

Research Article

Spatial Transformer Network-Based Automatic Modulation Recognition of Blind Signals

Yuxin Huang 

School of Electronic and Electrical Engineering, Minnan University of Science and Technology, Quanzhou 362700, Fujian, China

Correspondence should be addressed to Yuxin Huang; yiasom@163.com

Received 3 November 2021; Revised 29 November 2021; Accepted 2 December 2021; Published 16 December 2021

Academic Editor: Bai Yuan Ding

Copyright © 2021 Yuxin Huang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Modulation recognition of communication signals plays an important role in both civil and military uses. Neural network-based modulation recognition methods can extract high-level abstract features which can be adopted for classification of modulation types. Compared with traditional recognition methods based on manually defined features, they have the advantage of higher recognition rate. However, in actual modulation recognition scenarios, due to inaccurate estimation of receiving parameters and other reasons, the input signal samples for modulation recognition may have large phase, frequency offsets, and time scale changes. Existing deep learning-based modulation recognition methods have not considered the influences brought by the above issues, thus resulting in a decreased recognition rate. A modulation recognition method based on the spatial transformation network is proposed in this paper. In the proposed network, some prior models for synchronization in communication are introduced, and the priori models are realized through the spatial transformation subnetwork, so as to reduce the influence of phase, frequency offsets, and time scale differences. Experiments on simulated datasets prove that compared with the traditional CNN, ResNet, and the CLDNN, the recognition rate of the proposed method has increased by 8.0%, 5.8%, and 4.6%, respectively, when the signal-to-noise ratio is greater than 0. Moreover, the proposed network is also easier to train. The training time required for convergence has reduced by 4.5% and 80.7% compared to the ResNet and CLDNN, respectively.

1. Introduction

Modulation recognition of communication signals plays an important role in both civil and military applications. In civil use, modulation recognition technology is the basis for both the communication parties to automatically adjust the modulation type according to the current channel conditions or transmission quality, i.e., adaptive communication. In military applications, especially when receiving signals blindly, it is often impossible to know the relevant information of the received signal in advance, especially the information of modulation type. However, for further processing, the modulation type should be known in advance in many military applications.

For current modulation recognition technologies, there are mainly two types of methods: traditional methods and deep learning-based methods. The two methods are introduced as follows. In traditional methods, the features of the

signal are manually defined, such as spectral characteristics, instantaneous feature statistics, high-order moments, high-order cumulants, and so on [1–3]. Then, classification models can be established according to classic classifiers, such as decision tree (DT), support vector machine (SVM), and so on. The advantage for the type of methods is that the manually defined features can have better theoretical support. Because they have clear physical meanings, the synchronization parameters of the signal can be analyzed and extracted in the process of modulation recognition. The shortcomings for this type of methods are mainly as follows: (1) there is a lack of generalization ability, which affects the extraction of features under different channel conditions, resulting in a decrease in accuracy; (2) when there are many modulation types for recognition, the methods will also lead to a decline in the recognition rate due to limited number of manual features. For methods based on deep learning, the features for modulation recognition are automatically

extracted through training sample adopting deep neural networks, which can effectively avoid the shortcomings of traditional methods and achieve a higher recognition rate. However, deep learning-based methods also have shortcomings, such as the lack of interpretability of features and the inability to estimate signal parameters, such as symbol rate, in the process of modulation recognition.

Artificial intelligence has been successfully applied in the field of image and natural language processing (NLP). As the modulation recognition problem can be transferred to an image recognition problem, deep learning-based modulation recognition has also become a research hot spot. The following publications have summarized the application of deep learning in modulation recognition. References [4–6] directly adopt baseband samples for modulation recognition, which assumes that the input contains the same number of symbols. The effects of different neural network structures on modulation recognition rates are compared. The authors in [7, 8] adopt the constellation diagram for modulation recognition after preprocessing. The preprocessing includes procedures such as sampling time synchronization and symbol rate synchronization. Note that there are blind estimation processes in the preprocessing, including symbol rate estimation, frequency offset estimation, and so on. The author in [2] directly assumes that under the condition of cooperative communication, the phase jitter has been eliminated, and the symbol synchronization has been completed. The convolutional neural network is then adopted for modulation recognition.

From the aforementioned modulation recognition methods, the influence of different symbol rates, frequencies, and phase offsets is eliminated through the receiving synchronization process under both cooperative communication condition and blind receiving condition. However, in real application, due to the inaccurate estimation of blind receiving parameters, the input signal samples for modulation recognition still have large phase and frequency offsets and different time scales. Existing deep learning modulation recognition methods have not taken the mentioned situations into consideration, which can lead to a decrease in the modulation accuracy. A novel modulation recognition method based on the spatial transformation network is proposed in this paper. In the network, prior models for synchronization in communication are introduced, and the priori models are realized through the spatial transformation subnetwork, which can reduce the phase and frequency offsets. The influence of different time scales or the number of symbols on modulation recognition can also be reduced. Through the simulation dataset generated adopting gnuradio [9, 10], the experiments are carried out. The difference between the proposed method and the spatial transformer-based method in [11] is threefold. (1) The structure of the parameter regression module is different. The paper has adopted both time and frequency-domain samples as input, which have better ability to extract features from both domains. (2) The paper has given more details about the spatial transformer-based model. (3) For the model, the training process has added the supervision of symbol rate according to the baseline symbol rate model. Therefore, the

proposed method has the ability for symbol rate estimation. Overall, the paper can be regarded as an improvement of [11], which has better parameter regression capability and ability for symbol rate estimation. The results show that in the presence of different symbol rates and different frequency offsets and phase offsets, the proposed method has a recognition rate of 8.3%, 4.9%, and 5.2% higher compared with the traditional CNN, ResNet, and CLDNN, but the training convergence time has reduced by 3.5%, 27%, and 85%, respectively.

2. Methods

The overall structure of the proposed method is shown in Figure 1. In the proposed network, the spatial transformation subnetwork is inserted into the traditional convolutional neural network, where other parts are similar to the traditional convolutional neural network. The structure of the spatial transformation subnetwork is also shown in Figure 1, which is mainly composed of three substructures: the parameter regression estimation module, the time compensation module, and the phase frequency offset compensation module. Among them, the parameter regression estimation module is composed of a few convolutional layers, the input of which is the feature extracted by the previous layer. The output of the last convolutional layer can output some parameter estimations. In our implementation, these parameters include time scaling parameters, frequency offset, and phase offset parameters. The dimension of the output parameters is 5, where the time compensation-related dimension is 3, and the phase frequency offset compensation-related dimension is 2. The design of output parameters is related to the subsequent parameter-based transformation model, which will be discussed in detail in the following section. The time compensation module adopts the time-related parameters to perform the corresponding transformations on the input samples, thereby compensating for problems introduced by different number of symbols in the signal sample. For the frequency and phase offsets, they are compensated according to the phase and frequency offsets estimations obtained by the parameter regression module. The compensated samples are then adopted to identify the modulation type. The following is a detailed discussion of the proposed spatial transformation-based method.

2.1. The Parameter Regression Module. The function of the parameter regression estimation module is to estimate the parameters for subsequent transformation. The estimated parameters can transform the samples accordingly to compensate for the time offset, scale changes, frequency offset, and phase offset of the input signal samples. In modulation recognition, the mentioned parameters have an impact on the accuracy of modulation recognition. The cost function of the recognition network also has a correlation with the mentioned parameters. Therefore, the parameter can be estimated through network training. In addition, in order to enable more direct extract of frequency-based

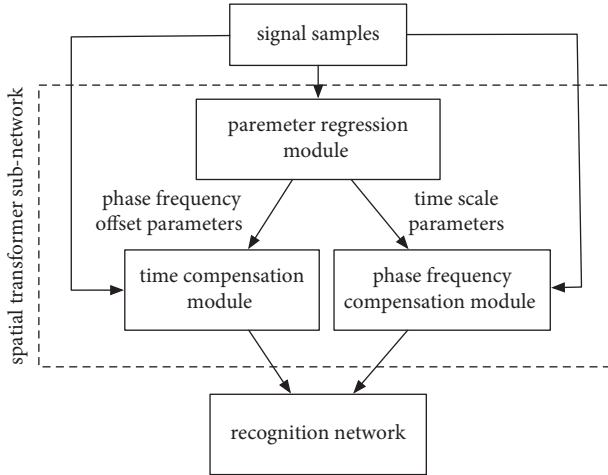


FIGURE 1: The overall structure of the method.

features, the frequency spectrum of the signal is also adopted as input to the network. This can better guide the network to learn the frequency-domain-based features, thereby avoiding the time-frequency domain conversion learning in the neural network. The structure of the parameter regression estimation module is shown in Figure 2. The time-domain and frequency-domain-based signals go through two feature extraction networks, respectively. Then, the obtained feature vectors are joined to form a larger feature vector. Another feature extraction network is added with input of the vector to obtain the estimation of the 5-dimension transformation parameters. Features A, B, and C are made of two convolutional layers and an all connected layer. In the model, there are in total 9 layers in depth. However, there are parallel layers in the model.

2.2. The Time Compensation Module. In real scenario modulation recognition applications, the signal samples may have different time scales and frequency offsets. For the explanation convenience, it is assumed that the signal samples have been already converted to baseband and the signal-to-noise ratio is high. For neural network input, although the length of the signal samples or the number of sampling points are the same, the following situations may exist: (1) due to different SPS (samples per symbol), there are different numbers of symbols in the signal sample with the same number of sampling point; for example, if one signal sample has SPS twice as the other, the number of symbols is also twice; (2) the number of symbols in the signal samples is the same, but the position of the signal starting point is different. The difference can be regarded as the offset in time. The factors of time offset and different numbers of symbols in the signal samples may cause the decrease of the modulation recognition rate. Therefore, after the time offset and time scale transformation parameters are obtained through parameter regression, the corresponding model is adopted to transform the signal samples. The transformation can compensate the time scale and offset, thereby reducing the effects in recognition rate. The transformation herein is

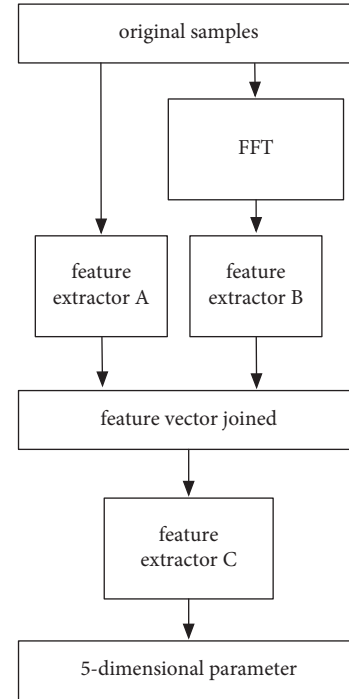


FIGURE 2: The structure of the parameter regression module.

based on the two-dimension affine transformation in the field of image processing [11, 12]:

$$\begin{bmatrix} x^s \\ y^s \end{bmatrix} = \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 \\ \theta_4 & \theta_5 & \theta_6 \end{bmatrix} \begin{bmatrix} x^t \\ y^t \\ 1 \end{bmatrix}, \quad (1)$$

where x^s, y^s denote the pixel position of the original image, x^t, y^t denote the positions of the transformed image, and $\theta_1 \sim \theta_6$ are the transformation parameters. In the aforementioned transformation, translation, rotation, and scaling are all included. In our application, since there is only translation and scaling in time, the transformation model can be simplified as follows:

$$\begin{bmatrix} x^s \\ y^s \end{bmatrix} = \begin{bmatrix} \theta_1 & 0 & \theta_2 \\ 0 & \theta_3 & 0 \end{bmatrix} \begin{bmatrix} x^t \\ y^t \\ 1 \end{bmatrix}. \quad (2)$$

After simplification, only 3 parameters are adopted to for translation (representing the offset in time) and scaling (representing the change in scale due to the difference in symbol rate). Note that x^s, y^s denote the position of the input signal feature and x^t, y^t denote the position of the transformed signal feature. After the transformation, the corresponding original coordinate position may be a decimal number and can exceed the range of feature dimension. Therefore, in practical applications, it is also necessary to interpolate for the value of the feature in the following equation, and the value of at position l after interpolation can be written as

$$V_l^t = \sum_i^I \sum_j^J V_{ij}^s k(x_l^s - i; \Phi_x) k(y_l^s - j; \Phi_y), \quad (3)$$

where V_l^t represents the value at the corresponding position l after interpolation, I and J denote the dimensions of the features, V_{ij}^s denotes the value of the input feature at the position of ij , $k(\cdot)$ represents the metric between two variables defined by the kernel function, Φ_x and Φ_y represent the corresponding kernel functions, and x_i^s and y_j^s denote the corresponding position of the input after the simplified affine transformation at position l . Note that here the position may be decimal. Generally speaking, if bilinear transformation is selected as the corresponding kernel function, the above equation can be written as

$$V_l^t = \sum_i \sum_j V_{ij}^s \max(0, |x_i^s - i|) \max(0, |y_j^s - j|). \quad (4)$$

The above equation can be regarded as the weighted average of the values near the position before the simplified affine transformation [13, 14]. Overall, the corresponding position can be obtained through affine transformation, and the value of the corresponding position can be obtained through bilinear interpolation. According to the mentioned processes, the feature output after time translation and scaling can be obtained. Figure 3 shows the processing flow of the time compensation module. It can be seen that the position of the output domain x^t, y^t is subjected to affine transformation according to the estimated parameters to obtain the position in the corresponding input domain x^s, y^s firstly. Then, the values at different positions in the output domain can be obtained through interpolation. Since the calculation of the bilinear transformation is differentiable, the time compensation module can be trained through the network.

For the training process, the supervision of symbol rate is added for enhancing parameter regression. As a matter of fact, the parameters of θ_1 and θ_3 represent the time scaling, which is symbol rate related. If the symbol rate is equal to the baseline symbol rate, then the parameter should be

$$\begin{aligned} \theta_1 &= 1, \\ \theta_3 &= 1. \end{aligned} \quad (5)$$

During the training, we added the following loss to the overall loss function:

$$|\theta_1 - \hat{\theta}|^2 + |\theta_3 - \hat{\theta}|^2, \quad (6)$$

where θ_1 and θ_3 represent the estimated parameters and $\hat{\theta}$ represents the actual relative symbol rate. Note that for each input sample, the value of $\hat{\theta}$ is different. Then, in the testing process, after the parameters θ_1 and θ_3 are acquired. The actual symbol rate can be estimated as

$$\frac{(\theta_1 + \theta_3) * \theta_{\text{base}}}{2}, \quad (7)$$

where θ_{base} denotes the baseline symbol rate signal.

2.3. The Phase Offset Compensation Module. The phase and frequency offset compensation is more intuitive, and it is processed directly according to the following equation:

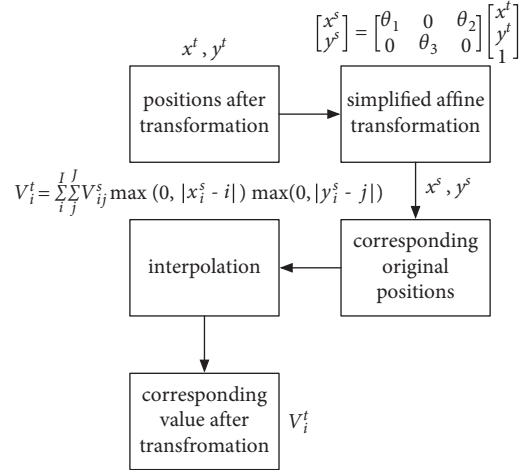


FIGURE 3: The processing flow of the time compensation module.

$$x_{\text{out}} = x_{\text{in}} * \exp((n\theta_4 + \theta_5)j). \quad (8)$$

Assuming that the processing of the above formula is complex value based, where x_{in} is the input and x_{out} is the output, the parameter θ_4 represents the frequency offset estimation, the parameter θ_5 represents the phase offset estimation, and n represents the time. In real implementation, since the input data are IQ time domain based, which can be regarded as the real and imaginary parts of complex values, the actual transformation can be written as

$$x_{\text{out}} = (x_{\text{in}}^I + jx_{\text{in}}^Q) * (\cos(n\theta_4 + \theta_5)) + j \sin(n\theta_4 + \theta_5). \quad (9)$$

The corresponding real part output is

$$x_{\text{out}}^I = x_{\text{in}}^I * \cos(n\theta_4 + \theta_5) - x_{\text{in}}^Q * \sin(n\theta_4 + \theta_5). \quad (10)$$

The corresponding imaginary part output is

$$x_{\text{out}}^Q = x_{\text{in}}^Q * \cos(n\theta_4 + \theta_5) + x_{\text{in}}^I * \sin(n\theta_4 + \theta_5). \quad (11)$$

As the transformation is also differentiable, the network can be trained. For the frequency compensation module, the number of n can be estimated. Adopting the estimation of n , the relative SPS can be estimated according to the original samples.

3. Experiments

In this paper, the spatial transformation network is adopted for modulation recognition, which takes into account different time scales, frequencies, and phase offsets in the model. The following describes the experimental process and results from the aspects of experimental dataset generation and method comparisons.

3.1. Dataset Generation. In order to verify the effectiveness of the proposed method, the open-source software radio platform gnuradio [15] is adopted for generating the dataset. The generated dataset contains 11 different modulation

types, including digital modulation types for BPSK, QPSK, 8PSK, PAM4, QAM16, QAM64, and CPFSK and analog modulation types for GFSK, AM-DSB, AM-SSB, and FM. When generating the dataset, the methods in [9] are adopted for reference. The source for the dataset includes real text and audio sources. For generation of modulated signals, including BPSK, QPSK, 8PSK, PAM4, QAM16, and QAM64, a root raised cosine filter is adopted to shape the transmitted signal to obtain a baseband modulated signal. For the GFSK signal, Gaussian filter is adopted for shaping, and the analog frequency modulation signal is adopted to obtain the corresponding two frequency peaks. After the modulated signals are generated, they are truncated in time to generate signal samples, in order to obtain samples of the same length. The dimension of the sample is 2×128 , where 2 denotes the IQ channels and 128 denotes number of sampling points in time. In the process of dataset generation, different from the dataset in [9], the different frequency offset and time scale changes are added. For frequency offsets, the related frequency offset parameter is set in the channel function `dynamic_channel_model` in `gnuradio`. Figure 4 shows the normalized frequency comparison of the same 8PSK signal with and without frequency offset. It can be seen that the frequency offset of the baseband signal can be generated by the frequency offset parameter in the channel model. In practical applications, such frequency offsets are prevalent due to inaccurate signal detection. In our experiment, we have adopted the open-source RadioML dataset for modulation recognition. In the dataset, we have additional added different frequency offsets and time scales adopting the `gnuradio` software. In the software, there are multirate signal processing modules for decimation and interpolation.

The scale changes are generated by changing the SPS accordingly. In Figure 5, the signal sample for the 8PSK modulation type is shown, and the corresponding dimension of the sample is 2×128 . In the figure, the upper one has set SPS to 4, and the bottom one has set the SPS to 6. It can be seen that the two signal samples contain different numbers of symbols, which are 6 and 4, respectively. This figure can intuitively show the problem of different time scales under the same modulation mode. The problems of different SPS are common in actual situations due to different signal bandwidths.

After the dataset is generated according to the above method, each signal sample is normalized according to the sample energy. Then, the dataset is split randomly into 50% for training and 50% for testing.

3.2. Method Comparisons. In order to fully illustrate the effectiveness of the proposed method, the recognition rate of the proposed method is compared with that of several classical neural network-based methods. The recognition rate of different methods under different signal-to-noise ratio conditions is shown in Figure 6. From the statistics in Table 1, it can be seen that when the signal-to-noise ratio is greater than 0, the recognition rate of the proposed method in this paper is 8.0%, 5.8%, and 4.6% higher than that of the traditional CNN, ResNet, and CLDNN, respectively [16].

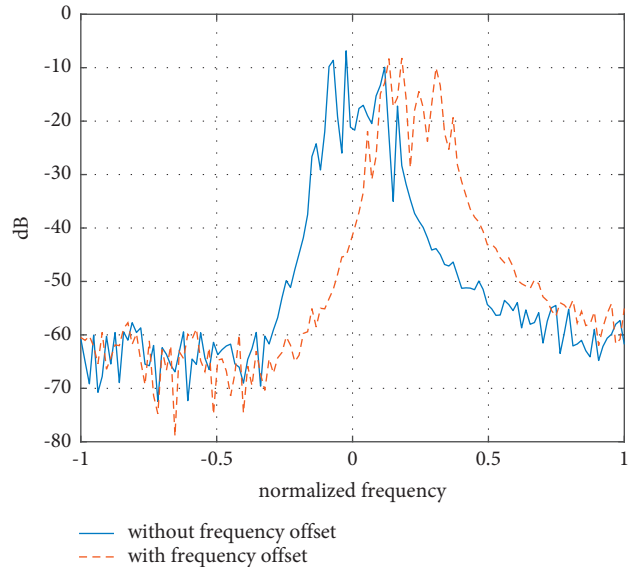


FIGURE 4: The comparison of the spectra with and without frequency offset for the 8PSK signal (noting that here the FFT is with 128 samples).

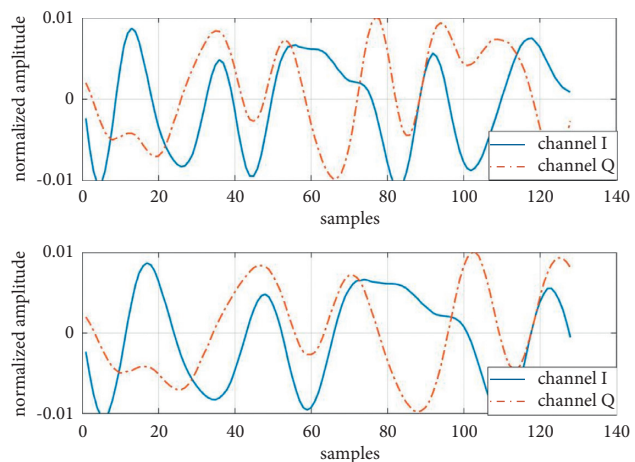


FIGURE 5: 8PSK modulated signal samples of different time scales (the SPS are 4 in the upper figure and 6 in the bottom figure).

Table 2 lists the comparisons of the total number of parameters and the training convergence time between the proposed method in this paper and the traditional CNN, ResNet, and CLDNN. It can be seen that, compared with the classic CNN, the proposed method has increased the number of network parameters by about 300% due to the addition of the parameter regression module, the time compensation module, and the phase frequency offset compensation module. However, as the proposed network structure is designed with a priori time scale change and frequency phase offset model, it is easier to train. The training time required for convergence is reduced by 4.5% and 80.7% compared to the ResNet and CLDNN, which has fully demonstrated the effectiveness of the proposed method.

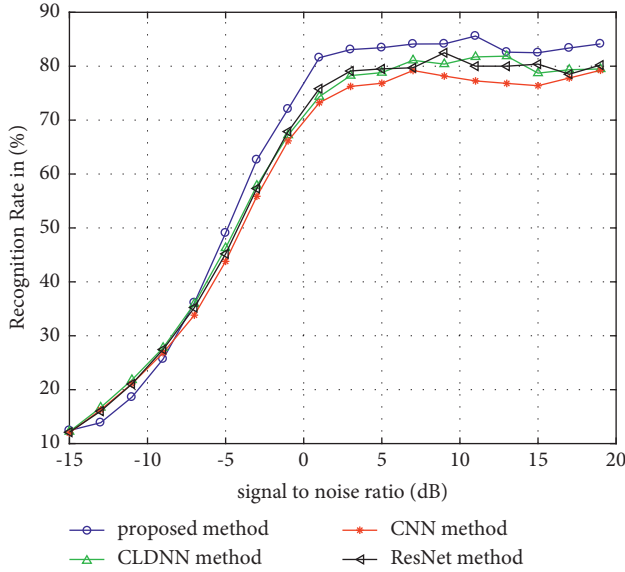


FIGURE 6: Comparison of correct recognition rates of different methods under different signal-to-noise ratio conditions.

TABLE 1: Comparisons of average recognition rate of different methods when SNR>0.

Method	Recognition rate (%)	Increase (%)
Proposed	84.5	—
CLDNN	80.8	4.6
CNN	78.2	8.0
ResNet	79.9	5.8

TABLE 2: Comparisons of the number of parameters and training time for different methods.

Method	Parameter number	Training time (s)
Proposed	4.9×10^6	319
CLDNN	5.8×10^5	1654
CNN	1.6×10^6	297
ResNet	1.8×10^6	334

3.3. SPS Estimation. As mentioned, from the estimation of n in the phase and frequency offset compensation module, the SPS of the signal sample can be estimated. The SPS estimation accuracy of the proposed method can reach 98.8%. This is another advantage over other deep neural network-based methods, which are not able to extract knowledge on SPS on the signal samples. Noting that as the estimated n can be a decimal number, the original estimated SPS can also be decimal. For the accuracy statistics, the estimated SPS are rounded. For the training process, the supervision of symbol rate is added for enhancing parameter regression. For the experiment, we have chosen the SPS 4 as the baseline symbol rate model. For both training and testing, the symbol rate ranges from 2 to 8. Then, ground truth parameter of θ_1 and θ_3 should be in the range of 0.5 to 2 accordingly.

4. Conclusions

Blind signal modulation recognition has great application potential in both civil and military uses. For real scenario modulation recognition applications, signals of the same modulation type may have encountered the effects of different time scales and frequency offsets. A modulation recognition method based on spatial transformation network is proposed in this paper. Compared with the classic CNN recognition network, a parameter regression estimation module, a time compensation module, and phase frequency offset compensation module are added. Among them, the parameter regression module can estimate the time scale transformation parameters (3 dimensions) and the frequency and phase offset parameters (2 dimensions). The time compensation module and the phase frequency offset compensation module can perform the corresponding compensational transformations on the original signal samples according to the estimated parameters. Through the open-source software radio gnuradio, the experimental dataset is generated. The dataset includes signal samples of 11 modulation types with different signal-to-noise ratios, different SPS, and different frequency offsets. Experiments adopting the generated dataset prove that compared with the traditional CNN, ResNet, and CLDNN, the recognition rate of the proposed method has increased by 8.0%, 5.8%, and 4.6%, respectively, when the signal-to-noise ratio is greater than 0. Moreover, the proposed network in this paper is easier to train, and the training time required for convergence has reduced by 4.5% and 80% compared with the ResNet and CLDNN, respectively.

Data Availability

The RadioML dataset is publicly available.

Conflicts of Interest

The author declares that there are no conflicts of interest.

Acknowledgments

This study was supported by 2020 Science and Technology Project of Quanzhou City (2020C011R), 2020 Undergraduate Education and Teaching Reform Research Project of Fujian Province (FBJG20200136), and Quanzhou High-Level Talent Team Project (no. 2019CT003).

References

- [1] Z. Zhu and A. K. Nandi, *Automatic Modulation Classification: Principles, Algorithms and Applications*, Wiley Publishing, Hoboken, New Jersey, 2015.
- [2] Z. K. Lei, *Research on Deep Learning Based Modulation Recognition Technologies*, University of Electronic Science and Technology of China, China, Sichuan, 2019.
- [3] Y. Lin, Y. Tu, Z. Dou, and Z. Wu, "The application of deep learning in communication signal modulation recognition," in *Proceedings of the 2017 IEEE/CIC International Conference on Communications in China (ICCC)*, October 2017.

- [4] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," Edited by C. Jayne and L. Iliadis, Eds., in *Proceedings of the Engineering Applications of Neural Networks. EANN 2016*, vol. 629Cham, September 2016.
- [5] N. E. West and T. O'Shea, "Deep architectures for modulation recognition," in *Proceedings of the 2017 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pp. 1–6, Baltimore, MD, USA, March 2017.
- [6] T. J. O'Shea, N. West, M. Vondal, and T. C. Clancy, "Semi-supervised radio signal identification," in *Proceedings of the 2017 19th International Conference on Advanced Communication Technology (ICACT)*, pp. 33–38, PyeongChang, Korea (South), February 2017.
- [7] C. Yang, Z. He, Y. Peng, Y. Wang, and J. Yang, "Deep learning aided method for automatic modulation recognition," *IEEE Access*, vol. 7, Article ID 109063, 2019.
- [8] K. Karra, K. Scott, and J. Petersen, "Modulation recognition using hierarchical deep neural networks," in *Proceedings of the 2017 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, March 2017.
- [9] T. J. O'Shea and N. West, "Radio machine learning dataset generation with GNU radio," *Proceedings of the GNU Radio Conference, [S.l.]*, vol. 1, no. 1, pp. 1–5, 2016.
- [10] R. Gandhiraj and K. P. Soman, "Modern analog and digital communication systems development using GNU Radio with USRP," *Telecommunication Systems*, vol. 56, no. 3, pp. 367–381, 2014.
- [11] T. J. O'Shea, L. Pemula, D. Batra, and T. C. Clancy, "Radio transformer networks: attention models for learning to synchronize in wireless systems," in *Proceedings of the 2016 50th Asilomar Conference on Signals, Systems and Computers*, pp. 662–666, Pacific Grove, CA, November 2016.
- [12] E. W. Weisstein, "Affine transformation," 2004, <https://mathworld.wolfram.com>.
- [13] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2 (NIPS'15)*, MIT Press, Cambridge, MA, USA, December 2015, <https://arxiv.org/abs/1506.02025>.
- [14] L. Tan, Z. Li, and Q. Yu, "Deep Face Attributes Recognition Using Spatial Transformer Network," in *Proceedings of the 2016 IEEE International Conference on Information and Automation (ICIA)*, pp. 1928–1932, Ningbo, August 2016.
- [15] *Gnuradio Software. Website*, <https://www.gnuradio.org>.
- [16] T. Zhang, C. Shuai, and Y. Zhou, "Deep learning for robust automatic modulation recognition method for IoT applications," *IEEE Access*, vol. 8, Article ID 117689, 2020.