

Article

The Optimally Designed Variational Autoencoder Networks for Clustering and Recovery of Incomplete Multimedia Data

Xiulan Yu, Hongyu Li *, Zufan Zhang  and Chenquan Gan

School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China; yuxl@cqupt.edu.cn (X.Y.); zhangzf@cqupt.edu.cn (Z.Z.); gancq@cqupt.edu.cn (C.G.)

* Correspondence: s160101070@stu.cqupt.edu.cn; Tel.: +86-187-2563-3001

Received: 29 January 2019; Accepted: 13 February 2019; Published: 16 February 2019



Abstract: Clustering analysis of massive data in wireless multimedia sensor networks (WMSN) has become a hot topic. However, most data clustering algorithms have difficulty in obtaining latent nonlinear correlations of data features, resulting in a low clustering accuracy. In addition, it is difficult to extract features from missing or corrupted data, so incomplete data are widely used in practical work. In this paper, the optimally designed variational autoencoder networks is proposed for extracting features of incomplete data and using high-order fuzzy c-means algorithm (HOFCM) to improve cluster performance of incomplete data. Specifically, the feature extraction model is improved by using variational autoencoder to learn the feature of incomplete data. To capture nonlinear correlations in different heterogeneous data patterns, tensor based fuzzy c-means algorithm is used to cluster low-dimensional features. The tensor distance is used as the distance measure to capture the unknown correlations of data as much as possible. Finally, in the case that the clustering results are obtained, the missing data can be restored by using the low-dimensional features. Experiments on real datasets show that the proposed algorithm not only can improve the clustering performance of incomplete data effectively, but also can fill in missing features and get better data reconstruction results.

Keywords: feature learning; incomplete multimedia data; fuzzy c-means; variational autoencoder

1. Introduction

The rapid development of communication technologies and sensor networks leads to the increase of heterogeneous data. The proliferation of these technologies in communication networks also has facilitated the development of the wireless multimedia sensor network (WMSN) [1]. Currently, multimedia data on WMSNs are successfully used in many applications, such as industrial control [2], target recognition [3] and intelligent traffic monitoring [4].

Nowadays, multimedia sensors produce a great deal of heterogeneous data, which require new models and technologies to process, particularly neural computing [5], to further promote the design and application of WMSNs [6,7]. However, heterogeneous networks and data are often very complex [8,9], which consist of structured data and unstructured data such as picture, voice, text, and video. Because heterogeneous data come from many input channels in the real world, these data are typical multimodal data, and there is a nonlinear relationship between them [10]. Different modes usually convey different information [11]. For example, images have many details, such as shadows, rich colors and complex scenes, and use titles to display invisible things like the names of objects in the image [12]. Moreover, different forms have complex relationships. In the real world, most multimedia

data suffer from a lot of missing values due to sensor failures, measurement inaccuracy and network data transmission problems [13,14]. These features, especially incompleteness, lead to the widespread use of incomplete data in practical applications [15,16]. Lack of data values will affect the decision process of the application servers for specific tasks [17]. The resulting errors can be important for subsequent steps in data processing. Therefore, the recovery of data missing values is essential for processing big data in WMSNs.

As a fundamental technology of big data analysis, clustering divides objects into different clusters based on different similarity measures, making objects in the same cluster more similar to other objects in different groups [18,19]. They are commonly used to organize, analyze, communicate, and retrieve tasks [20]. Traditional data clustering algorithms focus on complete data processing, such as image clustering [21], audio clustering [22] and text clustering [23]. Recently, heterogeneous data clustering methods have been widely concerned by researchers [24–26]. In addition, many algorithms have been proposed—for example, Meng et al. optimized the unified objective function by an iterative process, and a spectral clustering algorithm is developed for clustering heterogeneous data based on graph theory [27]. Li et al. [28] proposed a high-order fuzzy c-means algorithm to extend the conventional fuzzy c-means algorithm from vector space to tensor space. A high-order possibilistic c-means algorithm based on tensor decompositions was proposed for data clustering in Internet of Things (IoT) systems [29]. These algorithms are effective to improve clustering performance for heterogeneous data. However, they can only obtain clustering results and lack further analysis of incomplete data low-dimensional features. Therefore, their performance is limited with the heterogeneous data in the WMSNs' big data environment. More importantly, other existing feature clustering algorithms do not consider data reconstruction and missing data. WMSN systems require different modern data analysis methods, and deep learning (DL) has been actively applied in many applications due to its strong data feature extraction ability [30]. Deep embedded clustering (DEC) learns to map from data space to low-dimensional feature space, where it optimizes the clustering objectives [31]. Ref. [32] shows the feature representation ability of variational autoencoder (VAE). VAE learns the multi-faceted structure of data and achieves high clustering performance [33]. In addition, VAE has a strong ability in feature extraction and reconstruction, and it can be a good tool for handling incomplete data.

Aiming at this research object, the variational autoencoder based high-order fuzzy c-means (VAE-HOFCM) algorithm is presented to cluster and reconstruction incomplete data in WMSNs in this paper. It can effectively cluster complete data and incomplete data and get better reconstruction results. VAE-HOFCM is mainly composed of three steps: feature learning and extraction, high-order clustering, and data reconstruction. First, the feature learning network is improved by using a variational autoencoder to learn the feature of incomplete data. To capture nonlinear correlations of different heterogeneous data, tensors are applied to form a feature representation of heterogeneous data. Then, the tensor distance is used as the distance measure to capture the unknown distribution of data as much as possible in the clustering process. The results of feature clustering and VAE output both affect the final clustering results. Finally, in the case of clustering results, the missing data can be restored by the low-dimensional features.

The rest of the paper is organized as follows: Section 2 presents related work to this paper. The proposed algorithm is illustrated in Section 3, and experimental results and analysis are described in Section 4. Finally, the whole paper is concluded in the last section.

2. Preliminaries

This section describes the variational autoencoder (VAE) and the fuzzy c-means (FCM), which will be useful in the sequel.

2.1. Variational Autoencoder

The variational autoencoder, which is a new method for nonlinear dimensionality reduction, is a great case of combining probability plots with deep learning [34,35]. Consider a dataset

$X = \{x_1, x_2, \dots, x_N\}$ which consists of N independent and identically distributed samples of continuous or discrete variables x . To generate target data x from hidden variable z , two blocks are used: encoder block and decoder block. Suppose that z is generated by some prior normal distribution $p_\theta = N(\mu, \sigma^2)$.

The true posterior density $p_\theta(z|x)$ is intractable. Approximate recognition model $q_\phi(z|x)$ as a probabilistic encoder. Similarly, refer to $p_\theta(x|z)$ as a probability decoder because, given the code z , it produces a distribution over the possible corresponding value x . The parameters θ and ϕ are used to represent the structure and weight of the neural network used. These parameters are adjusted as part of the VAE training process and are considered constant later. Minimize the true posterior approximation of the KL divergence (Kullback–Leibler Divergence). When the divergence of KL is zero, $p_\theta(z|x) = q_\phi(z|x)$. Then, the true posterior distribution can be obtained. The KL divergence of approximation from the true posterior $D_{KL}(q_\phi(z|x) \| p_\theta(z|x))$ can be formulated as:

$$\begin{aligned} (q_\phi(z|x) \| p_\theta(z|x)) &= \int_{-\infty}^{\infty} q_\phi(z|x) \log \frac{q_\phi(z|x)}{p_\theta(z|x)} dz \\ &= \log p_\theta(x) + D_{KL}(q_\phi(z|x) \| p_\theta(z)) - E_{q_\phi(z|x)}[\log p_\theta(x|z)] \\ &\geq 0, \end{aligned} \quad (1)$$

which can also be written as:

$$\log p_\theta(x) \geq -D_{KL}(q_\phi(z|x) \| p_\theta(z)) + E_{q_\phi(z|x)}[\log p_\theta(x|z)]. \quad (2)$$

The right half of the inequality is called the variational lower bound on the marginal likelihood of data x , and can be written as:

$$L(\theta, \phi; x) \geq -D_{KL}(q_\phi(z|x) \| p_\theta(z)) + E_{q_\phi(z|x)}[\log p_\theta(x|z)]. \quad (3)$$

The second term $E_{q_\phi(z|x)}[\log p_\theta(x|z)]$ requires estimation by sampling. A differentiable transformation $g_\phi(x, \varepsilon)$ of an auxiliary noise variable ε is used to reparameterize the approximation $q_\phi(z|x)$. Then, form a Monte Carlo estimates of $E_{q_\phi(z|x)}[\log p_\theta(x|z)]$:

$$E_{q_\phi(z|x)}[\log p_\theta(x|z)] = \frac{1}{M} \sum_{m=1}^M \log p_\theta(x|z^m), \quad (4)$$

where $z^m = g_\phi(x, \varepsilon^m) = \mu + \varepsilon^m \odot \sigma$, $\varepsilon^m \sim N(0, I)$ and m denotes the number of samples.

2.2. Fuzzy C-Means Algorithm (FCM)

The fuzzy c -means algorithm (FCM) is a typical soft clustering technique [36,37]. Given a dataset $X = \{x_1, x_2, \dots, x_N\}$ with N objects and m observations, fuzzy partition of set X into predefined cluster number c and the number of clustering centers denoted by $V = \{v_1, v_2, \dots, v_c\}$. Their membership functions are defined as $u_{ik} = u_{v_i}(x_k)$, in which u_{ik} denotes the membership of x_k towards the i th clustering center and c denotes. FCM is defined by a $c \times m$ membership matrix $U = \{u_{ik} | 1 \leq i \leq c; 1 \leq k \leq m\}$. FCM minimizes the following objective function [38,39] to calculate the membership matrix U and the clustering centers V :

$$J_m(U, V) = \sum_{k=1}^n \sum_{i=1}^c (u_{ik}) d^2(x_k, v_i), \quad (5)$$

where every u_{ik} belongs to the interval $(0,1)$, the summary of all the u_{ik} belonging to the same point is one ($\sum_{i=1}^c u_{ik} = 1$). In addition, none of the fuzzy clusters is empty, neither do any contain all the data

$0 < \sum_{k=1}^m u_{ik} < m, 1 \leq i \leq c$. Update the membership matrix and clustering centers by minimizing Equation (5) via the Lagrange multipliers method:

$$u_{ik} = \frac{1}{\sum_{j=1}^c \left(d_{ik} / d_{jk} \right)^{1/(m-1)}}, \quad (6)$$

$$v_i = \frac{\sum_{k=1}^n u_{ik}^m x_k}{\sum_{k=1}^n u_{ik}^m}. \quad (7)$$

In the traditional FCM algorithm, d_{ik} denotes the Euclidean distance between x_i and v_k , and d_{jk} denotes the Euclidean distance between x_j and v_k .

3. Problem Formulation and Proposed Method

Consider a dataset $X = \{x_1, x_2, \dots, x_N\}$ with N objects. Each object is represented by m observations, in the form of $Y = \{y_1, y_2, \dots, y_m\}$. The purpose of data clustering is to divide datasets into several similar classes based on similarity measure, so that objects in the same cluster have great similarity and are easy to be analyzed. Multimedia data cluster tasks bring many problems and challenges, especially for missing or damaged data. Key challenges are discussed in three areas as below.

1. Learning the features of incomplete data: feature extraction and analysis are the basic steps of clustering. In general, many feature extraction methods, such as machine learning and deep learning, have been successfully applied to image, text, and audio feature learning. However, the current algorithm focuses on feature learning and extraction of high quality data. In other words, they can not effectively extract the features of lossy data. Therefore, feature learning of incomplete data is the primary problem of heterogeneous data clustering.
2. Clustering in feature space: an important feature of large-scale multimedia data is its diversity, which means that large-scale data sources are diverse, including structured, unstructured data and semi-structured data from a large number of sources. In particular, a large number of objects in large data sets are multi-model. For example, web pages usually contain both images and text. Each mode of multimodal object has its own characteristics, which leads to the complexity of data. Therefore, the feature representation of multimedia data is significant in cluster tasks.
3. Filling missing values to reconstruct data: in wireless multimedia sensor networks, reliable data transmission is critical to provide the ideal quality of network-based services. However, multimedia data transmission may not be successful due to different reasons such as sensory errors, connection errors, or external attacks. These problems can result in incomplete data and degrade the performance of WMSNS applications. After feature extraction and cluster analysis, it is very important to recover missing data from the sensor network.

3.1. Description of the Proposed Method

The variational autoencoder based high-order fuzzy c-means (VAE-HOFCM) algorithm is divided into three stages: unsupervised feature learning, high-order feature clustering, and data reconstruction. Architecture of the proposed method is shown in Figure 1.

To learn the features of incomplete multimedia data, the original data set is divided into two different subsets X_c and X_{inc} . Samples in subset X_c have no missing values while each sample contains some missing values in subset X_{inc} .

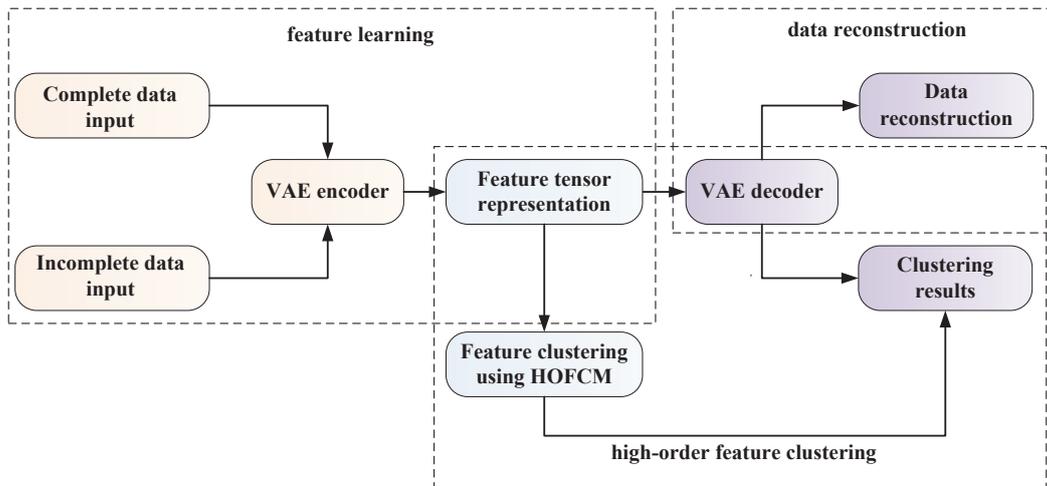


Figure 1. Architecture of the proposed method.

3.2. Feature Learning Network Architecture

For trained variational autoencoder, $q_\phi(z|x)$ will be very close to $p_\theta(z|x)$, so the encode network can reduce the dimensionality of the real dataset $X = \{x_1, x_2, \dots, x_N\}$ and obtain low-dimensional distribution. In this case, the potential variables may get better results than the traditional dimensionality reduction methods. When the improved VAE model is obtained, the encode network is used to learn the potential feature vectors of missing value sample $z = Encoder(x) \sim q_\phi(z|x)$. The decode network is then used to decode the vector z to generate the original sample $\hat{x} = Decoder(z) \sim p_\theta(x|z)$.

According to the original VAE and to build a better generation model, convolution kernels are added to the encoder. There is a variational constraint on the latent variable z , that is, z obeys the Gauss distribution. Here, each $x_i (1 \leq i \leq N)$ is fitted with an exclusive normal distribution. Sample z is then extracted from the exclusive distribution, since z_i is sampled from the exclusive x_i distribution, the original sample x_i can be generated through a decoder network. The improved VAE model is shown in Figure 2.

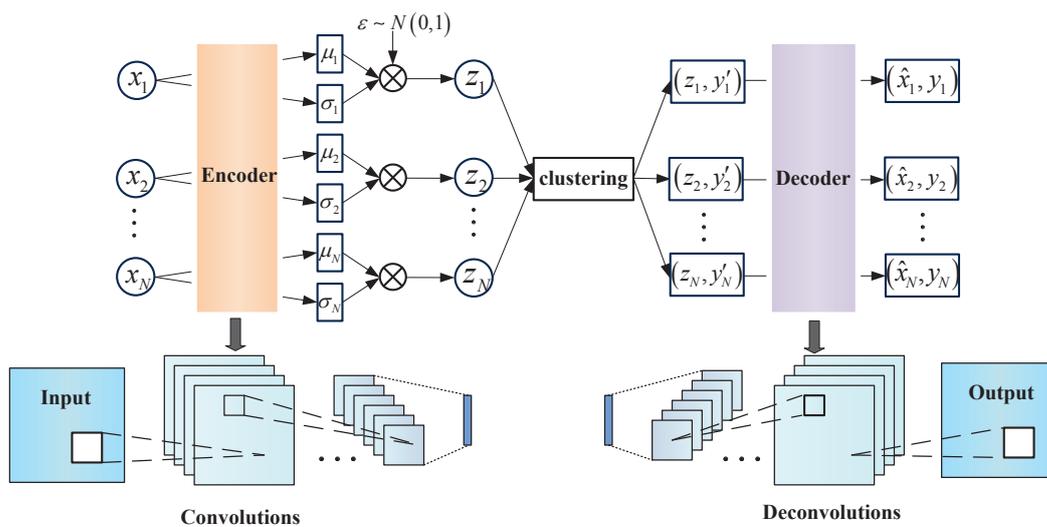


Figure 2. The improved VAE model.

In general, assume that $q_\phi(z)$ is the standard normal distribution, $q_\phi(z|x)$, $p_\theta(x|z)$ are the conditional normal distribution, and then plug in the calculation to get the normal loss of VAE, where z is a continuous variable representing the coding vector, and y is a discrete variable that represents a category. If z is directly replaced in the formula with (z, y) , the loss of the clustered VAE is obtained:

$$D_{KL}(q_\phi(z, y|x) \| p_\theta(z, y|x)) = \int_{-\infty}^{\infty} q_\phi(z, y|x) \log \frac{q_\phi(z, y|x)}{p_\theta(z, y|x)} dz. \quad (8)$$

Set the scheme as: $q_\phi(z, y|x) = q_\phi(y|z) q_\phi(z|x)$, $p_\theta(x|z, y) = p_\theta(x|z)$, $p_\theta(z, y) = p_\theta(z|y) p_\theta(y)$. Substituting them into Equation (8) and it can be simplified as follows:

$$E_{q_\phi(x)} \left[-\log p_\theta(x|z) + \sum_y q_\phi(y|z) D_{KL}(q_\phi(z|x) \| p_\theta(z|y)) + D_{KL}(q_\phi(y|z) \| p_\theta(y)) \right], \quad (9)$$

where the first term $-\log p_\theta(x|z)$ wants the reconstruction error to be as small as possible, that is, z keeps as much information as possible. $\sum_y q_\phi(y|z) D_{KL}(q_\phi(z|x) \| p_\theta(z|y))$ plays the role of clustering. In addition, $D_{KL}(q_\phi(y|z) \| p_\theta(y))$ makes the distribution of each class as balanced as possible; there will not be two nearly overlapping situations. The above equation describes the coding and generation process:

- Sampling to x from the original data, coding feature z can then be obtained by $q_\phi(z|x)$. Then, the coding feature is classified by classifier $q_\phi(y|z)$ to obtain the classification.
- Select a category y from distribution $p_\theta(y)$, select a random hidden variable z from distribution $p_\theta(z|y)$, and then decode the original sample through generator $p_\theta(x|z)$.

The VAE is outlined in Algorithm 1.

Algorithm 1 Variational Autoencoder Optimization.

Input: Training set $X = \{x_t\}_{t=1}^N$, corresponding labels $Y = \{y_t\}_{t=1}^N$, loss weight $\lambda_1, \lambda_2, \lambda_3$.

Output: VAE parameters θ, ϕ .

- 1: **Initialization:** random initialized θ_0, ϕ_0 .
 - 2: **Repeat:** Sample x_t in the minibatch.
 - 3: $\mu_{z_t} = \text{Encoder}(x_t) \sim q_\phi(z|x)$
 - 4: **Sample:** $z_t \leftarrow \mu_{z_t} + \varepsilon \odot \sigma_z, \varepsilon \sim \mathcal{N}(0, \mathbf{I})$
 - 5: $\mu_{x_t} = \text{Decoder}(z_t) \sim p_\theta(x|z)$
 - 6: Compute reconstruction loss: $L_{rec} = -\log p_\theta(x_t|z_t)$.
 - 7: Compute regularization loss: $L_{reg} = D_{KL}(q_\phi(y_t|z_t) \| p_\theta(y_t))$.
 - 8: Compute clustering loss: $L_{cls} = \sum_y q_\phi(y_t|z_t) D_{KL}(q_\phi(z_t|x_t) \| p_\theta(z_t|y_t))$.
 - 9: Fuse the three loss: $L(\theta, \phi) = \lambda_1 L_{rec}(\theta, \phi) + \lambda_2 L_{reg}(\theta, \phi) + \lambda_3 L_{cls}(\theta, \phi)$.
 - 10: Back-propagate the gradients.
 - 11: **Until** maximum iteration reached.
-

3.3. Variational Autoencoder Based High-Order Fuzzy C-Means Algorithm

Variational autoencoder gets the low-dimensional features and initial clustering results of data by feature learning. Then, the final clustering results will be optimized by the FCM algorithm clustering results. Traditional FCM work in vector space. It is better to use higher-order tensor to represent the feature of data because the tensor distance can capture the correlation in the high-order tensor space and measures the similarity between two higher-order complex data samples. Given an N -order tensor $X \in R^{I_1 \times I_2 \times \dots \times I_N}$, x is denoted as the vector form representation of X , and the element $X_{i_1 i_2 \dots i_N} (1 \leq i_j \leq I_j, 1 \leq j \leq N)$ in X is corresponding to x_l . That is, the N element in X is $l = i_1 + \sum_{j=2}^N \prod_{t=1}^{j-1} I_t$. Then, the tensor distance between two N -order tensors is defined as:

$$d_{td} = \sqrt{\sum_{l,m=1}^{I_1 \times I_2 \times \dots \times I_N} g_{lm} (x_l - y_l) (x_m - y_m)} = \sqrt{(x - y)^T G (x - y)}, \quad (10)$$

where g_{lm} is the metric coefficient and used to capture the correlations between different coordinates in the tensor space, which can be calculated by:

$$g_{lm} = \frac{1}{2\pi\delta^2} \exp \left\{ -\frac{\|p_l - p_m\|_2^2}{2\delta^2} \right\}, \quad (11)$$

where $\|p_l - p_m\|_2$ is defined as:

$$\|p_l - p_m\|_2 = \sqrt{(i_1 - i_1')^2 + \dots + (i_N - i_N')^2}. \quad (12)$$

Minimizing the objective function of high-order fuzzy c-means algorithm:

$$J_m(U, V) = \sum_{k=1}^n \sum_{i=1}^c (u_{ik}) d_{td}^2. \quad (13)$$

To update the membership value u_{ik} , we differentiate with respect to u_{ik} , as follows:

$$\begin{aligned} \frac{\partial J_m(U, V)}{\partial u_{ij}} &= \frac{\partial ((u_{ik})^m d_{td}^2(x_k, v_i))}{\partial u_{ij}} \\ &= m \cdot (u_{ij})^{m-1} d_{td}^2(x_j, v_i). \end{aligned} \quad (14)$$

Setting Equation (14) to 0, u_{ik} is calculated:

$$u_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d_{(td)ik}}{d_{(td)jk}} \right)^{1/(m-1)}}. \quad (15)$$

Then, the equation for updating v_i is obtained:

$$v_i = \frac{\sum_{j=1}^n u_{ij}^m x_j}{\sum_{j=1}^n u_{ij}^m}. \quad (16)$$

For each iteration, this operation requires $O(c \times n)$, so the total computational complexity of k iterations is $O(kc \times n)$. From the above, the VAE-HOFCM algorithm can be described as Algorithm 2:

Algorithm 2 The VAE-HOFCM algorithm.

Input: $X = \{x_1, x_2, \dots, x_n\}$

Output: $U = (u_{ij})$ and $V = (v_i)$.

- 1: Initialize $X = \{x_1, x_2, \dots, x_n\}$ randomly.
 - 2: Perform Algorithm 1 to calculate low dimensional representation of dataset X : $x = \text{Encoder}(x_n)$
 - 3: **for** $iteration = 1, 2, \dots, \text{max iter}$
 - 4: **for** $i = 1, 2, \dots, c$
 - 5: $v_i = \frac{\sum_{j=1}^n u_{ij}^m x_j}{\sum_{j=1}^n u_{ij}^m}$
 - 6: **for** $i = 1, 2, \dots, c$
 - 7: **for** $j = 1, 2, \dots, c$
 - 8: $u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{d_{(td)ik}}{d_{(td)jk}} \right)^{1/(m-1)}}$
 - 9: $(x, y) = \text{Decoder}(z_t)$.
 - 10: Obtain the modified clustering results using the u_{ij} .
-

By comparing the steps of the HOFCM algorithm, VAE-HOFCM can restore incomplete data simultaneously in the clustering process. Equally, the VAE-HOFCM algorithm has a total time complexity of $O(kc \times n)$. However, before that, it needs to train the variational autoencoder network.

4. Experiments

This section evaluates the performance of the proposed VAE-HOFCM algorithm on three representative datasets. To show the effectiveness of VAE-HOFCM, the unsupervised clustering accuracy (ACC) and adjusted rand index (ARI) for verification are adopted. ACC is calculated by:

$$ACC = \max_m \frac{\sum_{i=1}^n 1 \{l_i = m(c_i)\}}{n}, \quad (17)$$

where l_i and c_i indicate the ground-truth label and the cluster assignment produced by the algorithm, respectively. m ranges overall possible one-to-one mappings between clusters and labels. ARI is used to measure the agreement between two possibilistic partitions of a set of objects, where U denotes the true labels of the objects in datasets, and U' denotes a cluster generated by a specific algorithm. A higher value of $ARI(U, U')$ represents that the algorithm has more accurate clustering results.

To study the performance and generality of different algorithms, experiments are performed on three datasets:

- MNIST: The MNIST dataset consists of 70,000 hand-written digits of 28-by-28 pixel size. The digits are centered and the size is standardized.
- STL-10: A dataset consists of 96-by-96 color images. It contains 13,000 labeled images and 100,000 unlabeled images.
- NUS-WIDE: The NUS-WIDE dataset consists of 269,648 images and can be downloaded from [Flickr.com](https://www.flickr.com/photos/nus-wide/), a famous photo-sharing website.

4.1. Experimental Results on Complete Datasets

This section evaluates the performance of variational autoencoder based high-order fuzzy c-means algorithm (VAE-HOFCM) in clustering compared to other algorithms. The input dimensions of these three datasets are 784, 3072 and 500, respectively. The dimension of VAE hidden layer is set as 25, and the number of training iterations of the training set as 50. After obtaining the low-dimensional features, start clustering, and the membership factor is set as 2.5. Then, the required clustering center is calculated and the final normalized membership matrix U is returned to obtain the clustering result.

The clustering results are shown in Tables 1 and 2. Table 1 displays the optimal performance of unsupervised clustering accuracy of each algorithm. For MNIST data clustering class, the proposed VAE-HOFCM algorithm has achieved the highest accuracy of 85.54%. Compared with VAE clustering, the VAE-HOFCM encoder training time and cluster running time sum is slightly more than the former, but the clustering accuracy is improved. Then, the clustering performance and running time of VAE-HOFCM algorithm are generally better than traditional clustering algorithms, such as k-means and fuzzy c-means. Since the dimension of STL-10 dataset is higher and the information content is larger, the operation time of extracting features and clustering is relatively long. However, the proposed algorithm still gets the best running results. Visual features and text features are extracted from the NUS-WIDE dataset, and then these features are connected to form feature vectors. Finally, the feature vectors are clustered. The clustering results show the performance of the proposed algorithm.

Table 1. Clustering accuracy of ACC.

Algorithm/Dataset	MNIST	STL-10	NUS-WIDE
k-means	53.49%	28.40%	81.51%
HOPCM	80.34%	33.12%	92.75%
VAE	84.20%	35.48%	93.32%
DEC	84.31%	35.90%	93.75%
VAE-HOFCM	85.54%	36.44%	95.14%

Table 2 shows the clustering results in terms of $ARI(U, U')$, VAE-HOFCM produces high value than other algorithms in most cases. K-means usually has the worst performance and the longest running time, whereas VAE and DEC achieve the better result than HOPCM. ARI is not used as an indicator in the STL-10 dataset because the value may be negative in the case of clustering accuracy.

Table 2. Clustering accuracy of ARI.

Algorithm/Dataset	MNIST	STL-10	NUS-WIDE
k-means	0.41	-	0.74
HOPCM	0.69	-	0.89
VAE	0.75	-	0.90
DEC	0.76	-	0.90
VAE-HOFCM	0.78	-	0.92

There are two reasons for the results of these results in terms of ACC and ARI. On the one hand, HOFCM integrates the learning characteristics of different modes, uses the cross product to model the nonlinear correlation under various modes, and uses the tensor distance as a measure to capture the high-dimensional distribution of multimedia data. On the other hand, VAE successfully learns low-dimensional features and achieves the best performance in feature dimension reduction and clustering accuracy.

VAE has good data clustering and data generation performance. Feature extraction is carried out by the VAE to reduce the dimension to two dimensions. These categories have clear boundaries as shown in Figure 3, indicating that the VAE has effectively extracted low-dimensional features. This proves that the VAE has strong data feature expression ability.

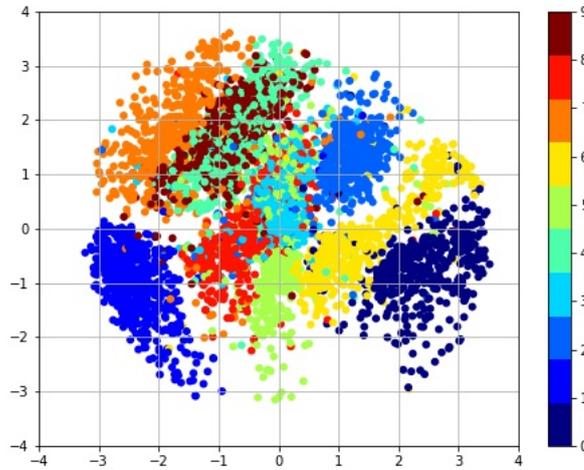


Figure 3. Visual analysis of MNIST datasets.

To obtain better performance in the three constraints of data feature dimension, clustering performance and reconstruction quality, the quality of data reconstruction in different dimensions is compared. Figure 4 shows the reproduction performance of learning generation models for different dimensions. When the latent space is set at 25, this method can obtain a good reconstruction quality.

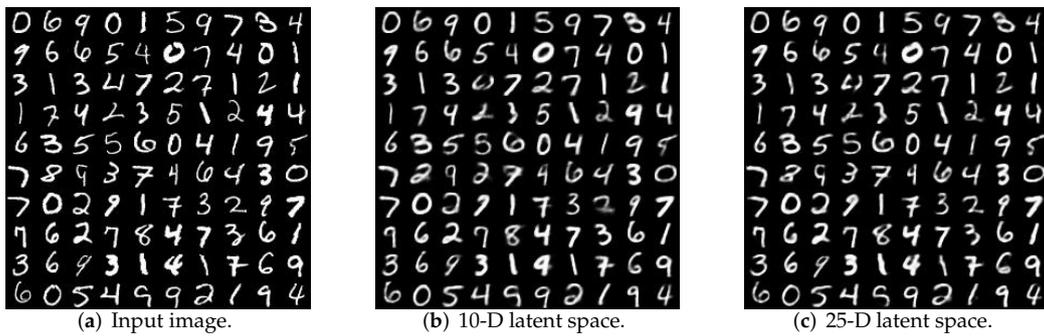


Figure 4. Reconstruction quality for different dimensionalities.

Figure 5 shows the generated images of two clustering results categories 1 and 6 of MNIST.

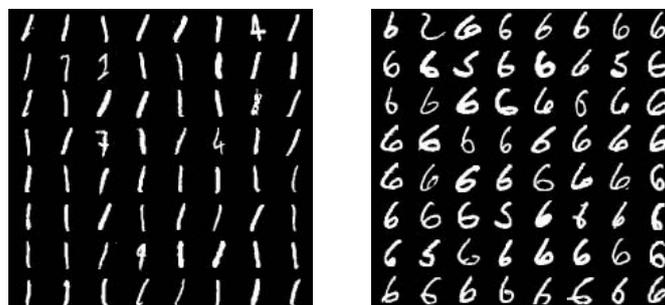


Figure 5. Cluster category sampling.

4.2. Experimental Results on Incomplete Data Sets

To estimate the robustness of the proposed algorithm, each dataset is divided into complete datasets and incomplete datasets. Now, incomplete datasets are used for simulation analysis. Since clustering performance depends on the number of missing values, six miss rates are set, which are 5%, 10%, 15%, 20%, 25% and 30%, respectively.

Figure 6 shows the clustering results accuracy of ACC with the increase of the missing ratio on the MNIST dataset and NUS-WIDE dataset. Figure 7 shows the average values of ARI with the increase of the missing ratio on the MNIST dataset and NUS-WIDE dataset. The results show that the increase of missing rate will lead to the decrease of clustering accuracy. However, the proposed algorithm still has a high accuracy because VAE successfully extracts incomplete data features and reduces the difference with the incomplete data features.

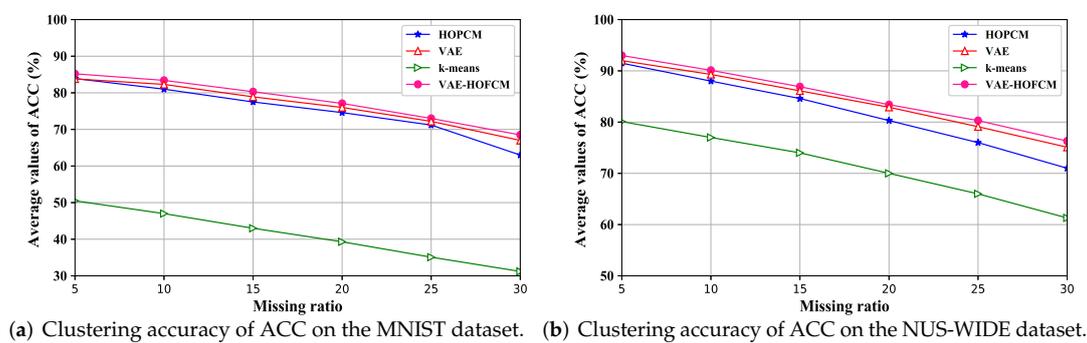


Figure 6. Clustering accuracy of ACC.

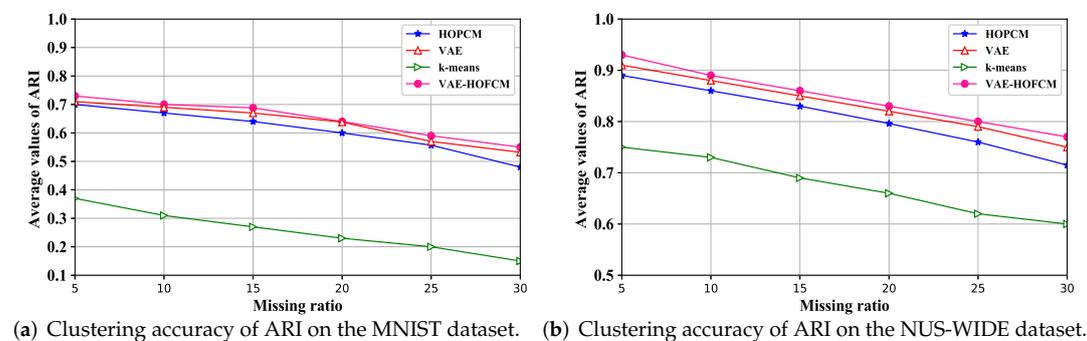


Figure 7. Clustering accuracy of ARI.

According to Figures 6 and 7, with the increase of missing rate, the average value of ACC and ARI would decrease, which indicates that the missing rate destroys the original data content, leading to the decrease of clustering accuracy. The average ACC and ARI values based on the VAE-HOFCM algorithm are significantly higher than those of the other three methods at the six missing rates. Therefore, VAE-HOFCM clustering has the best performance, indicating that VAE-HOFCM is also effective for clustering incomplete data.

Then, data with different missing rates are reconstructed, as shown in Figure 8. Inputs are incomplete data with different missing rates, and the output are recovered data using VAE. The reconstruction results show that the proposed algorithm not only improves the clustering accuracy, but also ensures that the data can be reconstructed with high quality.

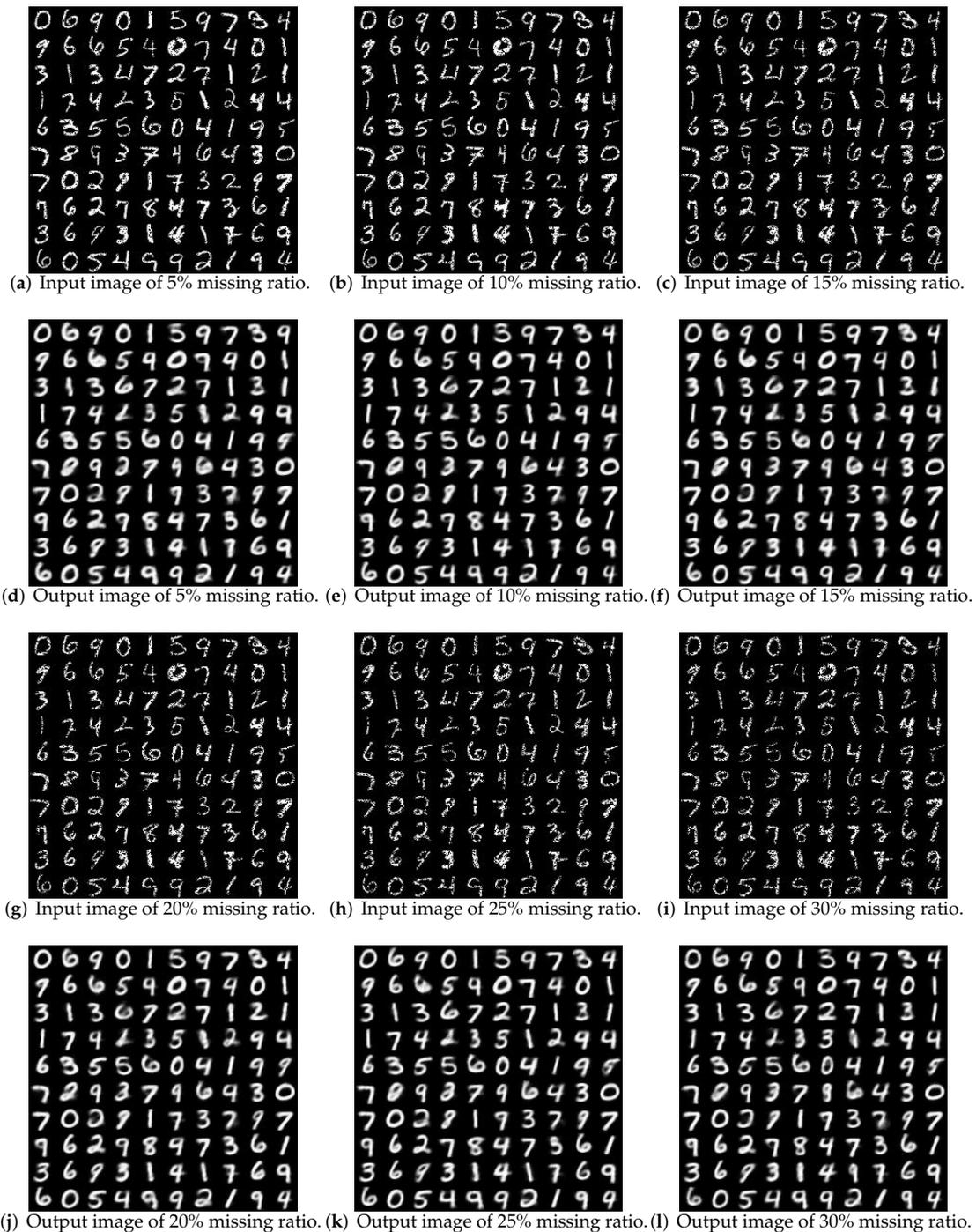


Figure 8. Reconstruction quality for different dimensionalities.

The variational auto-coder also has the function of de-noising. As shown in Figure 9, noise is added into the input data to enable VAE to effectively de-noise and restore the original input image.

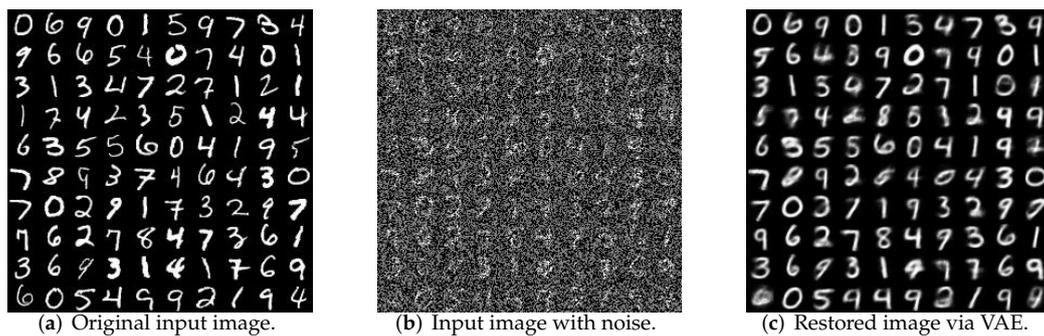


Figure 9. Reconstruction quality for noise data.

5. Conclusions

In this paper, a VAE-HOFCM algorithm, which can improve the performance of multimedia data clustering, has been proposed. Unlike many existing technologies, the VAE-HOFCM algorithm learns the data features by designing an improved VAE network, and uses a tensor based FCM algorithm to cluster the data features in the feature space. In addition, VAE-HOFCM captures as many features of high quality multimedia data and incomplete multimedia data as possible. In experiments, the performance of the proposed scheme has been evaluated on three heterogeneous datasets, MNIST, STL-10 and NUS-WIDE. Compared with traditional clustering algorithms, the results show that VAE can achieve a high compression rate of data samples, save memory space significantly without reducing clustering accuracy, and enable low-end devices in wireless multimedia sensor networks to achieve clustering of large data. In addition, VAE can effectively fill the missing data and generate the specified data at the terminal, so that the incomplete data can be better utilized and analyzed. Although VAE needs to be trained well, the sum time of training and clustering is still less than most clustering algorithms. Therefore, when performing clustering tasks on low-end equipment with limited computing power and memory space, trained VAE-HOFCM can be adopted.

Author Contributions: Conceptualization, X.Y. and Z.Z.; Data curation, C.G.; Formal analysis, X.Y. and H.L.; Funding acquisition, X.Y.; Investigation, H.L.; Supervision, Z.Z.; Validation, H.L. and C.G.; Visualization, Z.Z. and C.G.; Writing-original draft, X.Y. and H.L.; Writing-review and editing, X.Y., Z.Z. and C.G.

Funding: This work is supported by the Natural Science Foundation of China (Grant Nos. 61702066 and 11747125), the Chongqing Research Program of Basic Research and Frontier Technology (Grant No. cstc2017jcyjAX0256 and cstc2018jcyjAX0154), and the Research Innovation Program for Postgraduate of Chongqing (Grant Nos. CYS17217 and CYS18238)

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, Z.J.; Lai, C.F.; Chao, H.C. A green data transmission mechanism for wireless multimedia sensor networks using information fusion. *IEEE Wirel. Commun.* **2014**, *21*, 14–19. [[CrossRef](#)]
2. Wu, Y.; Guan, Y.; He, S.; Xin, M. An Industrial-Based Framework for Distributed Control of Heterogeneous Network Systems. *IEEE Trans. Cybern. Syst.* **2018**, *99*, 1–9. [[CrossRef](#)]
3. Wu, X.; Wang, H.; Liu, C.; Jia, Y. Cross-View Action Recognition Over Heterogeneous Feature Spaces. *IEEE Trans. Image Process.* **2015**, *24*, 4096–4108. [[PubMed](#)]
4. Shan, Z.; Xia, Y.; Hou, P.; He, J. Fusing Incomplete Multisensor Heterogeneous Data to Estimate Urban Traffic. *IEEE MultiMed.* **2016**, *23*, 56–63. [[CrossRef](#)]
5. Zhang, Z.; Zou, Y.; Gan, C. Textual sentiment analysis via three different attention convolutional neural networks and cross-modality consistent regression. *Neurocomputing* **2018**, *275*, 1407–1415. [[CrossRef](#)]
6. Mantri, D.S.; Prasad, N.R.; Prasad, R. Mobility and Heterogeneity Aware Cluster-Based Data Aggregation for Wireless Sensor Network. *Wirel. Pers. Commun.* **2016**, *86*, 975–993. [[CrossRef](#)]
7. Akbar, A.; Khan, A.; Carrez, F.; Moessner, K. Predictive Analytics for Complex IoT Data Streams. *IEEE Internet Things J.* **2017**, *4*, 1571–1582. [[CrossRef](#)]

8. Yim, H.J.; Seo, D.; Jung, H.; Back, M.K.; Kim, I.; Lee, K.C. Description and classification for facilitating interoperability of heterogeneous data/events/services in the Internet of Things. *Neurocomputing* **2017**, *256*, 13–22. [[CrossRef](#)]
9. Qiu, T.; Chen, N.; Li, K.; Atiquzzaman, M.; Zhao, W. How Can Heterogeneous Internet of Things Build Our Future: A Survey. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 2011–2027. [[CrossRef](#)]
10. Yang, J.; Han, Y.; Wang, Y.; Jiang, B.; Lv, Z.; Song, H. Optimization of real-time traffic network assignment based on IoT data using DBN and clustering model in smart city. *Future Gener. Comput. Syst.* **2017**. [[CrossRef](#)]
11. Zhang, Q.; Yang, L.T.; Chen, Z.; Xia, F. A High-Order Possibilistic C-Means Algorithm for Clustering Incomplete Multimedia Data. *IEEE Syst. J.* **2017**, *11*, 2160–2169. [[CrossRef](#)]
12. Han, Y.; Wang, Z.; Li, D.; Guo, Q.; Liu, G. Low-Complexity Iterative Detection Algorithm for Massive Data Communication in IIoT. *IEEE Access* **2018**, *6*, 11166–11172. [[CrossRef](#)]
13. Fekade, B.; Maksymyuk, T.; Kyryk, M.; Jo, M. Probabilistic Recovery of Incomplete Sensed Data in IoT. *IEEE Internet Things J.* **2018**, *5*, 2282–2292. [[CrossRef](#)]
14. Mendes, L.D.P.; Rodrigues, J.J.P.C.; Lloret, J.; Sendra, S. Cross-Layer Dynamic Admission Control for Cloud-Based Multimedia Sensor Networks. *IEEE Syst. J.* **2014**, *8*, 235–246. [[CrossRef](#)]
15. Zhao, L.; Chen, Z.; Yang, Z.; Hu, Y.; Obaidat, M.S. Local Similarity Imputation Based on Fast Clustering for Incomplete Data in Cyber-Physical Systems. *IEEE Syst. J.* **2018**, *12*, 1610–1620. [[CrossRef](#)]
16. Zhang, Z.; Zeng, T.; Yu, X.; Sun, S. Social-aware D2D Pairing for Cooperative Video Transmission Using Matching Theory. *Mob. Netw. Appl.* **2018**, *23*, 639–649. [[CrossRef](#)]
17. Li, T.; Zhang, L.; Lu, W.; Hou, H.; Liu, X.; Pedrycz, W.; Zhong, C. Interval kernel Fuzzy C-Means clustering of incomplete data. *Neurocomputing* **2017**, *237*, 316–331. [[CrossRef](#)]
18. Zhang, S.; Yang, Z.; Xing, X.; Gao, Y.; Xie, D.; Wong, H.S. Generalized Pair-Counting Similarity Measures for Clustering and Cluster Ensembles. *IEEE Access* **2017**, *5*, 16904–16918. [[CrossRef](#)]
19. Hoecker, M.; Polsterer, K.L.; Kugler, S.D.; Heuveline, V. Clustering of Complex Data-Sets Using Fractal Similarity Measures and Uncertainties. In Proceedings of the 2015 IEEE 18th International Conference on Computational Science and Engineering, Porto, Portugal, 21–23 October 2015; pp. 82–91.
20. Zhou, L.; Wu, D.; Zheng, B.; Guizani, M. Joint physical-application layer security for wireless multimedia delivery. *IEEE Commun. Mag.* **2014**, *52*, 66–72. [[CrossRef](#)]
21. Li, F.; Qiao, H.; Zhang, B. Discriminatively boosted image clustering with fully convolutional auto-encoders. *Pattern Recognit.* **2018**, *83*, 161–173. [[CrossRef](#)]
22. Gebru, I.D.; Alameda-Pineda, X.; Forbes, F.; Horaud, R. EM Algorithms for Weighted-Data Clustering with Application to Audio-Visual Scene Analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 2402–2415. [[CrossRef](#)]
23. Abualigah, L.M.; Khader, A.T.; Al-Betar, M.A. Unsupervised feature selection technique based on genetic algorithm for improving the Text Clustering. In Proceedings of the 2016 7th International Conference on Computer Science and Information Technology (CSIT), Amman, Jordan, 13–14 July 2016; pp. 1–6.
24. Saadaoui, F.; Bertrand, P.R.; Boudet, G.; Rouffiac, K.; Dutheil, F.; Chamoux, A. A Dimensionally Reduced Clustering Methodology for Heterogeneous Occupational Medicine Data Mining. *IEEE Trans. Nanobiosci.* **2015**, *14*, 707–715. [[CrossRef](#)] [[PubMed](#)]
25. Zhou, Q. Research on heterogeneous data integration model of group enterprise based on cluster computing. *Clust. Comput.* **2016**, *19*, 1275–1282. [[CrossRef](#)]
26. Ramachandran, N.; Perumal, V. Delay-aware heterogeneous cluster-based data acquisition in Internet of Things. *Comput. Electr. Eng.* **2018**, *65*, 44–58. [[CrossRef](#)]
27. Meng, L.; Tan, A.H.; Xu, D. Semi-Supervised Heterogeneous Fusion for Multimedia Data Co-Clustering. *IEEE Trans. Knowl. Data Eng.* **2014**, *26*, 2293–2306. [[CrossRef](#)]
28. Li, P.; Chen, Z.; Yang, L.T.; Zhao, L.; Zhang, Q. A privacy-preserving high-order neuro-fuzzy C-means algorithm with cloud computing. *Neurocomputing* **2017**, *256*, 82–89. [[CrossRef](#)]
29. Zhang, Q.; Yang, L.T.; Chen, Z.; Li, P. High-order possibilistic c-means algorithms based on tensor decompositions for big data in IoT. *Inf. Fusion* **2018**, *39*, 72–80. [[CrossRef](#)]
30. Mohammadi, M.; Al-Fuqaha, A.; Sorour, S.; Guizani, M. Deep Learning for IoT Big Data and Streaming Analytics: A Survey. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 2923–2960. [[CrossRef](#)]
31. Xie, J.; Girshick, R.; Farhadi, A. Unsupervised Deep Embedding for Clustering Analysis. *arXiv* **2015**, arXiv:1511.06335.

32. Kingma, D.P.; Welling, M. Auto-Encoding Variational Bayes. *arXiv* **2013**, arXiv:1312.6114.
33. Li, X.; Chen, Z.; Poon, L.K.M.; Zhang, N.L. Learning Latent Superstructures in Variational Autoencoders for Deep Multidimensional Clustering. *arXiv* **2018**, arXiv:1803.05206.
34. Hou, X.; Shen, L.; Sun, K.; Qiu, G. Deep Feature Consistent Variational Autoencoder. In Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, USA, 24–31 March 2017; pp. 1133–1141.
35. Lopez-Martin, M.; Carro, B.; Sanchez-Esguevillas, A.; Lloret, J. Conditional Variational Autoencoder for Prediction and Feature Recovery Applied to Intrusion Detection in IoT. *Sensors* **2017**, *17*, 1967. [[CrossRef](#)] [[PubMed](#)]
36. Celikyilmaz, A.; Trksen, I.B. *Modeling Uncertainty with Fuzzy Logic: With Recent Theory and Applications*; Springer Publishing Company: Berlin/Heidelberg, Germany, 2009.
37. Dovžan, D.; Škrjanc, I. Recursive fuzzy C-means clustering for recursive fuzzy identification of time-varying processes. *ISA Trans.* **2011**, *50*, 159–169. [[CrossRef](#)] [[PubMed](#)]
38. Jérôme, M.; Rui, A.; Francisco, S. Adaptive fuzzy identification and predictive control for industrial processes. *Expert Syst. Appl.* **2013**, *40*, 6964–6975.
39. Rastegar, S.; Araujo, R.; Mendes, J. Online Identification of Takagi-Sugeno Fuzzy Models Based on Self-Adaptive Hierarchical Particle Swarm Optimization Algorithm. *Appl. Math. Model.* **2017**, *45*, 606–620. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).