without retraining. However, under the influence of low compression rate and fading channel, this training method based on transfer learning is not ideal. In addition, Chanhong Liu et al. [4] proposed a compression ratio and resource allocation (CRRA) algorithm to support multi-users to perform tasks at low compression rate and occupy fewer resources. However, CRRA is difficult to apply because of its high complexity.

To overcome the limitations of current research, we propose an end-to-end semantic communication system based on meta-learning. In particular, the encoder and the decoder are designed based on linear layers, because the image information can be sparse by linear transformation and be compressed by a sens-

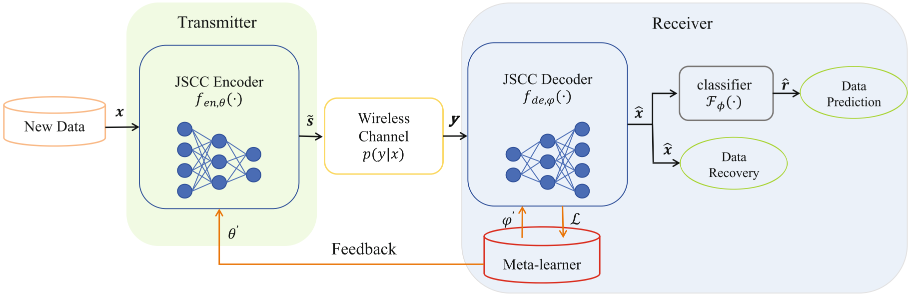

Fig. 1. Illustration of end-to-end semantic communication system.

ing matrix. A classifier is also designed to perform specific tasks. The proposed network is training based on the Model-agnostic meta-learning (MAML) algorithm [9]. Then the trained encoder is deployed at the sending end. The trained decoder and classifier are deployed at the receiving end. To accommodate the new data, a few samples of new data with labels are calculated with the trained model, and some parameters are fed back from the receiver to the transmitter for model update. The experimental results indicate that under low compression ratios (CR) and in the presence of fading channels, the reconstructed images transmitted through semantic encoding can be visually discernible to the human eye. Furthermore, the reconstructed images can still perform specific classification tasks.

2 System Model

Consider an end-to-end semantic communication system, where both of transmitter and receiver have single-antenna. As shown in Fig. 1, the input image signal is first encoded by a joint source and channel coder (JSCC) at the transmitter for compression wireless transmission. Meanwhile, a receiver designed by a deep neural network is used to reconstruct the image data as well as the intelligent recognition task. Especially, the input image data $\mathbf{x} \in \mathbb{R}^{N\times N\times L}$ is encoded by a feature vector $\tilde{\mathbf{s}} \in \mathbb{R}^{K}(K \ll N^2)$ containing the semantic information of the input image, i.e.,

$$\tilde{\mathbf{s}} = f_{en,\theta}(\mathbf{x}), \tag{1}$$

where $f_{en,\theta}(\cdot)$ represents the encoder with parameter θ, N is the size of the image and L is the numbers of channels of the image. Further, the vector $\mathbf{s}$ is normalized as $\mathbf{s} = \eta\tilde{\mathbf{s}}$. where $\eta = \frac{1}{||\tilde{\mathbf{s}}||_2^2}$ is the normalized coefficient.

At the receiver, the received signal vector $\mathbf{y} \in \mathbb{C}^K$ is formulated as

$$\mathbf{y} = h\tilde{\mathbf{s}} + \mathbf{z}, \tag{2}$$

where $h \in \mathbb{C}$ is a block fading channel coefficient, and $\mathbf{z} \sim \mathcal{CN}(0, \sigma^2\mathbf{I}_K)$ represents the symmetric complex additive Gaussian noise with mean zero and variance σ^2. The semantic decoder deployed at the receiving end will reconstruct the original image from $\mathbf{y}$, which is given by

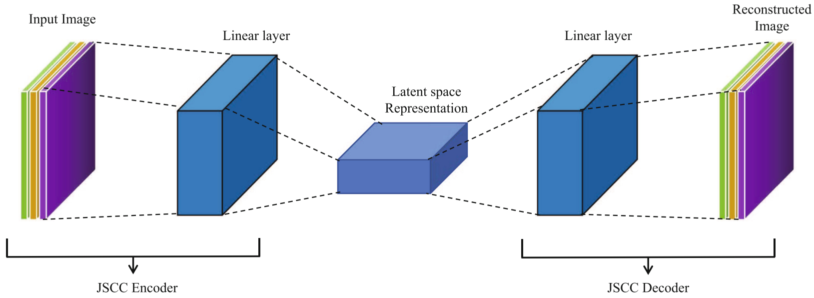

Fig. 2. The proposed JSCC encoder and decoder networks.

$$\hat{\mathbf{x}} = f_{de,\varphi}(\mathbf{y}), \tag{3}$$

where $f_{de,\varphi}(\cdot)$ represents the decoder with parameter φ. The recovery image $\hat{\mathbf{x}}$ is further used to perform the recognition task, i.e.,

$$\hat{r} = \mathcal{F}_{\phi}\left(\hat{\mathbf{x}}\right), \tag{4}$$

where $\mathcal{F}_{\phi}(\cdot)$ is the classifier with parameter ϕ and $\hat{r}$ is the prediction result.

Note that $f_{en,\theta}$, $f_{de,\theta}$ and $\mathcal{F}_{\phi}$ of semantic communication are designed by the deep neural networks and trained with the empirical data. The trained $f_{en,\theta}$ is deployed at the transmitter for compressed code transmission. The trained $f_{de,\theta}$ and $\mathcal{F}_{\phi}$ are deployed at the receiver side for image recovery and the classification task. Thus the semantic communication can reduce the bandwidth requirement while ensuring the communication quality, and improve the intelligent commitment of the receiver. However, the performance of semantic communication degrades seriously when the communication data is dynamically transformed or a new communication data appears. To this end, a semantic communication network is designed to adapt to the new communication data quickly. Different from the data adaptation network in [12], where a GAN is used to generate the target data, the networks of semantic communication are trained with Meta-learning framework in this paper.

3 Proposed Semantic Communication Networks

In this section, we first introduce deep neural network architectures that satisfy requirements of coder and decoder for semantic communication, including the joint source and channel encoder, decoder for image recovery, and classifier for performing intelligent tasks. Then, the data adaptive semantic communication network is obtained by introducing the MAML algorithm. When new communication data needs to be transmitted, the semantic communication model is updated through a few of feedbacks and adapted to the reliable transmission of new data.

3.1 Network Structures of Encoder and Decoder

On the one hand, the encoder of semantic communication needs to extract the features of image data well, reduce the amount of transmitted data as much as possible and reduce the communication bandwidth overhead while ensuring the transmission quality. On the other hand, the decoder needs to guarantee the quality of recovered image. It is important to point out that the proposed semantic communication system requires less cycles of model feedback updates. In addition, lightweight networks of JSCC coder and encoder should be designed. To this end, we propose two simple networks for the encoder and the decoder of semantic communication, which mainly consists of two linear layers, as shown in Fig. 2.

In the end-to-end semantic communication system, the encoder and decoder are deployed at the transmitter and receiver, respectively. The output of encoder is transmitted through the wireless channel. Therefore, the encoder need to compress source information and resist wireless channel fading. Meanwhile, the decoder has to generate the image based on the compressed feature of original image with noise. Note that the image can be sparsely represented as

$$\mathbf{D} = \boldsymbol{\Psi}\mathbf{X}\boldsymbol{\Psi}^T, \tag{5}$$

where $\boldsymbol{\Psi} \in \mathbb{R}^{N\times N}$ is a wavelet basis matrix. The wavelet coefficient matrix $\mathbf{D}$ is parse in (5), which can be further expressed as

$$\mathrm{vec}(\mathbf{D}) = (\boldsymbol{\Psi} \otimes \boldsymbol{\Psi})\mathrm{vec}(\mathbf{X}), \tag{6}$$

where $\mathrm{vec}(\cdot)$ denotes the vectorization of a matrix, and $\otimes$ denotes the Kronecker product. Then the sparse vector $\mathrm{vec}(\mathbf{D}) \in \mathbb{R}^{NN\times 1}$ can be compressed by a dictionary matrix of compressed sensing, results in

$$\bar{\mathbf{s}} = \mathbf{A}\mathrm{vec}(\mathbf{D}) = \mathbf{A}(\boldsymbol{\Psi} \otimes \boldsymbol{\Psi})\mathrm{vec}(\mathbf{X}), \tag{7}$$

where $\mathbf{A} \in \mathbb{R}^{M\times N}$ is a dictionary matrix of compressed sensing, and $\bar{\mathbf{s}} \in \mathbb{R}^{M\times 1}$ is the obtained compressed vector. In fact, the compressed vector $\bar{\mathbf{s}}$ is equivalent to the feature vector $\tilde{\mathbf{s}}$ in(1) if the number of rows of dictionary matrix is designed as K, i.e., $K = M$. However, the design of dictionary matrix $\mathbf{A}$ is based on the sparsity of $\mathrm{vec}(\mathbf{D})$, and the quality of image recovery decreases without Restricted Isometry Property (RIP)-like conditions. To this end, a network of encoder with one linear layer can be used to implement the compression of image in (7). The encoder is

$$f_{en,\theta} \triangleq \mathbf{A}(\boldsymbol{\Psi} \otimes \boldsymbol{\Psi}) \triangleq \boldsymbol{\Lambda} \in \mathbb{R}^{K\times N}. \tag{8}$$

Note that the input of decoder based on (2) can be expressed as

$$\tilde{\mathbf{y}} = \mathbf{s} + \tilde{\mathbf{z}}, \tag{9}$$

where $\tilde{\mathbf{y}} = \mathbf{y}/h$ and $\tilde{\mathbf{z}} = \mathbf{z}/h$. Then the recovery image is given by

$$\hat{\mathbf{x}} = f_{de,\varphi}(\mathbf{s} + \tilde{\mathbf{z}}). \tag{10}$$

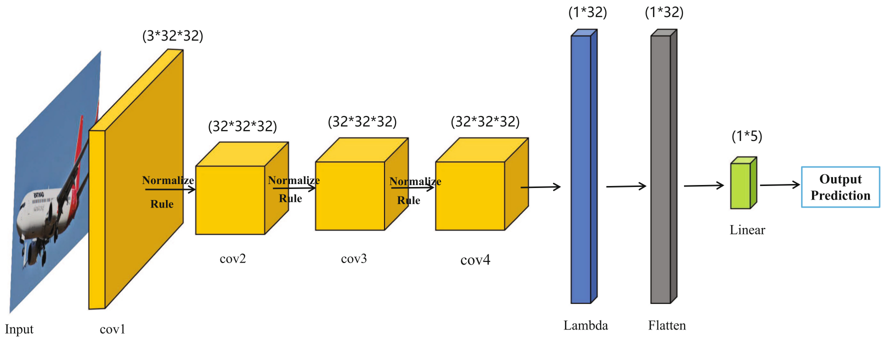

Fig. 3. The proposed network of classifier for semantic communication.

Based on the encoder presented in (8), the input of decoder can be re-written as

$$\tilde{\mathbf{y}} = \boldsymbol{\Lambda}\text{vec}(\mathbf{X}) + \tilde{\mathbf{z}}, \tag{11}$$

which is a linear expression of original image. Applying the linear MMSE, we have

$$\text{vec}(\hat{\mathbf{X}}) = \left(\boldsymbol{\Lambda}^T\boldsymbol{\Lambda} + \frac{\sigma^2}{|h|^2}\mathbf{I}_K\right)^{-1}\boldsymbol{\Lambda}^T\tilde{\mathbf{y}} \triangleq \boldsymbol{\Omega}\tilde{\mathbf{y}}. \tag{12}$$

It is important to point out that the linear transformation matrixes $\boldsymbol{\Lambda}$ and $\boldsymbol{\Omega}$ are unknown and difficult to design. Thus, we use the linear layer to achieve image recovery in our proposed networks.

3.2 Network Structures of Classifier

The purpose of deploying classifier at the receiving end of semantic communication is to quickly and intelligently recognize the meaning of the transmitted information. In fact, it requires the receiver equipped with a classifier to classify the received information. To this end, we proposed a classifier network, which consists of 4 convolution blocks, Lambda layer, Flatten layer and fully connected layer, as shown in Fig. 3. Each convolution block consists of several key components, including a 3×3 convolution with stride 1 and padding 1, a regularization layer, a Relu activation function, and a maximum pooling layer. The function of these convolutional blocks is to better extract the features of the input image. Through the extraction layer by layer, different information in the image can be captured. After four convolution blocks are processed, the information enters the Lambda layer. The Lambda layer is used for mean pooling of features to calculate the mean value of each channel. This action helps to further reduce the dimension of the feature while effectively preserving important information. After processing by the Flatten layer, the features are flattened into one-dimensional vectors for easy feeding into the fully connected layer. The function of the Flatten layer is to convert the features extracted from the convolution layer into a suitable form for processing by the fully connected layer such that the classification task can be better performed. Finally, the feature is passed to the fully connected layer, which classifies and maps the input features to each class's probability distribution. By training the fully connected layer's parameters, the classifier accurately determines the image's category based on the input features.

3.3 Model Training

For the model training, we use meta-learning framework to train the transmission network and the classifier network. In particular, the meta-learning of semantic communication system is divided into the meta-training stage and the meta-adaptation stage.

1) Meta-training stage: The training can be divided into inner-loop stage and outer-loop stage. The inner-loop stage is the semantic codec performing gradient descent for the loss of a specific task. And the outer-loop is updating the randomly initialized model parameters by calculating the gradient relative to the optimal parameters in each new task. It is assumed that there are I tasks. The task $T_i, i = 1, 2, \cdots, I$, has a training set D_i^{tr} and a validation set D_i^{val}. For the inner-loop phase, task T_i updates its model parameters by randomly sampling $\mathcal{K}$ samples ($\mathcal{K}$ is a small integer) according to its own specific task. Specifically, the encoder generates few low-dimensional feature vectors $\tilde{\mathbf{s}}_{i,j}, j = 1, 2, \cdots, \mathcal{K}$, from the input of training set D_i^{tr} in task T_i, and then gets $\mathbf{y}_{i,j}$ after normalization through fading channels. The decoder reconstructs an image similar to the input image through the decoder network. Then the loss of the original image and the generated image is

$$\mathcal{L}_i^{tr} = \mathcal{D}_{MSE}(X_i^{tr}, \hat{X}_i^{tr}), \tag{13}$$

where X_i^{tr} and $\hat{X}_i^{tr}$ are the image of the i-th task and the reconstructed image, respectively. In addition, $\mathcal{D}$ denotes the mean square error (MSE) of X_i^{tr} and $\hat{X}_r^{tr}$. Then, task T_i uses $\mathcal{L}_i^{tr}$ to perform gradient descent. The parameter update of the i-th task can be expressed as

$$\psi'(\theta', \varphi') \leftarrow \begin{cases} \theta_i' \leftarrow \theta_i - \alpha \arg\min \nabla_\theta \mathcal{L}_i^{tr}, \\ \varphi_i' \leftarrow \varphi_i - \alpha \arg\min \nabla_\varphi \mathcal{L}_i^{tr}, \end{cases} \tag{14}$$

where α is the learning rate of the inner-loop. Furthermore, task T_i uses the updated model parameters $\psi'(\theta', \varphi')$ to find the loss between the reconstructed image and the original image on validation set D_i^{val}, which is given by

$$\mathcal{L}_i^{val} = \mathcal{D}_{MSE}(X_i^{val}, \hat{X}_i^{val}), \tag{15}$$

where X_i^{val} and $\hat{X}_i^{val}$ are the image of the i-th task and the reconstructed image, respectively. After completing the above inner-loop, we get the loss $\mathcal{L}_i^{val}$ for $i = 1, 2, \cdots I$. For the outer-loop, the sum losses is

Algorithm 1. Model training and data adaptation of semantic communication

Meta-training Stage: Train semantic encoders and decoders

Input: Training $T_i, i = 1, 2...I$, tasks, where the batch size is $\mathcal{K}$

1: **Initialize:**randomly initialize $\psi(\theta, \varphi)$
2: **while** not done **do**
3: **for all** T_i **do**
4: Sample $\mathcal{K}$ data $\mathbf{x}_{i,j} \in D_i^{tr} (j = 1, 2, \cdots, \mathcal{K})$ for train task T_i
5: Encoding: $\tilde{\mathbf{s}}_{i,j} = f_{en,\theta}(\mathbf{x}_{i,j})$
6: Decoding: $\hat{\mathbf{x}}_{i,j} = f_{de,\varphi}(\mathbf{y}_{i,j})$
7: Evaluate $\arg\min \nabla_\theta \mathcal{L}_i^{tr}$ using D_i^{tr}
8: Compute train parameters $\psi'(\theta', \varphi')$ with gradient decent
9: Feed $\psi'(\theta', \varphi')$ back to the network for encoder and decoder
10: Get the loss $\mathcal{L}_i^{val}$ for task T_i with data D_i^{val}
11: **end for**
12: Update $\psi(\theta, \varphi)$ by equation (16) and feed it back to the encoder and decoder
13: **end while**

Meta-adaptation Stage: The trained encoder and decoder are deployed at the transmitter and receiver, respectively

Input: a new task B, the adaptive number of iterations is U, few samples for update the transmission model, the trained $\psi(\theta, \varphi)$ in Meta training stage

14: **for** $u = 1, 2, \cdots, U$ **do**
15: The transmitter encodes the new sample similar to line 5
16: The receiver decodes the information similar to line 6
17: The receiver evaluates $\arg\min \nabla_\theta \mathcal{L}^{tr}$ similar to line 7 and computes $\psi'(\theta', \varphi')$ similar to line 8
18: Parameter $\psi'(\theta', \varphi')$ is feed back to the transmitter from the receiver
19: Get the loss $\mathcal{L}^{val}$ similar to line 10
20: **end for**
21: Update $\hat{\psi}(\hat{\theta}, \hat{\varphi})$ and feed it back to the transmitter

$$\psi(\theta, \varphi) \leftarrow \begin{cases} \theta \leftarrow \theta - \beta \arg\min \nabla_\theta \sum_{i=1}^{I} \mathcal{L}_i^{val}, \\ \varphi \leftarrow \varphi - \beta \arg\min \nabla_\varphi \sum_{i=1}^{I} \mathcal{L}_i^{val}, \end{cases} \tag{16}$$

where β is the learning rate of the outer-loop.

2) Meta-adaptation stage: The meta-learning training method based on parameter optimization aims to make the model have the ability to quickly adapt to the new task B. When the meta training is over, we deploy the encoder and decoder at the transmitter and receiver, respectively. Specifically, the data of new task is also divided into training sets D^{tr} and validation sets D^{val}, which enter the encoder for feature extraction through (1). The transmitter transmits the obtained signal (2) through the fading channel to the receiver. Furtherly, the loss (13) is obtained between the data reconstructed by decoding the decoder and the original data in D^{tr}. The meta-learner which in receiver updates the parameters of the encoder and decoder according to the loss as shown in (14) and sends them back to the encoder and decoder. The transmitter and the receiver use the

data from the validation set D^{val} to obtain the loss shown in (15). The parameters are then updated and transmitted back to the transmitter. The parameters of the new task are updated by

$$\hat{\psi}(\hat{\theta}, \hat{\varphi}) \leftarrow \begin{cases} \hat{\theta} \leftarrow \hat{\theta} - \beta \arg\min \nabla_{\hat{\theta}} \mathcal{L}^{val}, \\ \hat{\varphi} \leftarrow \hat{\varphi} - \beta \arg\min \nabla_{\hat{\varphi}} \mathcal{L}^{val}. \end{cases} \tag{17}$$

When the training converges, the encoder and decoder are able to compress and recover the new data. An update that usually takes only a few times. Model training and data adaptation of proposed semantic communication network is shown in **Algorithm** 1. Since the training of the classifier model also uses MAML algorithm, the training process of the classifier with cross entropy loss function is not introduced here.

4 Experimental Results

In this section, we verify the reconstruction performance and classification accuracy of semantic communication systems under various data sets, including MNIST, KMNIST, FasionMNIST, Omniglot, CIFAR-10 and STL-10. In addition, we evaluate the adaptive ability of the proposed networks, i.e., the model reconstructs data that was not used during training.

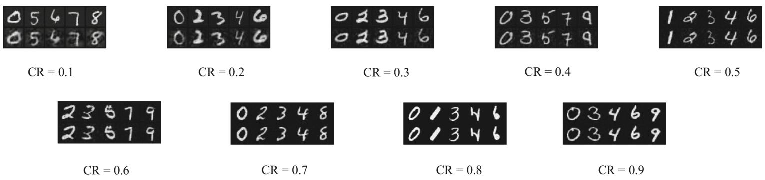

Fig. 4. Reconstruction results of MNIST over fading channel, where SNR = 10 dB.

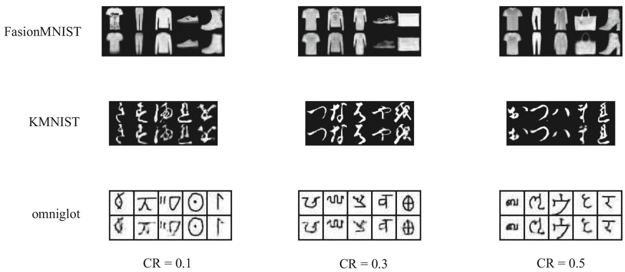

Fig. 5. Reconstruction performance of different single-channel data sets over fading channels, where SNR = 10 dB.

4.1 Image Recovery

To validate the experimental performance of the proposed network for image restoration, we conducted tests using the MNIST dataset first, which comprises a training set of 60000 handwritten numerical images and a test set of 10000 samples. Each sample in grayscale format is a 28×28 pixels image associated with a label representing the correct identification of the handwritten number shown. The MNIST data set has been pre-processed and normalized to ensure that all images have a consistent size and orientation. The CR is defined as the ratio of the length of the feature vector to the total number of pixels in the original image. In semantic communication system, the feature vector passes through the Rayleigh fading channel after compression, and finally, the signal is transmitted to the decoder for reconstruction.

As shown in Fig. 4, the reconstruction results are presented for MNIST data set with CR from 0.1 to 0.9, where SNR = 10 dB. We compare the image recovery performance of different data sets in Fig. 5, including KMNIST, FasionMNIST and Omniglot data set. The variation trend of PSNR with the compression ratio is shown in Fig. 6. It can be observed that the PSNR increases with the increase of CR. Because the pixels of the original image are different, the PSNRs of image recovery are different at the same CR.

In Fig. 7, PSNR comparison of different methods are presented for MNIST data set, including Autoencoder (AE) with only linear layers, AE with convolutional layers, Variational Autoencoder (VAE), and GAN-Data Adaptation Networks [12]. It can be seen that the proposed network has better reconstruction result than VAE and data transfer learning. Meanwhile, we can obtain similar PSNRs between AE based on convolutional neural network and our network. But the simpler network structure of our method results in lower network overhead for model update feedback.

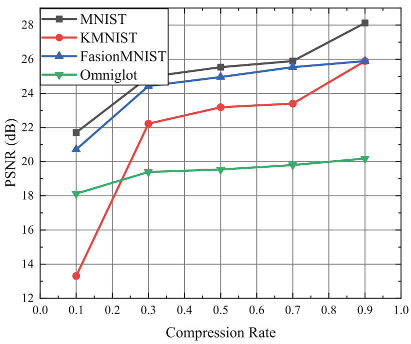

Fig. 6. PSNRs of different data sets over fading channel, where SNR = 10 dB.

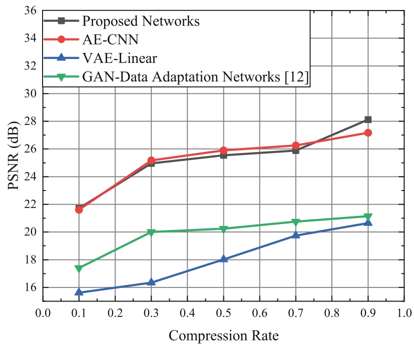

Fig. 7. Comparison of different methods over fading channel, where SNR = 10 dB.

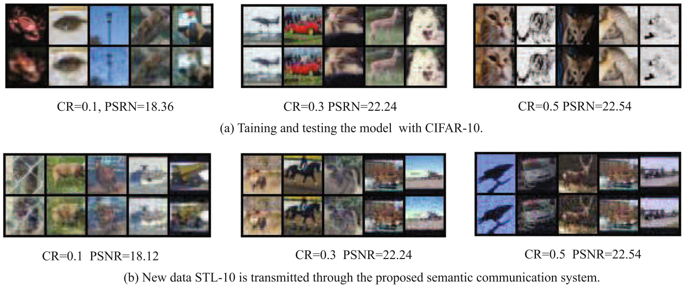

Fig. 8. Reconstructions of CIFAR-10 and STL-10 over fading channels, where SNR = 10 dB.

4.2 Data Adaptation

In this subsection, the main focus is to validate the adaptive capability of the semantic communication model. Specifically, we first train the model using the CIFAR-10 dataset, which consists of 60000 color images in $3 \times 32 \times 32$ format. These images are divided into 10 different categories, including cats, dogs, airplanes, trucks, and more, with each category containing 6000 images. Then, we test the model using the STL-10 dataset, which has different distribution compared to CIFAR-10. The images in these datasets are source from real-world photos and aim to reflect the diversity and complexity of real-life scenes.

We use a 5-ways, 1-shot training setup, i.e., there are 5 categories in the training set, and each category has only one sample. This means that the model needs to learn from just one sample in each category and be able to correctly reconstruct other untrained samples during the testing phase. In the training

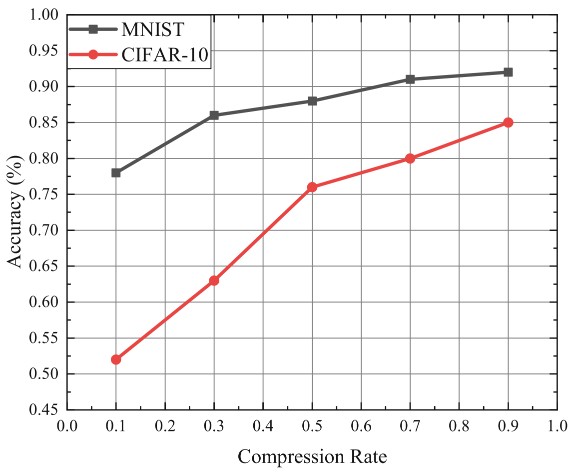

Fig. 9. Classification accuracy.

stage, we randomly divided CIFAR-10 into 32 subtasks, each of which contains 5 categories and only one sample for each category. The gradient update of the model is carried out by MAML algorithm. During the test phase, we use the model trained by CIFAR-10 to reconstruct STL-10 data. Figure 8 (a) shows the reconstruction effect of the model on CIFAR-10. Figure 8 (b) shows the adaptive effect of the model on STL-10 as new data. It can be seen that the model can reconstruct relatively clear pictures for both CIFAT-10 and STL-10. In particular, STL-10 as new data without training can be reconstructed by the proposed networks.

4.3 Classification Task

It should be noted that the classifier used is a pre-trained model. In order to make the classifier have strong generalization ability, the classifier training is also trained by MAML algorithm. This allows the trained classifier to directly classify images reconstructed by the transmit network. It verifies that the image reconstructed can be used to perform a specific tasks (such as classification tasks) in the semantic communication system. Figure 9 shows the accuracy of classification task over MNIST and CIFAR-10 data sets reconstructed by transmit network. It can be seen that the classification accuracy of MNIST can reach 0.78, and the classification accuracy of CIFAR-10 can reach 0.51 for CR = 0.1 and SNR = 10 dB. When the CR is 0.5, the reconstructed image can achieve a better classification accuracy, which indicates that the semantic communication system can capture the semantic content of the image rather than completely retain every detail of the original image.

5 Conclusion

We have considered an end-to-end semantic communication system in this paper. A simple linear network was designed for encoder and decoder based on the special sparse structure of image in a linear space. The proposed network extracted the feature of image for wireless transmission, and the decoder recovered the image based on the noise version of feature. In addition, a classifier was designed for performing special intelligent tasks with the recovery image. To further adapt new communication data, the proposed network was trained in Meta learning framework with few feedback from receiver to transmitter. The obtained results show that the proposed networks of semantic communication have a superior performance of image compression transmission over fading channels compared with the existing semantic communication systems.

Acknowledgment. This work was supported by the Science and Technology Major Project of Tibetan AutonomousvRegion of China under Grant No. XZ202201ZD0006G02, and the Science and Technology Project of Guangzhou under Grant No. 202201010200.

References

1. Xie, H., Qin, Z., Tao, X., Letaief, K.B.: Task-oriented multi-user semantic communications. IEEE J. Sel. Areas Commun. **40**(9), 2584–2597 (2022)
2. Weng, Z., Qin, Z.: Semantic communication systems for speech transmission. IEEE J. Sel. Areas Commun. **39**(8), 2434–2444 (2021)
3. Xie, H., Qin, Z., Li, G.Y., Juang, B.-H.: Deep learning enabled semantic communication systems. IEEE Trans. Signal Process. **69**, 2663–2675 (2021)
4. Liu, C., Guo, C., Yang, Y., Jiang, N.: Adaptable semantic compression and resource allocation for task-oriented communications. arXiv preprint arXiv:2204.08910 (2022)
5. Taubman, D.S., Marcellin, M.W., Rabbani, M.: JPEG2000: image compression fundamentals, standards and practice. J. Electron. Imaging **11**(2), 286–287 (2002)
6. Richardson, I.E.: H. 264 and MPEG-4 Video Compression: Video Coding for Next-Generation Multimedia. Wiley, Hoboken (2004)
7. Zhang, P., et al.: Toward wisdom-evolutionary and primitive-concise 6G: a new paradigm of semantic communication networks. Engineering **8**, 60–73 (2022)
8. Wang, Y., Yao, Q., Kwok, J.T., Ni, L.M.: Generalizing from a few examples: a survey on few-shot learning. ACM Comput. Surv. (CSUR) **53**(3), 1–34 (2020)
9. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In International Conference on Machine Learning, pp. 1126–1135. PMLR (2017)
10. Li, Z., Zhou, F., Chen, F., Li, H.: Meta-SGD: learning to learn quickly for few-shot learning. arXiv preprint arXiv:1707.09835 (2017)
11. Xie, H., Qin, Z., Li, G.Y.: Task-oriented multi-user semantic communications for VQA. IEEE Wirel. Commun. Lett. **11**(3), 553–557 (2021)
12. Zhang, H., Shao, S., Tao, M., Bi, X., Letaief, K.B.: Deep learning-enabled semantic communication systems with task-unaware transmitter and dynamic data. IEEE J. Sel. Areas Commun. **41**(1), 170–185 (2022)