The Journal of Engineering

IET International Radar Conference (IRC 2018)

Radar HRRP recognition based on CNN

eISSN 2051-3305
Received on 25th February 2019
Accepted on 5th April 2019
E-First on 23rd September 2019
doi: 10.1049/joe.2019.0725
www.ietdl.org

Jia Song[1,2], Yanhua Wang[1,2] ✉, Wei Chen[1,2], Yang Li[1,2], Junfu Wang[3]

[1]Radar Research Laboratory, School of Information and Electronics, Beijing Institute of Technology, Beijing, People's Republic of China
[2]Beijing Key Laboratory of Embedded Real-time Information Processing Technology, Beijing Institute of Technology, Beijing, People's Republic of China
[3]Beijing Racobit Electronic Information Technology Co., Ltd., Beijing, People's Republic of China
✉ *E-mail: wyhlucky@bit.edu.cn*

Abstract: In this study, ground target recognition based on one-dimensional convolutional neural network (CNN) is studied by exploiting the targets' high-resolution range profiles (HRRPs). Contrary to conventional methods which need feature extraction artificially, CNN can automatically discover features for classification. The authors propose a multi-channel CNN architecture that can be applied on diverse forms of HRRP such as amplitude, complex, spectrum etc. Experimental results demonstrate the superiorities of the proposed method over conventional methods based on handcrafted features and single-channel CNN in terms of recognition accuracy. Visualisation of the 'deep features' shows higher separability than handcrafted features, thus providing an insight into its effectiveness in exploiting the intrinsic structures.

1 Introduction

High-resolution range profile (HRRP) is one of the common approaches in radar automatic target recognition (RATR) [1] for its advantages in data acquisition and processing. HRRP is the coherent summation of the complex responses of the scattering centres. It represents the projection of the three-dimensional (3D) scattering structure of the target onto the radar line-of-sight and thus contains geometric structural information that can be used for RATR [2].

Most of existing HRRP recognition methods are based on machine learning and involve two steps: feature extraction and classification [3–5]. For decades, the success of an HRRP recognition system requires considerable domain expertise and careful engineering skills to design a feature extractor that transforms the raw HRRP into a suitable representation. Typical features include scale, spectrum and polarisation [4–6]. Although these features have shown good performance in specified applications, it is still a big challenge to design satisfactory features for a given task. Therefore, there exists a strong demand to develop a universal feature extractor that requires few handcrafted efforts and can be applied in various conditions.

Last few years have witnessed the booming and success of deep learning or deep neural networks, which offers an end-to-end representation capability and thus can automatically discover the intrinsic patterns in data. Convolutional neural network (CNN) is a widely applied structure in deep learning. It has brought a huge revolution in computer vision-related tasks since it achieved a great leap in ImageNet contest. It is a special type of deep neural network for processing data with grid-like topology [7]. In RATR, CNN has been applied in synthetic aperture radar (SAR)/inverse SAR (ISAR) recognitions and shown promising performances [8, 9]. Few CNN-based HRRP recognition methods have been reported in the literature. In [10], a two-layer CNN is built to recognise airborne targets for multistatic radar. The network takes the normalised HRRP as the input and predicts the label by averaging the results of all radars. In [11], a 2D HRRP data matrix, which is formed by stacking the replica of a single HRRP, is used as the input of a CNN to recognise ship targets. Unlike SAR/ISAR recognition, CNN in HRRP recognition has not been thoroughly exploited and evaluated.

In this paper, CNN is used for ground target HRRP recognition. We propose a multi-channel CNN architecture that can be applied on diverse forms of HRRP such as real, complex, spectrum, polarised and sequence. With the designed network, there is no need to manually construct a feature extractor. Comparing to the single-channel form, the proposed method shows considerable improvement in recognition accuracy. To provide insight for explaining the effectiveness of CNN, the 'deep features' are compared with the handcrafted features and show a better separability.

The rest of this paper is structured as follows. Section 2 introduces the principle and component of CNN. In Section 3, the proposed method is presented; then, the experimental results are illustrated in Section 4. Finally, this paper is summarised in Section 5.

2 Convolutional neural network

2.1 Structure

Fig. 1 shows a general CNN structure for 1D input. It is composed of two types of layers: convolution layers and pooling layers. The convolutional layer is the core block on CNN. It consists of a set of learnable filters, whose parameters will be updated in the back-propagation stage to learn the representations of input. Each filter produces an activation of the input, also known as feature map. The pooling layer downsamples the resulting feature maps to reduce the size of the representations.

2.2 Convolutional layer

In convolutional layer, the input (or feature maps from previous layers) is filtered by convolution operation with learnable kernels. The outputs of the kernels go through an activation function (AF) to form the output feature maps. Mathematically, the operation is formulated as

$$\boldsymbol{O} = f(\boldsymbol{x} \times \boldsymbol{k}) \tag{1}$$

where $\boldsymbol{x}$ is the input with d-channels and $\boldsymbol{k}$ is convolutional kernels. If $\boldsymbol{O}\ (t,i)$ is the tth feature map of input $\boldsymbol{x}$ at position i and $\boldsymbol{k}_t$ is the tth filter with length kw, the multi-channel convolution operation can be denoted as

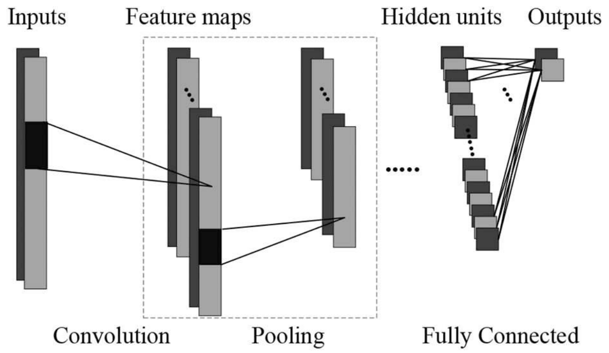

Fig. 1 *Structure of CNN*

$$\boldsymbol{O}(t,\, i) = f\left(\sum_{d}\sum_{j\, =\, i\, -\, (kw\, -\, 1)/2}^{i\, +\, (kw\, -\, 1)/2}\boldsymbol{x}(j,\, d)\boldsymbol{k}_t(j,\, d)\right) \tag{2}$$

It is the AF that introduces universal representation capability. Common choices of AF include sigmoid function (Sigmoid), hyperbolic tangent (Tanh) and rectified linear units (Relu)

$$\mathrm{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \tag{3}$$

$$\mathrm{Tanh}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{4}$$

$$\mathrm{Relu}(x) = \max\,(x,\, 0) \tag{5}$$

Owing to the non-saturating non-linearity, Relu works faster in network training. Moreover, it mitigates the gradient vanishing problem and allows networks to obtain sparse representations easily [12].

2.3 Pooling layer

Following a convolutional layer, a pooling layer is usually inserted to reduce the size of the output feature map and thus reduce the number of parameters in the next layer. It is known that the pooling operation helps to alleviate the sensitivity to the data shift. The input feature map is divided into non-overlapping patches with the same length. The pooling function is operated on each patch to generate a single value. There are two pooling functions: max pooling and average pooling. In max pooling, the highest value is selected from the patch, while on average, pooling the mean value of the patch is computed as the output.

2.4 Classification layer

The feature map generated by the final convolutional layer is considered as the 'deep features' of the input and used for classification. In theory, any classifier can be used, but a fully connected network followed by a softmax function is a common choice. There are no strict rules on the number of layers which are incorporated in the network model, but the number of neurones in the last layer should be same as the number of classes. Then, the softmax function squashed the output to a vector of values between zero and one that sum to one, which can be interpreted as predicted probabilities for each class

$$\hat{y}_i(z) = \frac{\mathrm{e}^{z_i}}{\sum_{i\,=\,1}^{C} e^{z_i}}, \quad \text{for} \quad i = 1,\, \ldots,\, C \tag{6}$$

where z is the last fully connected layer output and C is the number of classes.

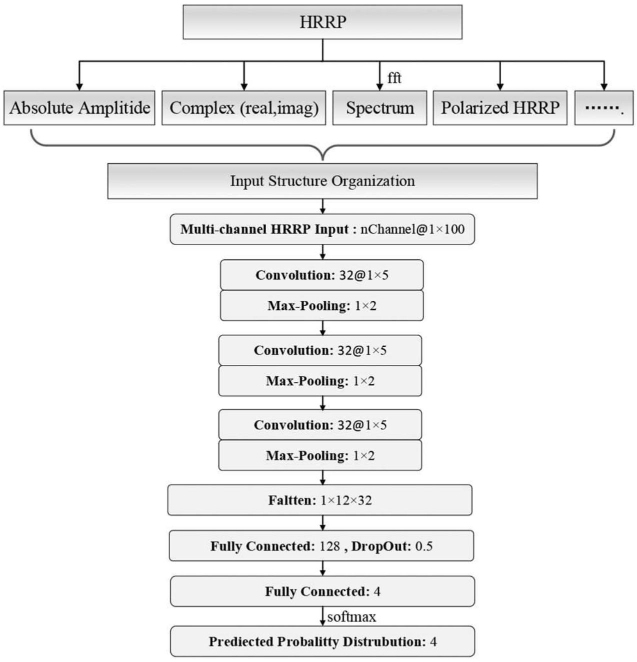

Fig. 2 *Proposed multi-channel CNN architecture*

2.5 Loss function

The loss function measures the distance between the predicted value and the true label. In CNN, cross-entropy is commonly used in classification tasks

$$L = -\sum_i y_i \log \hat{y}_i \tag{7}$$

where y_i is the true label and $\hat{y}_i$ is the predicted value.

2.6 DropOut

DropOut refers to eliminating some units in a neural network to prevent overfitting. It provides a way of approximately combining many different neural network architectures efficiently [13]. It is implemented by turning off some neurones according to the probability that has been set by users. When a neurone is dropped, it will not contribute to the training process in both the forward and backward phases of back propagation.

3 Proposed method

The proposed multi-channel CNN architecture for diverse forms of HRRP is shown in Fig. 2. It is a pragmatic method that can explore the correlative HRRP data. Multi-channel CNN is inspired by three-colour-channel red, green and blue image data, which contain more information than the grey-scale images, so it is same for HRRP. Comparing to the single-channel form, it can take advantage of diverse forms of HRRP, hence enrich the information of input to achieve better performance.

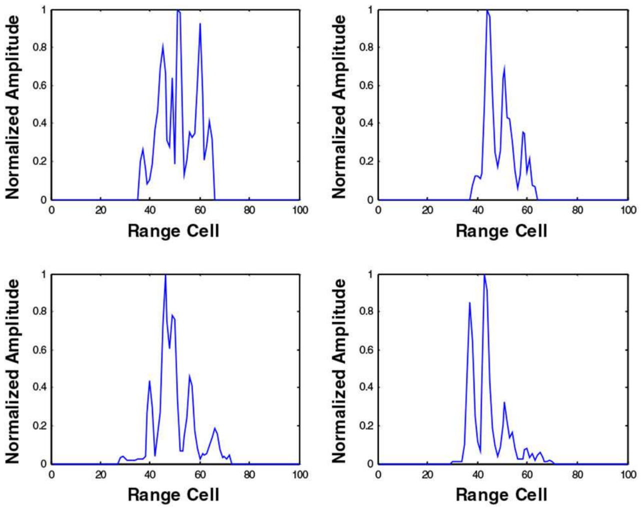

Fig. 3 *Examples of HRRP of four different targets*

In the input structure organisation, we can choose the combination of diverse forms. For instance, complex HRRP data of real and imaginary parts or time and frequency HRRP data. We will test performances of different types of combinations in the following section. When building network, The CNN architecture determined experimentally consists of three convolutional layers and two fully connected layers with DropOut.

In convolutional layer, the core operation is multi-channel convolution. Input data $\boldsymbol{x}$ is a matrix with three dimensions, the first dimension is the number of HRRP frames, the second dimension is the length of input HRRP and the last dimension is the channel numbers of input. The convolutional layer has 32 kernels with a size of 1 × 5, the output $\boldsymbol{O}_{\rm c}$ is

$$\boldsymbol{O}_{\rm c} = (f(\boldsymbol{x} \times \boldsymbol{k}_1);\ \ldots;\ f(\boldsymbol{x} \times \boldsymbol{k}_5)) \qquad (8)$$

We use Relu as the AF $f(\cdot)$ in the network.

In between successive convolutional layers, we use max pooling of stride of 1 × 2 to downsample the representations. Max pooling is most commonly used pooling method and has a level of invariance when input change in position. The output of pooling layer is

$$\boldsymbol{O}_{\rm p} = \text{max-pooling}(\boldsymbol{O}_{\rm c}) \qquad (9)$$

After three stacks of convolutional and max-pooling layers, the feature maps are flattened to connect with fully connected layer, denoted the flattened feature vectors as $O_{\rm flat}$. we use DropOut on the first fully connected layer. If it has weights $\boldsymbol{w}$ and bias $\boldsymbol{b}$, and there are 128 neurones in this layer, applying DropOut on it

$$\boldsymbol{O}_{\rm fc} = t \odot f(\boldsymbol{w}\boldsymbol{O}_{\rm flat} + \boldsymbol{b}) \qquad (10)$$

where t is an N-dimensional vector of Bernoulli variable

$$P\{t_i = m\} = p^m(1-p)^{1-m}, \quad m = 0,\ 1(0 < p < 1,\ 1 \le i \le 128) \qquad (11)$$

Then, after all fully connected layers followed softmax function, there comes the output of CNN. It is a 4D vector (because there are four categories) with each element corresponding to the probability

$$\hat{y}_i = P(y = i|\boldsymbol{x}), \quad i = 1,\ \ldots,\ 4 \qquad (12)$$

The final predicted is the class with the maximum probability

$$y_{\rm pred} = r, \quad \text{where} \quad \hat{y}_{\rm r} = \max(\hat{y}) \qquad (13)$$

4 Experimental results

Table 1 Comparison of different recognition methods

	Test accuracy, %
SVM	72.97
RFs	86.23
GBDT	86.63
CNN	**94.30**

Bold value emphasises the proposed method achieves the best result.

Table 2 Confusion matrix of CNN

	Class 1, %	Class 2, %	Class 3, %	Class 4, %
class 1	**97.07**	2.03	0.90	0.00
class 2	3.39	**95.71**	0.90	0.00
class 3	7.45	2.03	**90.52**	0.00
class 4	5.19	0.23	0.68	**93.91**

Bold values represent the correct classification results for each class.

4.1 Experiment setup

The HRRP dataset was acquired with a radar with a bandwidth of 1250 MHz synthetic bandwidth. There were four classes of targets, whose HRRPs are shown in Fig. 3. At the training phase, each kind of target contains 6000 HRRP samples, whereas at the testing phase 2000 samples are used.

Before training, the pre-processing of HRRP is necessary. To avoid amplitude sensitivity, the HRRP is normalised to the interval of [0, 1]. To meet the requirement that the input of CNN should have a single size, a target chip of 100 range cells is extracted from the raw HRRP. Shifting is used for training data augmentation. In the training phase, we employ the stochastic gradient descent with Adam (adaptive moment estimation) and use early stopping to end the training before overfitting. The learning rate is 0.001 and batch size is 128.

4.2 Case 1: single-channel CNN

We first evaluated the recognition performance using the amplitude of HRRP. To measure feature learning ability of CNN, we compare its performance with three widely used recognition methods based on handcrafted features: support vector machine (SVM) with radial basis function (RBP) kernels, random forest (RF) and gradient boosting decision tree (GBDT). For these methods, we extracted 36 features from HRRP including structure, scale and spectrum features.

Table 1 shows the overall classification accuracy of different methods. As a single classifier, SVM has the simplest structure and obtains the lowest recognition rate. Ensemble methods such as RF and GBDT get higher accuracies, which suggest the effectiveness of combining several week classifiers. CNN achieves the best performance with an overall classification accuracy of 94.30%.

Table 2 shows the confusion matrix of CNN. It indicates that CNN works well for all targets. Even in the worst case (class 3), accuracy is still higher than 90%. It also can be seen that some samples of class 3 are incorrectly identified as class 1, it is because there are a few samples in class 3 that are similar to class 1.

To evaluate the pros and cons of the deep features learnt by CNN, we visualise features using 2D reduction method: PCA and *t*-SNE [14]. We also compared the deep features with the hand-designed features. The result is shown in Fig. 4. We can intuitively grasp the distribution of different categories of features through the visualisation. As can be seen, artificially designed features have a certain degree of separability. However, classes 3 and 4 have lot of overlaps. Features automatically learnt by CNN are more distinguishable than artificially designed features and each category is grouped. Thus, CNN can classify different categories more precisely and achieve better performance.

4.3 Case 2: multi-channel CNN

The above study illustrates the good performance of CNN for the single-channel HRRP data. Next, we evaluated the recognition

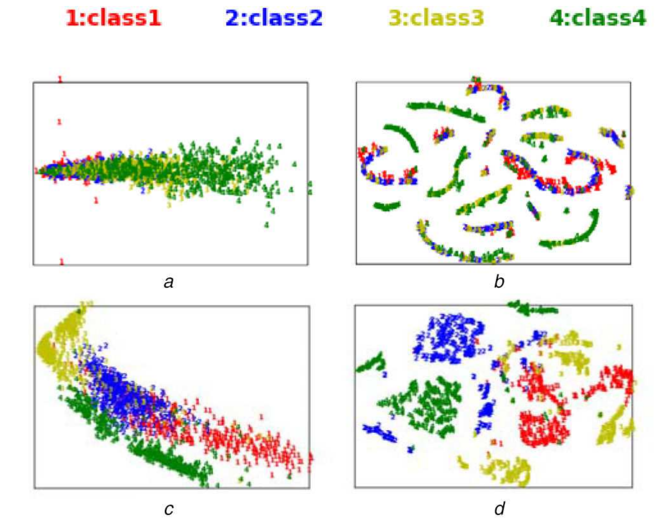

Fig. 4 *Visualisation of handcrafted features and deep features learnt by CNN*
(a) Hand-designed features by PCA, ***(b)*** Hand-designed features by *t*-SNE, ***(c)*** Deep features by PCA, ***(d)*** Deep features by *t*-SNE

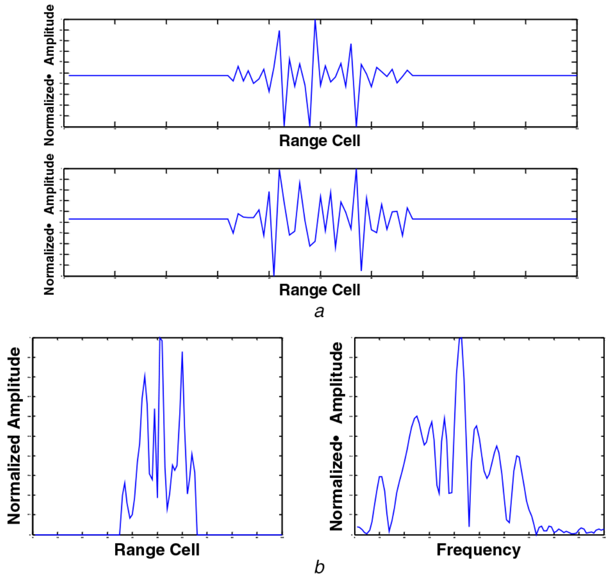

Fig. 5 *Sample in different input type*
(a) Real and imaginary parts of HRRP, ***(b)*** Amplitude and spectrum of HRRP

performance using the complex HRRP. We compared two possible multi-channel forms of HRRP: real and imaginary, amplitude and spectrum (Fig. 5). The single-channel CNN is used as baseline.

Method 1: real–imaginary representation. One channel is the real part of HRRP and the other is an imaginary part.

Method 2: time–frequency representation. One channel is the amplitude of HRRP and the other is its spectrum.

The results are shown in Table 3. It can be seen that the multi-channel CNN outperforms single-channel CNN. Time–frequency representation achieves better performances over real–imaginary representation. This is probably because the real–imaginary representation is sensitive to phases which can be easily disturbed by noises and clutters.

The deep features of the two types of multi-channel input CNNs are shown in Fig. 6. The learnt feature of time–frequency HRRP-based CNN is more distinguishable and grouped, which partially interprets its superiority as well.

Table 3 Comparison of different CNN schemes

	Test accuracy, %
baseline	94.30
method 1	95.78
method 2	98.30

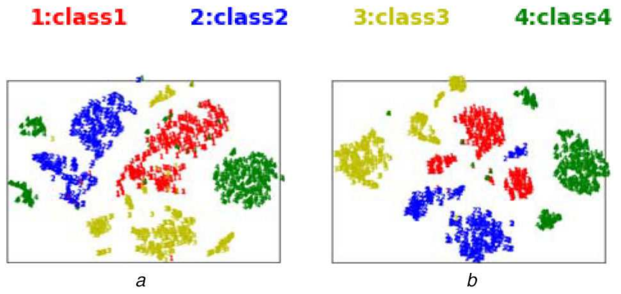

Fig. 6 *Visualisation of deep features learnt by multi-channel CNN*
(a) Deep features of method 1 by *t*-SNE, ***(b)*** Deep features of method 2 by *t*-SNE

5 Conclusion

We use CNN to recognise HRRP of ground targets, which can automatically discover features for classification, the CNN learnt ‘deep features’ perform better and are more distinguishable than handcrafted features. Moreover, we propose a multi-channel CNN architecture based on diverse forms of HRRP, which can enrich the information of input, and experimental results show that it has better performance than basic single-channel HRRP-based CNN.

6 Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant no. 61701026), the Chang Jiang Scholars Programme (Grant no. T2012122) and 111 Project of China (Grant no. B14010).

7 References

[1] Monti Guarnieri, A., Rocca, F.: ‘Options for continuous radar earth observations’, *Sci. China Inf. Sci.*, 2017, **60**, (6), p. 060301. doi: 10.1007/s11432-016-9067-7

[2] Wang, G.Y., Xie, J.L.: ‘Radar high resolution range profile target recognition based on T-mixture model’, IEEE Radar Conf. (RADAR), Kansas City, MO, USA, May 2011, pp. 762–767

[3] Wang, S.X., Li, J.B., Wang, Y.H.*, et al.*: ‘Radar HRRP target recognition based on gradient boosting decision tree’. Ninth Int. Congress on Image and Signal Processing, Datong, China, October 2016, pp. 1013–1017

[4] Du, L., Liu, H., Bao, Z.*, et al.*: ‘Radar HRRP target recognition based on higher order Spectra’, *IEEE Trans. Signal Process.*, 2005, **53**, (7), pp. 2359–2368. doi: 10.1109/TSP.2005.849161

[5] Liu, J., Fang, N., Xie, Y.J.*, et al.*: ‘Scale-space theory-based multi-scale features for aircraft classification using HRRP’, *Electron. Lett.*, 2016, **52**, (6), pp. 475–477. doi: 10.1049/el.2015.3583

[6] Wang, X.S.: ‘Status and prospects of radar polarimetry techniques’, *J. Radars*, 2016, **5**, (2), pp. 119–131

[7] Goodfellow, I., Bengio, Y., Courville, A.: ‘Convolutional networks’, in Brown, S. (Ed.): ‘*Deep learning*’ (MIT Press, Cambridge, MA, USA, 2017, 1st edn.), pp. 326–327

[8] Zhang, Z., Wang, H., Xu, F.*, et al.*: ‘Complex-valued convolutional neural network and its application in polarimetric SAR image classification’, *IEEE Trans. Geosci. Remote Sens.*, 2017, **55**, (12), pp. 7177–7188. doi: 10.1109/TGRS.2017.2743222

[9] Ding, J., Chen, B., Liu, H.W.: ‘Convolutional neural network with data augmentation for SAR target recognition’, *IEEE Geosci. Remote Sens. Lett.*, 2016, **13**, (3), pp. 364–368

[10] Lundén, J., Koivunen, V.: ‘Deep learning for HRRP-based target recognition in multistatic radar systems’. Radar Conf. IEEE, Philadelphia, PA, USA, May 2016, pp. 1074–1079

[11] Karabayır, O., Yücedağ, O.M., Kartal, M.Z.*, et al.*: ‘Convolutional neural networks based ship target recognition using high resolution range profiles’. 18th Int. Radar Symp. (IRS), Prague, Czech Republic, June 2017, pp. 1–9

[12] Glorot, X., Bordes, A., Bengio, Y.: ‘Deep sparse rectifier neural networks’. *Int. Conf. Artif. Intell. Stat.*, 2012, **15**, pp. 315–323

[13] Strivastava, N., Hinton, G.E., Krizhevsky, A.*, et al.*: ‘DropOut: a simple way to prevent neural networks from overfitting’, *J. Mach. Learn. Res.*, 2014, **15**, (1), pp. 1929–1958

[14] Maaten, L.V.D., Hinton, G.E.: ‘Visualizing data using *t*-SNE’, *J. Mach. Learn. Res.*, 2008, **9**, (2605), pp. 2579–2605