Review

A Review of Quantum-Inspired Metaheuristic Algorithms for Automatic Clustering

Alokananda Dey, Siddhartha Bhattacharyya, Sandip Dey, Debanjan Konar, Jan Platos, Vaclav Snasel, Leo Mrsic and Pankaj Pal

Special Issue

Hybrid Metaheuristic Algorithms for Portfolio Optimization and Its Applications

Edited by
Prof. Dr. Siddhartha Bhattacharya and Dr. Mohamed Abd Elaziz

Review

A Review of Quantum-Inspired Metaheuristic Algorithms for Automatic Clustering

Alokananda Dey [1,†] , **Siddhartha Bhattacharyya** [2,3,*,†] , **Sandip Dey** [4,†] , **Debanjan Konar** [5,†] , **Jan Platos** [6,†] , **Vaclav Snasel** [6,†] , **Leo Mrsic** [3,7,†] **and Pankaj Pal** [1,†]

[1] RCC Institute of Information Technology, Kolkata 700015, West Bengal, India; alokananda.dey@rcciit.org.in (A.D.); pankaj.pal@rcciit.org.in (P.P.)

[2] Rajnagar Mahavidyalaya, Rajnagar 731130, Birbhum, India

[3] Department of Data Analysis, Algebra University College, Catholic University of Croatia, 10000 Zagreb, Croatia; leo.mrsic@algebra.hr

[4] Sukanta Mahavidyalaya, Dhupguri 735210, Jalpaiguri, India; sandipdey@sukantamahavidyalaya.ac.in

[5] Center for Advanced Systems Understanding (CASUS), Helmholtz-Zentrum Dresden-Rossendorf (HZDR), 02826 Görlitz, Germany; d.konar@hzdr.de

[6] Faculty of Electrical Engineering and Computer Science, VŠB-Technical University of Ostrava, 70800 Poruba-Ostrava, Czech Republic; jan.platos@vsb.cz (J.P.); vaclav.snasel@vsb.cz (V.S.)

[7] Public Research Institute Rudolfovo Scientific and Technological Centre, 8000 Novo mesto, Slovenia

[*] Correspondence: dr.siddhartha.bhattacharyya@ieee.org; Tel.: +91-98-3035-4195

[†] These authors contributed equally to this work.

Abstract: In real-world scenarios, identifying the optimal number of clusters in a dataset is a difficult task due to insufficient knowledge. Therefore, the indispensability of sophisticated automatic clustering algorithms for this purpose has been contemplated by some researchers. Several automatic clustering algorithms assisted by quantum-inspired metaheuristics have been developed in recent years. However, the literature lacks definitive documentation of the state-of-the-art quantum-inspired metaheuristic algorithms for automatically clustering datasets. This article presents a brief overview of the automatic clustering process to establish the importance of making the clustering process automatic. The fundamental concepts of the quantum computing paradigm are also presented to highlight the utility of quantum-inspired algorithms. This article thoroughly analyses some algorithms employed to address the automatic clustering of various datasets. The reviewed algorithms were classified according to their main sources of inspiration. In addition, some representative works of each classification were chosen from the existing works. Thirty-six such prominent algorithms were further critically analysed based on their aims, used mechanisms, data specifications, merits and demerits. Comparative results based on the performance and optimal computational time are also presented to critically analyse the reviewed algorithms. As such, this article promises to provide a detailed analysis of the state-of-the-art quantum-inspired metaheuristic algorithms, while highlighting their merits and demerits.

Keywords: automatic clustering; metaheuristics; quantum computing; quantum-inspired metaheuristics

MSC: 81-02

1. Introduction

Data clustering is considered an unsupervised method for dividing unlabelled data into several groups according to some dissimilarity measures [1,2]. Clustering or cluster analysis is the process of segregating groups with similar objects based on some similarity measures. The segregated groups are called clusters, wherein the objects within a cluster resemble one another more than those within other clusters. When there is no external information about the objects, the clustering task is considered an instance of unsupervised learning [3,4]. Exploratory data analysis is performed in clustering to uncover the concealed

patterns in a dataset. Data clustering has various applications in various fields as it uses valuable hidden information within groups. The application areas of clustering include computer science, medical science, engineering, life science, Earth science, economics and bioinformatics, to name a few [3]. Examples of a few particular real-world uses of clustering are available in [5–10].

1. Data mining: Data mining is the practice of analysing data from different perspectives to extract valuable information from a huge amount of data to develop new products, reduce costs and improve decision-making tasks.

2. Marketing: The common uses of cluster analysis include market segmentation to identify the groups of several entities, viz., people, products, services and structure of an organisation, understanding customer behaviour, identifying the opportunities of new products and so on.

3. Community detection: The application area of community detection includes the analysis of social networks (e.g., LinkedIn®, Facebook®, Twitter®, etc.), politics (e.g., influence of political parties, astroturfing, etc.), public health (e.g., growth of epidemic spread, detection of cancer, tumours, etc.), smart advertising, criminology (e.g., criminal identification, fraud detection, criminal activity detection, etc.) and so on.

4. Insurance: Clustering is used to identify groups of insurance policyholders, assist insurers in taking the necessary measures for mitigating the impact of mortality deviations on the economy and help comprehend the company's experience with the emergence of death claims and fraud identification.

5. Image segmentation: Clustering is widely used in image segmentation for processing satellite images, medical images, real-life images, surveillance images, benchmark datasets, object identification, criminal investigations and security system in airports, to name a few.

The process of data clustering is divided into seven levels, viz., data collection, preliminary data screening, data representation, evaluating the clustering tendency, configuring the clustering strategy, data validation and finally, the interpretation of the clustering [11]. Clustering algorithms are chosen based on the availability of data and on the specific purpose and application. The major classification of clustering methods includes hierarchical algorithms, partitional algorithms, density-based algorithms, grid-based algorithms and model-based algorithms [12–16].

Hierarchical clustering [17–19] constructs a hierarchy or tree structure of clusters for data objects. The hierarchical clustering technique is divided into two types, viz., agglomerative and divisive [20,21] clustering. Agglomerative hierarchical clustering [20] is also known as the bottom–up approach, which initially considers each data point as a singleton cluster. After that, in each iteration, all similar clusters are merged to form new clusters until only a single cluster is left. On the other hand, divisive hierarchical clustering [21] is a top–down approach in which the clustering process begins with a single cluster and interactively produces smaller clusters by dividing them.

Partitional clustering initially constructs a set of disjoint clusters with *K* number of partitions by decomposing a dataset. After that, it uses an iterative relocation process that moves objects from one group to another to enhance the dataset's segmentation. The partitional clustering algorithms [18] are further classified into two basic types, viz., soft and hard clustering [10]. In soft clustering, the data objects are assigned to two or more clusters with a degree of membership value. Examples of these types of clustering methods include Fuzzy *C*-Means (FCM) [22], Fuzzy *C*-Shells (FCS) [23] and Mountain Method [24]. On the other hand, in hard clustering, the data objects are partitioned into disjoint clusters with respect to the objective function. The hard clustering method is classified into three categories, viz., *K*-means-based, histogram-based thresholding and metaheuristics-based clustering [10]. Examples of some prominent partitional clustering algorithms [18] include the *K*-means algorithm [25], *K*-mediods algorithm [26], CLARA [27] and CLARANS [28], to name a few.

Density-based clustering is an unsupervised learning method that identifies distinctive clusters by separating the data points from a contiguous region of high point density to low point density [18,29]. Some well-known density-based clustering methods include the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [30], Ordering Points To Identify the Clustering Structure (OPTICS) [31] and Mean-shift [32].

Grid-based clustering uses a multi-resolution grid data structure in which the object space is quantised into a finite number of cells. The quantised space is used for performing all the required clustering operations. The advantage of the grid-based method is its fast processing capability. Its processing time depends on the number of cells in each dimension in the quantised space. The grid-based methods include Clustering In QUEst (CLIQUE) [33], Statistical Information Grid (STING) [34], Merging of Adaptive Intervals Approach to Spatial Data Mining (MAFIA) [35] and Wave Cluster [36], to name a few.

Model-based clustering is a probabilistic approach. The mixture of various probability distributions generates the data in which each component represents different clusters. Model-based clustering includes two approaches: classification likelihood and mixture likelihood approaches. In the classification likelihood approach, the estimation of parameters is maximised. The sum of weighted component densities in the mixture likelihood approach represents the probability function. Examples of some model-based clustering algorithms include the expectation maximisation (EM) algorithm [37], Gaussian mixture density decomposition (GMDD) [38], COOLCAT [39] and STUCCO [40], to name a few.

Although the *K*-means algorithm [25] is considered the best performing algorithm among other clustering algorithms, it requires a priori information regarding the number of clusters in a dataset. Sometimes, knowing the exact number of clusters belonging to a dataset in advance becomes exceedingly difficult due to insufficient and improper information about it. Several researchers have stressed the need for automatic clustering to overcome this issue. The crucial issue of automatic clustering is to not only identify the appropriate number of clusters in a given dataset but it is also essential to evaluate the goodness of the outcome. It is possible to compare several findings and select the best satisfying one in the context of a given application. Thus, clustering can be considered an optimisation problem where the set of parameters that can be changed comprises the number of clusters, the similarity measure and other parameters in addition to the cluster memberships of all the input patterns.

In recent years, metaheuristic algorithms have been well suited for solving complex optimisation and engineering problems. Several nature-inspired metaheuristic algorithms for automatic clustering have evolved to date to mitigate the drawbacks of classical algorithms. Nature-inspired metaheuristic algorithms include evolutionary algorithms, swarm intelligence and stochastic and population-based algorithms. The examples of some metaheuristic algorithms for automatic clustering include Genetic Algorithm (GA) [41], Differential Evolution (DE) [42], Particle Swarm Optimisation (PSO) [43], Firefly Algorithm (FA) [44], Artificial Bee Colony (ABC) [45], Symbiotic Organism Search algorithm (SOS) [46], Bacterial Evolutionary Algorithm (BEA) [47], Grey Wolf Optimiser (GWO) [48], Bat Algorithm (BA) [49], Cuckoo Search (CS) [50] and Teaching Learning-Based Optimisation (TLBO) [51], to name a few. Other notable metaheuristic algorithms include the Starling Murmuration Optimiser (SMO) [52] and the Binary Starling Murmuration Optimiser (BSMO) [53].

The metaheuristic algorithms are well suited for solving optimisation problems but can still experience early convergence. Classical metaheuristic algorithms are hybridised by combining with other algorithms or incorporating quantum computing features to overcome this problem. The term quantum-inspired metaheuristic algorithm [54–56] has been introduced by combining the fundamental ideas of quantum computing with classical metaheuristic algorithms. Quantum-inspired metaheuristic algorithms were invented to run algorithms on nonquantum machines. They incorporate the principles of quantum mechanical phenomena with nature-inspired metaheuristic algorithms to achieve supremacy

in the performance of the algorithms while running on classical machines. Several quantum-inspired metaheuristic algorithms have been proposed in different fields [54].

This study puts forward an exhaustive review of the state-of-the-art metaheuristic algorithms for automatic data clustering starting from the classical metaheuristics to the quantum-inspired metaheuristics. The relevant papers for this study were collected from the DBLP-Citationnetwork V12 dataset. The DBLP-Citationnetwork V12 dataset [57] contains papers from various domains such as computer science, mathematics and economics. The citation data are collected from various sources, viz., the DBLP, ACM and Microsoft Academic Graph (MAG). Among 6,593,446 publications from the aforementioned dataset, information from 951 research papers on automatic clustering algorithms and 2818 publications for quantum algorithms were collected. Then, the inverted indexed abstracts were used to select the relevant topics with higher topic coherence by adopting the procedure presented in [58]. The publications collected from [57] on automatic clustering algorithms during the period 1970–2023 are presented in Figure 1. The total number of citations of the published papers are computed from the DBLP-Citationnetwork V12 dataset [57] and are presented in Figure 2. During this study, it has been seen that publications on quantum algorithms have significant growth. The incremental growth in publications in quantum algorithms over time is depicted in Figure 3. Thus, it is evident from Figures 1–3 that the field of quantum computing is flourishing at a tremendous rate along with the publications regarding automatic clustering, thereby establishing the importance of quantum algorithms in this direction.
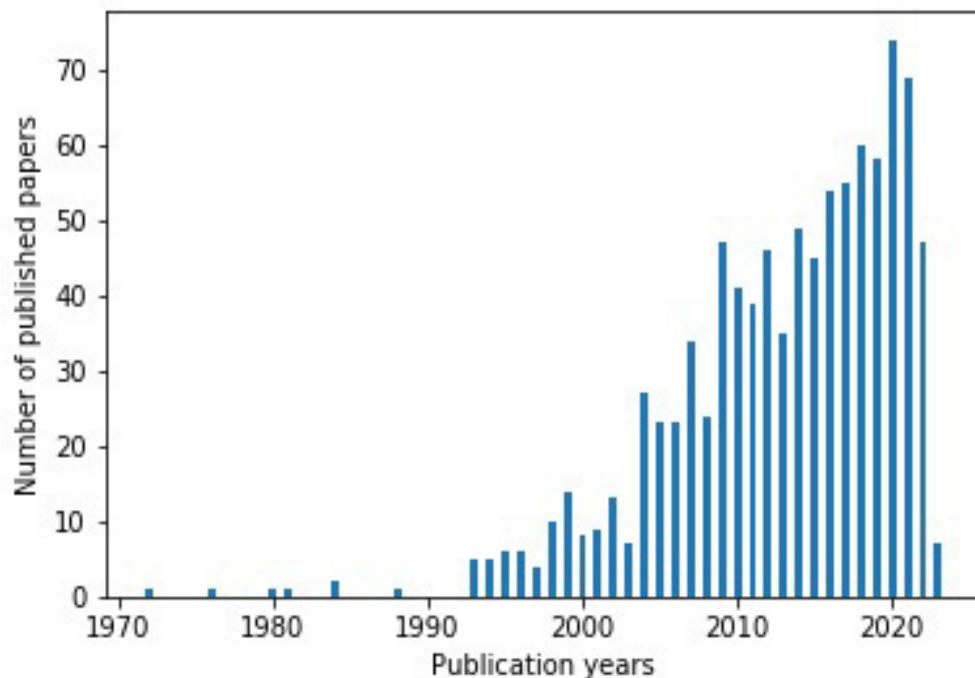
Figure 1. Publications on automatic clustering algorithms (1970–2022).
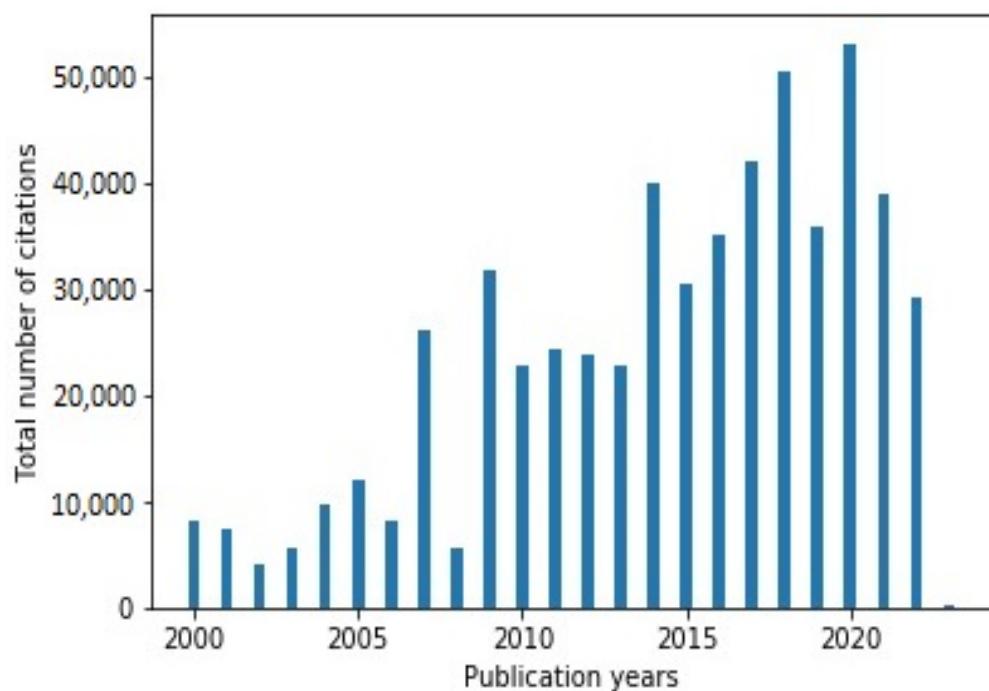
Figure 2. Total number of citations related to automatic clustering algorithms (2000–2023).
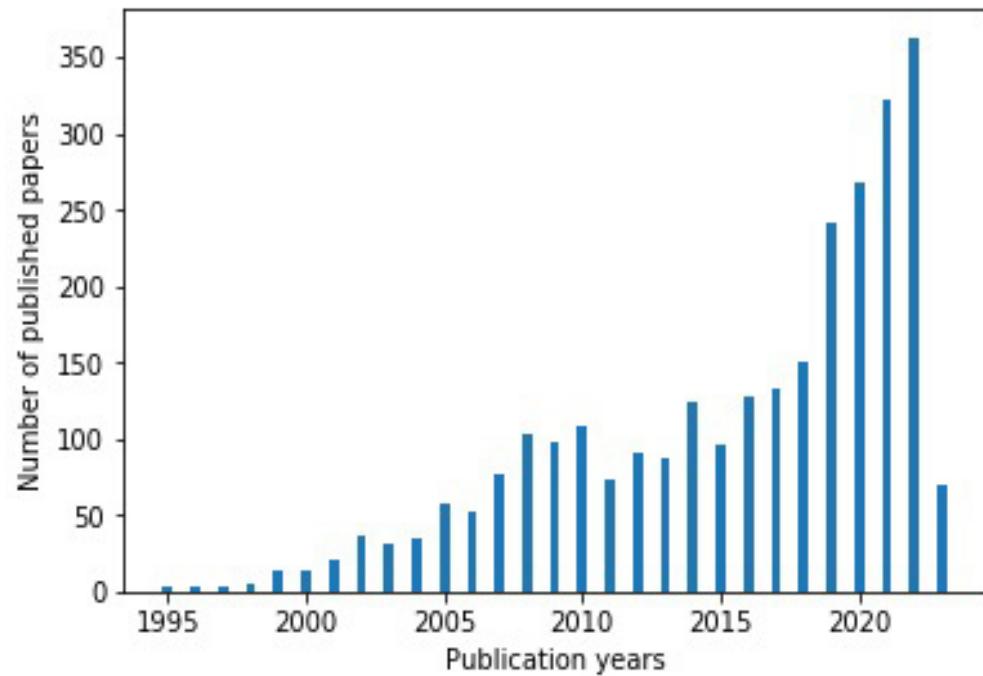
Figure 3. Publications related to quantum algorithms over time (1995–2023).

The wide application areas of clustering include pattern recognition, information retrieval, bioinformatics, optimisation, data mining, web analysis, machine learning, image segmentation and many more [3,59–63]. Among these major areas of clustering, the contribution of this work is enlightened within the domain of the automatic clustering of various types of datasets in single- and multi-objective environments. In this study, the challenges of the automatic clustering problem are demonstrated in nearly chronological order by employing classical approaches, metaheuristic approaches and quantum-inspired approaches. Figure 4 presents a graphical representation of these different clustering approaches for better visualisation. This study is organised to offer a qualitative insight into the pros and

cons of the mentioned algorithms for better comprehension and understanding. This article can be considered a unique avenue that can assist experienced researchers to easily access a wide range of meaningful works at the same place and judge the amount of work that has already been performed in this area. This article might serve as a starting point for aspiring researchers who want to explore the domain of quantum computing framework for the automatic clustering of various types of datasets. It would also serve as a reference for fellow researchers who wish to continue working on the issues commonly encountered in the problem of automatic clustering.

The purpose of this study was to present a thorough analysis of various algorithms for addressing the automatic clustering of various types of datasets. Moreover, comparative results based on the performance and optimal computational time are presented to substantiate the relative merits/limitations of the reviewed algorithms.

The rest of this paper is structured as follows. Section 2 presents a detailed description of the fundamentals of quantum computing. Automatic clustering is defined and described in Section 3. A discussion about a few major cluster validity indices is presented in Section 4. A brief discussion about the automatic clustering techniques based on classical, metaheuristic and quantum-inspired metaheuristic methods is provided in Sections 5–7. Finally, the study is concluded in Section 8. The different abbreviations used throughout the paper are presented in the abbreviations part.
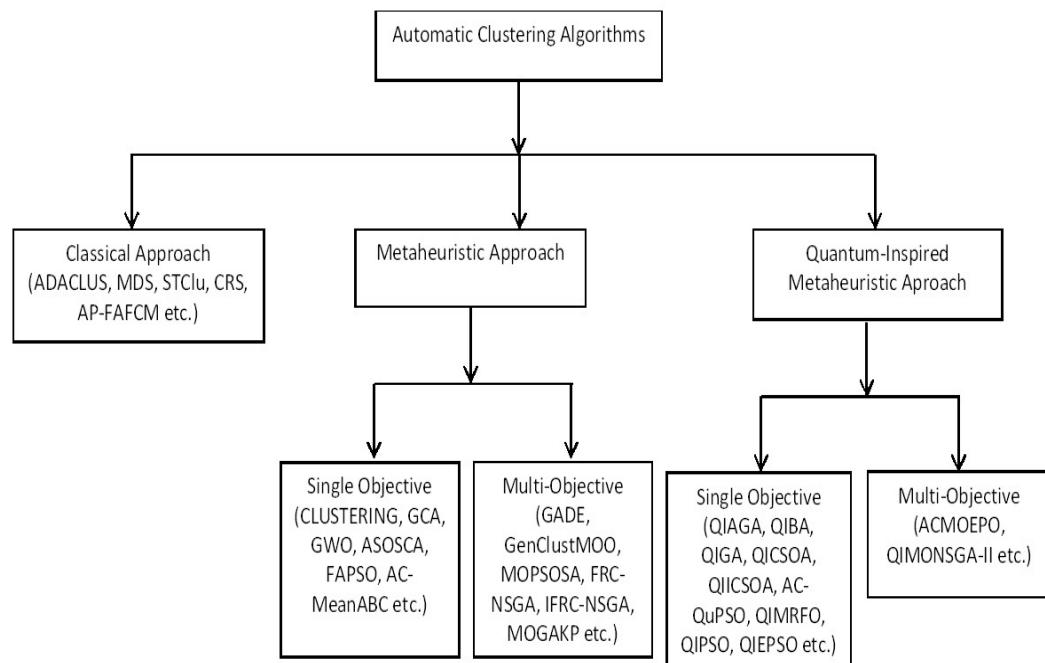
Figure 4. Classification of the automatic clustering algorithms.

2. Quantum Computing Fundamentals

A quantum computer emulates quantum mechanical principles to perform tasks that are difficult to complete using the laws of classical physics. As such, these machines can more effectively model complex real-life processes [64]. Quantum algorithms outperform all potential classical algorithms regarding speed or other efficiency improvements when executed on a quantum computer. Quantum algorithms can be classified into two basic types: pure quantum algorithms and quantum-inspired algorithms. Pure quantum algorithms can run on a quantum computer. The website Quantum Algorithm Zoo offers an extensive collection of quantum algorithms [65]. Quantum algorithms are useful for solving various problems such as searching and optimisation, cryptography, simulation of quantum systems and large systems of linear equations. Quantum machines can describe the structure and physical properties of atoms and subatomic particles and their effects

on the intersection. In the 1970s, a few physicists and computer scientists introduced the concept of quantum computing to computing devices [66]. Computing operations in these devices are reversible in nature in contrast to classical computers where electrical energy is dissipated in the form of heat energy during bit manipulation.

Paul Benioff's idea for creating a computer using the principles of quantum physics was initially conceived in 1981 [67,68]. After that, in 1982, Sir Richard Feynman first demonstrated that fundamental computations could be performed on a quantum system [69–71]. The first universal quantum Turing machine (QTM) was proposed by David Deutsch in 1985 [72], which led to the development of Deutsch–Josza's oracle [72] and Simon's oracle [73] in 1992 and 1994, respectively. Shor's factorisation technique, proposed in 1994, is considered a turning point in the advancement of quantum computing [74]. Then, in 1996, Grover proposed a fast database search algorithm referred to as Grover's algorithm [75].

In contrast to the conventional computer, a quantum computer employs quantum bits or *qubits* as a memory unit [76]. The manipulation of information in a quantum computer is performed by a quantum bit or *qubit*, which uses the two-state quantum-mechanical phenomena of superposition, entanglement and interference [77]. In quantum mechanics, the superposition state, $|\Psi\rangle$, is basically a combination of two basis states, $|0\rangle$ and $|1\rangle$. The information stored in the state $|0\rangle$, $|1\rangle$ or $|\Psi\rangle$ is used to realise a single *qubit*. Mathematically, the superposition state $|\Psi\rangle$ is represented as follows.

$$|\Psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \tag{1}$$

The two complex numbers α and β represent the probability amplitudes corresponding to the states $|0\rangle$ and $|1\rangle$, respectively. The values of $|\alpha|^2$ and $|\beta|^2$ determine which of these two states ($|0\rangle$ or $|1\rangle$) will collapse when the superposition state is destroyed out of measurement. The realisation of the superposition state can be modelled as follows.

$$|\Psi\rangle = \begin{cases} |0\rangle, & \text{if } |\alpha|^2 > |\beta|^2 \\ |1\rangle, \end{cases} \tag{2}$$

Otherwise, the Dirac notation corresponding to states $|0\rangle$, $|1\rangle$ and $|\Psi\rangle$ are $|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, $|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ and $|\Psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$, respectively, where $|\alpha|^2 + |\beta|^2 = 1$.

Several representation strategies attest to the superiority of a quantum bit over a classical bit. A quantum bit can concurrently store the values $|0\rangle$ and $|1\rangle$ in a single register, whereas a classical bit can only store 0 or 1 at a time. A classical computer can produce one of the four numbers 00, 01, 10 or 11 by combining two bits. Hence, it requires four registers to store the information. Contrarily, in a quantum computer, two quantum bits can simultaneously process and store four states of information, viz., $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$, using just two registers. Similarly, for an *n qubit* configuration, n is the number of quantum registers required to simultaneously store 2^n number of states. All 2^n states can be linearly superposed into a single sate $|\Psi\rangle$ as

$$|\Psi\rangle = \sum_{a=1}^{2^n} P_a\,|S_a\rangle \tag{3}$$

where $P_a \in \mathbb{C}$ and $\sum_{a=1}^{2^n} |P_a|^2 = 1$.

For a single *qubit*, Equation (3) transforms into

$$|\Psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \tag{4}$$

where

$$|\alpha|^2 + |\beta|^2 = 1 \tag{5}$$

In quantum computing, the linear superposition of the basis states $|\Psi\rangle$ is referred to as coherence and is represented by Equation (3). At the time of observation, all existing superposition states are forcefully destroyed and collapse into a single state, also called decoherence.

Quantum entanglement [78] is a phenomenon that occurs when two or more particles become correlated in such a way that their states cannot be separately characterised from one another, even when a large distance physically separates them. When two particles are entangled, measuring the state of one particle will instantaneously determine the state of the other particle, regardless of how far apart they are. Entanglement is a strange and fascinating property of quantum mechanics and it has become a subject of research.

In quantum computing, a *qutrit* [79,80] is a three-level quantum state consisting of three basis states, viz., $|0\rangle$, $|1\rangle$ and $|2\rangle$. In such a case, the superposition state can be represented as

$$|\Psi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle + \alpha_2 |2\rangle \tag{6}$$

where $\forall \alpha_i \in \mathbb{C}$, s.t. $i = 0, 1, 2$ and the normalisation constraint is $\alpha_0^2 + \alpha_1^2 + \alpha_2^2$. *Qutrits* have potential applications in quantum computing and communication, where they can encode more information per quantum state than a *qubit*. In particular, they can be used to improve the security of quantum key distribution protocols and enhance the capacity of quantum channels for transmitting information. However, the manipulation and control of *qutrits* are more challenging than those of *qubits* due to the higher dimensionality of their state space. Therefore, the study of *qutrits* is an emerging research topic in the realm of quantum information, and progress is being made towards their practical use in quantum technologies.

Qudits [81–83] are quantum systems with a state space with more than two dimensions. The term *qudit* is a generalisation of the term *qubit* to systems with higher dimensional state spaces. *A qudit* provides a larger state space to store and process information, which can reduce the complexity of the circuit, simplify the experimental setup and improve the algorithm's performance. A vector in the *n*-dimensional Hilbert space can be used to describe the state of a *qudit* [84], which is a quantum version of *n*-ary digits that can be represented by Equation (3). *Qudits* have potential applications in quantum information processing, including quantum computing and quantum communication, where they can be used to increase the information encoding capacity of quantum systems. Nowadays, in quantum information, the study of *qudits* is an active research area, and steps are being taken to make these systems useful in quantum technology.

A quantum computer employs a variety of quantum logic gates [85] to develop quantum algorithms. In essence, these gates are a collection of hardware components. The names of a few popular quantum gates include the Hadamard gate, NOT gate or Pauli-X gate, Pauli-Y gate, Pauli-Z gate, C-NOT gate, CCNOT gate or Toffoli gate, controlled phase-shift gate, Fredkin gate and Rotation gate [86,87].

Quantum evolutionary algorithms (QEAs) form a class of evolutionary algorithms which can run on both a classical computer and a quantum simulator [88] with minor changes in the code. In 2004, Yang et al. [89] proposed the QEA, which utilises the concept of quantum chromosomes, quantum mutation and quantum crossover. The QEA combines quantum theory with an evolutionary algorithm that runs on a classical computer with the flavour of quantum computing. In 2021, Acampora and Vitiello [88] proposed a new algorithm called HQGA, which uses a hybrid classical/quantum architecture to design a genetic algorithm. HQGA can run on a quantum processor and a simulator in the IBM Q Experience initiative. In order to experience the speed up in the classical computer, new versions of quantum algorithms have evolved, referred to as the quantum-inspired metaheuristic algorithms, which only run on classical computers. Quantum-inspired metaheuristic algorithms are further divided into three types, viz., single-objective-based [90], multi-objective-based [91] and many-objective-based [92] algorithms. In single-objective-based approaches, a single objective function is considered to obtain a single solution. Some examples of this type of approach include the Quantum-Inspired Genetic

Algorithm (QIGA) [90] proposed by Narayanan and Moore in 1996, the Quantum-Based Avian Navigation Optimiser Algorithm (QANA) [93] proposed by Zamani et al. in 2021 and the Binary Quantum-Based Avian Navigation Optimiser Algorithm (BQANA) [94] proposed by Nadimi-Shahraki et al. in 2022, to name a few. In multi-objective-based approaches, more than one objective function is considered to obtain a set of Pareto optimal solutions. An example of this type of approach is the Multi-Objective Quantum-Inspired Genetic Algorithm (Mo-QIGA), which was proposed by Konar et al. in 2018 [91]. The many-objective-based approach is a multi-objective approach with a large number of objectives. This approach considers more than four objectives to obtain a set of Pareto optimal solutions. In 2022, Balicki [92] proposed the Many-Objective Quantum-Inspired Particle Swarm Optimisation Algorithm (MQPSO) to ensure the diversity in a particle population by introducing quantum gates in PSO.

3. Automatic Clustering

As the name suggests, automatic clustering refers to the process of clustering in which the optimal number of clusters needs to be automatically identified before the clustering process in those situations where no a priori knowledge of the exact number of clusters belonging to a dataset is available [95–97]. Examples of some typical application areas of automatic clustering include browsing through personal photographs from a collection of photographs [95], database query execution plans [98], categorising colour images from synthetic data and a real-world COREL dataset [99], image segmentation [96,100] and automatic clustering of real-life datasets [97], to name a few.

Automatic clustering algorithms are required for the applications mentioned above as, most of the time, it is very difficult to analyse or cluster datasets without having prior information about them. It is seen that, in most cases, the dataset contains insufficient information. In this scenario, automatic clustering is crucial in identifying the exact number of clusters from the dataset for further processing.

Let us consider that the initial dataset, $D = \{x_1, x_2, x_3, ..., x_n\}$ contains n number of data points before clustering as depicted in Figure 5. The following conditions should be satisfied for a successful clustering of this dataset into K number of clusters.

$$C_i \neq 0, \; for \; i = 1, 2, ..., K \tag{7}$$

$$C_i \bigcap C_j = 0, \; for \; i, j = 1, 2, ..., K \, and \; i \neq j \tag{8}$$

$$\bigcup_{i=1}^{K} C_i = D \tag{9}$$

where C_i designates the individual clusters. However, if the number of clusters is unknown, an automatic clustering algorithm must group the data points in D into some groups or clusters. This grouping is performed based on some similarity measures. Hence, the main objective of an automatic clustering algorithm is to automatically identify a suitable number of clusters (k). The clustered dataset $C = \{C_1, C_2, C_3, ..., C_k\}$ containing a k number of clusters, thereby obtained, can be depicted as shown in Figure 6.

Cluster validity indices (CVI) play a crucial role in assessing the quality of clustering. However, there is no CVI that can guarantee accurate results for different data structures. In this regard, automatic clustering can be considered an optimisation problem to ensure more satisfactory results within a minimum time frame [101–103]. Considering it as an optimisation problem, researchers have designed algorithms which fall into three categories based on single-objective evolutionary approaches [101], based on multi-objective evolutionary approaches [103] and based on hybrid metaheuristic approaches [102]. Although the evolutionary algorithms provide almost ideal results within a minimum time frame, they nonetheless have the propensity to get stuck in local optima. A new research

direction has cropped up which entails quantum-inspired frameworks associated with classical metaheuristic approaches [104,105] to address this issue.
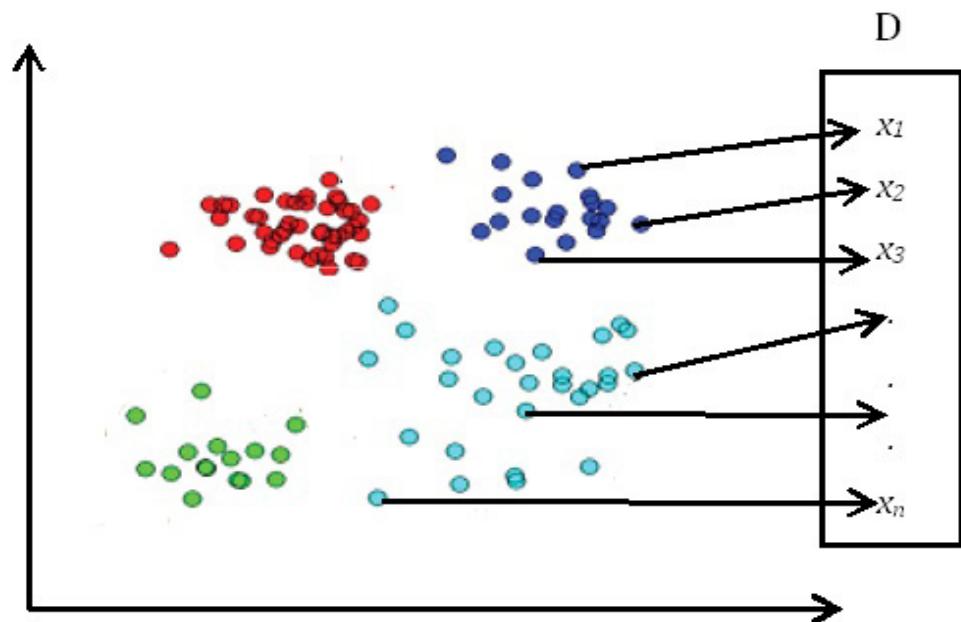
Figure 5. Initial dataset.
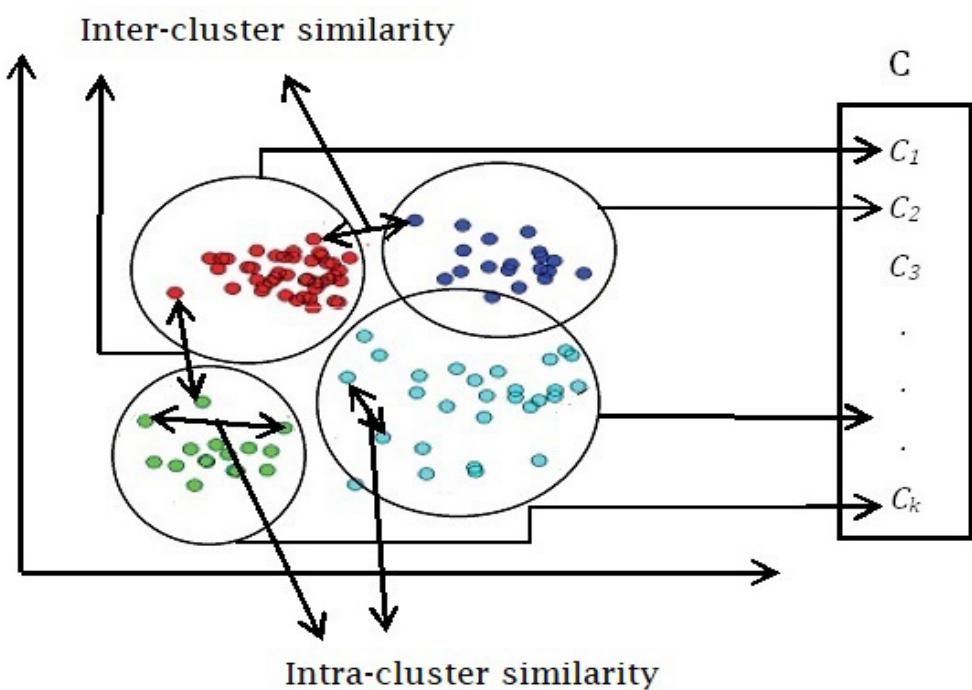
Figure 6. Clustered dataset.

4. Cluster Validity Indices

The cluster validity index (CVI) evaluates the goodness of a clustering algorithm by considering the information in the data themselves [106–115]. CVI is a mathematically justifiable function which can be either maximised or minimised. It defines a relationship between cluster cohesiveness and cluster separation to estimate the effectiveness of a

clustering solution. In metaheuristic algorithms, CVI is utilised as an objective function to be optimised. Some well-known cluster validity indices are discussed as follows.

4.1. Davies–Bouldin Index (DB)

The Davies–Bouldin index (DB) [116] was proposed by Davies and Bouldin in 1979. It has a positive correlation for the "within-class" scenario and a negative correlation for the "between-class" scenario. Mathematically, it is expressed as

$$DB = \frac{1}{N_c} \sum_{i=1}^{N_c} R_i \tag{10}$$

where R_i is computed as

$$R_i = \max_{i \neq j} R_{ij}, \tag{11}$$

$$R_{ij} = \frac{S_i + S_j}{D_{ij}}, \tag{12}$$

$$S_i = \left[\frac{1}{N_i} \sum_{l=1}^{N_i} \left\| X_l^{(i)} - Z_i \right\|^2 \right]^{\frac{1}{2}}, \tag{13}$$

and

$$D_{ij} = \left\| Z_i - Z_j \right\| \tag{14}$$

Here, S_i measures dispersal inside the cluster i, N_c represents the number of clusters, $X_l^{(i)}$ is a feature vector assigned to cluster i, Z_i is the centre of cluster i, N_i represents the cluster i and the Euclidean distance between the two centres of clusters i and j is represented by D_{ij}. The optimal result is achieved for a minimum value of the DB-index. A detailed explanation of the DB-index is available in [116].

4.2. Dunn Index (DI)

The Dunn index (DI) [117] was proposed by Dunn in 1973. It attempts to identify compact and well-separated sets of clusters. Mathematically, it is expressed as

$$DI = \min_{1 \leq i \leq N_c} \left(\min_{i+1 \leq j \leq N_c} \left(\frac{dist(C_i, C_j)}{\max_{1 \leq p \leq N_c} diam(C_p)} \right) \right) \tag{15}$$

where $dist(C_i, C_j)$ represents the distance from cluster C_i to cluster C_j and $diam(C_p)$ represents the diameter of cluster C_p.

A maximum value of DI indicates the optimal result. A detailed explanation of DI is available in [117].

4.3. Calinski–Harabasz Index (CH)

The Calinski–Harabasz index (CH) [118] was proposed by Caliński and Harabasz in 1974. In this index, the cohesiveness is calculated based on a measurement of the separation between the cluster points and their centroids. Mathematically, it is expressed as

$$CH = \frac{\sum_{i=1}^{k} n_i d_e^2(\mu_i, \mu) / (k-1)}{\sum_{i=1}^{k} \sum_{x \in C_i} d_e^2(x, \mu_i) / (n-k)} \tag{16}$$

where k represents the total number of clusters in a dataset with n data points, n_i and μ_i represent the number of points and the centroid of the i_{th} cluster, respectively, μ is the global centroid and $d_e^2(\mu_i, \mu) = \|\mu_i - \mu\|^2$. A maximum value of the CH index indicates the optimal result. A detailed explanation of the CH index is available in [118].

4.4. Silhouette Index (SI)

In 1987, Rousseeuw [119] proposed the Silhouette index (SI). This index compares the pairwise difference in distances within and between clusters to validate the clustering performance. Mathematically, it is expressed as

$$SI = \frac{1}{K} \sum_{i=1}^{K} S(C_i) \tag{17}$$

where K is the number of clusters and for a given cluster C_i, $S(C_i)$ is referred to as the Silhouette width of a point x and can be defined as

$$S(C_i) = \frac{1}{n_i} \sum_{x \in C_i} \frac{b(x) - a(x)}{max(a(x), b(x))} \tag{18}$$

where C_i consists of n_i number of patterns. The mean distance within a cluster and the shortest mean distance between two points are represented by $a(x)$, while $b(x)$ indicates the patterns in a different cluster. The average of the separations between x and the other patterns in the same cluster is used for this purpose. The value of $S(C_i)$ lies between -1 and 1. A value near 1 indicates that the sample is well clustered and assigned to a very appropriate cluster. Misclassifications can be identified for values closer to -1.

A maximum value of SI indicates the optimal result. A detailed explanation of SI is available in [117].

4.5. Xie–Beni Index (XB)

The Xie–Beni index (XB) [107] was proposed by Xie and Beni in 1991. It is an index for fuzzy clustering that also applies to crisp clustering. It is described as the ratio of the mean quadratic error to the smallest possible minimal squared distance between the points in the cluster. Mathematically, it is expressed as

$$XB = \frac{J}{D_{min}} \tag{19}$$

where

$$J = \frac{1}{N} \sum_{i=1}^{C} \sum_{j=1}^{N} u_{i,j}^m d^2(X_j, C_i) \tag{20}$$

$$d^2(X_j, C_i) = (X_j - C_i)^T A (X_j - C_i) \tag{21}$$

and

$$D_{min} = \min_{i,j} \left[d^2(C_i, C_j) \right] \tag{22}$$

where $u_{i,j}^m$ and C_i represent the fuzzy membership value and cluster centroid, respectively. The number of clusters and data points are represented by C and N, respectively. A represents a $p \times p$ positive definite matrix, p is the dimension of $X_j(j = 1, 2, ..., n)$ and $m > 1$ is the fuzzy index [120].

A minimum value of XB indicates the optimal result. A detailed explanation of XB is available in [107].

4.6. S_Dbw Index

The *S_Dbw* Index [121] was proposed by Halkidi and Vazirgiannis in 2001. It determines the compactness value of the clusters using the standard deviations of a set of objects and the standard deviations of a partition, where the distance between the centres of the

clusters determines the separation value. It is a ratio-type index in which the density of clusters is considered. Mathematically, it is expressed as

$$S\_Dbw = \frac{1}{k} \sum_{C_i \in C} \frac{\sigma(C_i)}{\sigma(X)} + \frac{1}{k(k-1)} \sum_{C_i \in C} \sum_{C_j \in C, C_i \neq C_j} \frac{Den(C_i, C_j)}{max\{Den(C_i), Den(C_j)\}} \tag{23}$$

where *Den* represents the density of the cluster defined as

$$Den(C_i) = \sum_{x_p \in C_i} f(x_p, \mu_i) \tag{24}$$

and

$$Den(C_i, C_j) = \sum_{x_p \in C_i \cup C_j} f\left(x_p, \frac{\mu_i + \mu_j}{2}\right) \tag{25}$$

where

$$f(x_p, \mu_i) = \begin{cases} 0, & d_e(x_p, \mu_i) > \sigma(C), \\ 1, & otherwise. \end{cases} \tag{26}$$

The lowest value of *S_Dbw* identifies the ideal outcome. A detailed explanation of *S_Dbw* is available in [121].

4.7. I Index

In 2002, Maulik et al. [122] proposed the *I* index. It is made up of three different components, viz., $\frac{1}{\mathcal{K}}$, $\frac{\mathcal{E}_1}{\mathcal{E}_\mathcal{K}}$ and $\mathcal{D}_\mathcal{K}$, and can be described as follows.

$$I = \left(\frac{1}{\mathcal{K}} \times \frac{\mathcal{E}_1}{\mathcal{E}_\mathcal{K}} \times \mathcal{D}_\mathcal{K}\right)^\rho \tag{27}$$

where $\mathcal{K}$ represents the number of clusters in the dataset, $\mathcal{DS}$. The power ρ controls the various configurations of clusters. A constant $\mathcal{E}_1$ represents the total distance of all data points from the centre of $\mathcal{DS}$. $\mathcal{E}_1$ is defined as

$$\mathcal{E}_1 = \sum_{\mathcal{P} \in \mathcal{DS}} \|\mathcal{P} - \mathcal{V}\| \tag{28}$$

where the centre of the patterns, $\mathcal{P} \in \mathcal{DS}$, is represented by $\mathcal{V}$. $\mathcal{E}_\mathcal{K}$ describes the total cluster scatters inside a pattern. It is defined as

$$\mathcal{E}_\mathcal{K} = \sum_{i=1}^{\mathcal{K}} \sum_{j=1}^{\mathcal{N}} \mathcal{U}_{ij} \|\mathcal{P}_j - \mathcal{V}_i\| \tag{29}$$

where the number of data points belonging to a dataset $\mathcal{DS}$ is represented by $\mathcal{N}$. The *i*th cluster centre is represented by $\mathcal{V}_i$ and a partition matrix, $\mathbf{U}(\mathcal{DS}) = [\mathcal{U}_{ij}]_{\mathcal{K} \times \mathcal{N}}$ is used to partition the data points. The cluster separation measure $\mathcal{D}_\mathcal{K}$ is defined as

$$\mathcal{D}_\mathcal{K} = \max_{i,j=1}^{\mathcal{K}} \|\mathcal{V}_i - \mathcal{V}_j\| \tag{30}$$

A maximum value of *I* indicates the optimal result. In reference [122], an elaborate discussion is presented on the individual contributions of $\frac{1}{\mathcal{K}}$, $\frac{\mathcal{E}_1}{\mathcal{E}_\mathcal{K}}$ and $\mathcal{D}_\mathcal{K}$.

4.8. CS-Measure (CSM)

The CS-Measure (*CSM*) [123] was proposed by Chou et al. in 2004. The ratio of the sum of the within-cluster scatter and between-cluster separation is used to generate this index. Mathematically, it is expressed as

$$CSM = \frac{\sum_{i=1}^{\mathcal{N}_e} \left[\frac{1}{\|\mathcal{N}_i\|} \sum_{\mathcal{D}_{\mathcal{P}i} \in \mathcal{D}_{\mathcal{S}\mathcal{E}\mathcal{T}i}} \max_{\mathcal{D}_{\mathcal{P}mx} \in \mathcal{D}_{\mathcal{S}\mathcal{E}\mathcal{T}i}} D_F(\mathcal{D}_{\mathcal{P}i}, \mathcal{D}_{\mathcal{P}mx})\right]}{\sum_{i=1}^{\mathcal{N}_e} \left[\min_{j \in \mathcal{N}_e, j \neq i} D_F(\mathcal{C}_{e i}, \mathcal{C}_{e j})\right]} \tag{31}$$

where $\mathcal{N}_i$ represents the total number of data points ($\mathcal{D}_{\mathcal{P}i}$) belonging to the *i*th cluster $\mathcal{D}_{\mathcal{S}\mathcal{E}\mathcal{T}i}$ and $D_F(\mathcal{D}_{\mathcal{P}i}, \mathcal{D}_{\mathcal{P}j})$ provides the distance between any two data points $\mathcal{D}_{\mathcal{P}i}$ and $\mathcal{D}_{\mathcal{P}j}$.

A minimum value of *CSM* indicates the optimal result. A detailed explanation of *CSM* is available in [123].

4.9. PBM Index (PBM)

In 2004, Pakhira et al. [124] proposed a CVI referred to as the *PBM* index (*PBM*). Mathematically, it is expressed as

$$PBM = \left(\frac{1}{N_C} \times \frac{E_0}{E_{N_C}} \times D_{N_C} \right)^2 \tag{32}$$

where N_C represents the number of clusters. A constant term, E_0, represents the summation of distances of all the data points in a pattern P_T from the centre of a dataset D_{SET}. E_0 can be measured as follows.

$$E_0 = \sum_{P_T \in D_{SET}} \|P_T - V_C\| \tag{33}$$

where V_C is the centre of the pattern $P_T \in D_{SET}$.

E_{N_C} defines the total cluster scatter belonging to P_T and is measured as

$$E_{N_C} = \sum_{i=1}^{N_C} \sum_{j=1}^{D_P} PM_{ij} \left\| P_{T_j} - V_{C_i} \right\| \tag{34}$$

where D_P denotes the number of data points in D_{SET} and V_{C_i} represents the *i*th cluster centre and $[PM_{ij}]_{N_C \times D_P}$ denotes the partition matrix of D_P. The term D_{N_C} represents the cluster separation measure and can be measured as follows.

$$D_{N_C} = \max_{i,j=1}^{N_C} \left\| V_{C_i} - V_{C_j} \right\| \tag{35}$$

PBM is generated after balancing the three factors, viz., $\frac{1}{N_C}$, $\frac{E_0}{E_{N_C}}$ and D_{N_C}.

A maximum value of *PBM* indicates the optimal result. A detailed explanation of *PBM* is available in [124].

4.10. Local Cores-Based Cluster Validity (LCCV) Index

The Local Cores-Based Cluster Validity (*LCCV*) Index [125] was proposed by Cheng et al. in 2019 to improve the performance of the *SI* index [119]. Arbitrary-shaped clusters can be effectively evaluated by this index. The performance of *SI* is enhanced by the *LCCV* index, which measures the dissimilarity between local cores using graph-based distance. Mathematically, it is expressed as

$$LCCV = \frac{1}{n} \sum_{i=1}^{n_l} (LCCV(i) \times n_i) \tag{36}$$

where n is the total number of objects in a dataset X. n_l represents the total number of local cores and n_i represents the total number of points in the ith local core and

$$LCCV(i) = \frac{b(i) - a(i)}{max\{b(i), a(i)\}} \tag{37}$$

where

$$a(i) = \left(\frac{1}{n_l(A)} - 1\right) \sum_{j \in A, j \neq i} D(i, j) \tag{38}$$

and

$$b(i) = min_{C \neq A}\{d(i, C)\} \tag{39}$$

where A represents the cluster in which a local core $i \in X$ is assigned and the total number of local cores in cluster A is represented by $n_l(A)$. $a(i)$ represents the average distance in a graph between the local core i and other local cores in A. $a(i)$ is computed by Equation (38). $d(i, C)$ represents the average distance in a graph between the local core i and the local cores in another cluster C, which is evaluated as follows.

$$d(i, C) = \left(\frac{1}{n_l(C)}\right) \sum_{j \in C} D(i, j) \tag{40}$$

$b(i)$ is selected as the smallest one among all $d(i, C)$ after considering all clusters C ($C \neq A$). The value of $LCCV(i)$ varies in the range of $(-1, 1)$. A better clustering result can be achieved for a maximum value of $LCCV$. A detailed explanation of the $LCCV$ index is available in [125].

5. Classical Approaches to Automatic Clustering

Classical approaches refer to those well-known methods that have been widely used and studied for a long time and have been evidenced to be reliable and effective solutions to the problem of automatic clustering. These algorithms iteratively improve the clustering solution by computing the means of current clusters and reassigning observations to the nearest cluster until the cluster means no longer changes or a maximum number of iterations is reached.

In 2004, Husain et al. proposed an efficient automatic clustering technique referred to as Similarity Index Fuzzy C-Means Clustering [126] to generate a more optimal Generalised Regression Neural Network (GRNN) [127,128] structure. This technique is used for dynamic system identification and modelling. In this work, the conventional fuzzy C-means clustering [22] algorithm and a similarity indexing technique were used to automatically cluster the relevant input data in a system. The proposed algorithm was compared with the existing clustering strategies, which rely on fuzzy C-means and a self-organising map. The outcome proved the supremacy of the proposed algorithm over others from the same domain.

An automatic clustering approach based on an adaptive influence function called ADACLUS was proposed by Nosovskiy et al. in 2008 for automatic clustering and boundary detection [129]. This algorithm can automatically identify the number of clusters from various two-dimensional datasets with complex shapes and diverse densities. The experimental results were found to be favourable towards the proposed algorithm.

In 2011, Li et al. [130] proposed an automatic classification method to classify the uncertain data by a soft classifier. The proposed classifier combines fuzzy C-means with a fuzzy distance function and an evaluation function to categorise the fuzzy objects in an ambiguous database. The experimental results indicated that the proposed classifier performs well in various types of databases that contain uncertain data. For experimental purposes, a location service environment was simulated by a synthetic dataset. The generation of the synthetic dataset involved a two-stage process. During the simulation process, a Gaussian random variable generator generated the deterministic dataset D having n

number of objects in the $[0, 1]^m$ unit space. After that, D was decomposed into two sets with $\frac{n}{2}$ number of objects. Experiments were conducted on two synthetic datasets (Synthetic I and Synthetic II) and the sensor database (Sensor). This paper compared the Automatic Fuzzy *C*-Means with Randomly Selected Initial Centres (AFCR) with Automatic Fuzzy *C*-Means with Selected Initial Centres by Histogram (AFCH). As per the experimental results, AFCH outperformed AFCR regarding running time.

In 2015, Zhang et al. [131] designed a clustering method referred to as Multi-Document Summarisation (MDS) to generate a summary from a given set of sentences by selecting appropriate sentences from the documents. The study described a multi-document summarising technique based on density peaks sentence clustering (DPSC) [132]. The proposed method can automatically find the best cluster centres and assign a rank to all the data points. In this paper, the experiments were conducted on the DUC2004 dataset [133]. The overall time complexity for the proposed method was found to be $O(K^2)$, where K represents the total number of sentences in the document. The proposed DPSC-based MDS method outperformed other comparable MDS methods, viz., DUC04Best [134], Centroid [135], ClusterHITS [136], SNMF [137], RTC [138], FGB [139] and the state-of-the-art unsupervised MDS methods such as LexRank [140] and CSFO [141], while providing similar results such as the WCS method [142].

In 2016, Wang and Song proposed an automatic clustering algorithm based on outward statistical testing on density metrics referred to as Statistical Test-Based Clustering (STClu)[143], which effectively clustered the objects by properly identifying the clustering centres. STClu provides a unique clustering approach to overcome the drawbacks of RLClu [132]. The robustness of STClu lies behind the local density evaluation metric, K-density $\widehat{\rho}$. It is reliable in detecting clustering centres compared to ρ in RLClu. The effectiveness of the proposed algorithm depends upon the number of nearest neighbours K. The distance of a cluster centre from its K nearest neighbours is generally less than the total distances between other non-clustering centres and their K neighbours. As a result, the K-density $\widehat{\rho}$ can more easily distinguish between several objects than the local density ρ in RLClu. The proposed algorithm performs three tasks, viz., extracting metrics, identifying cluster centres and clustering of objects. For experimental purposes, five groups of benchmark clustering datasets, viz., S-sets, A-sets, shape sets, high-dimensional datasets and real-world datasets were used [132,144]. The time complexity of STClu is $(O(n^2.O(dist)))$ which depends upon the distance (*dist*) between two objects. As a result, both STClu and RLClu exhibit the same time complexity. However, in most cases, STClu is more efficient than RLClu in identifying the clustering centres.

In 2018, Chen et al. proposed an automatic clustering algorithm based on region segmentation (CRS) [145]. Initially, it identifies an optimal number of clusters from a dataset. After that, depending on the data density, the clusters of the datasets evolve. In this work, all the experiments were conducted on six groups of synthetic datasets, viz., Can383, Jain, Aggregation, S1, S2 and S3 [146,147], and seven real-world datasets, viz., three UCI datasets, Geo-referenced Event Dataset of the Uppsala Conflict Data Program (UCDP) and three image datasets [148,149]. The overall time complexity of CRS is $O(n^2)$, where n is the number of data points in a given dataset. In this paper, the comparable algorithms included DBSCAN [30], IS-DBSCAN [146], DP [132], SCDOT [147] and STClu [143]. According to the experimental results, the proposed algorithm proved superior to other comparable algorithms and remained unaffected by the parameter settings.

In 2019, the authors proposed a fuzzy clustering algorithm referred to as AP-FA-FCM [150] for automatically determining the number of clusters in an image. This not only provided a solution for the automatic segmentation of images but also improved the quality of the image segmentation. Initially, this work determined the number of clusters using the Affinity Propagation (AP) clustering algorithm [151] by constructing a similarity matrix from the features of the extracted image. After that, the obtained number of clusters was used as an input to the fuzzy *C*-means (FCM) algorithm [22]. Then, the clustering centre was subsequently optimised using the FA [44]. The outcome of the experiments proved that the proposed algorithm has an excellent effect and can efficiently realise automatic image segmentation.

In 2020, Studiawan et al. proposed a graph-based automatic clustering technique referred to as Automatic Security Log Clustering (ASLoC) for security log clustering [152]. In this work, a graph data structure is used to represent the security logs, in which the logs are grouped by connecting their log entries for some similarity measures. This method used the *CH* [118] and *DB* indices [116] as the objective functions to validate the clustering results. For experimental purposes, five publicly available security log datasets, viz., SSH brute-force attacks records kept in authentication logs [153], event logs kept in the SecRepo (Security Repository) website provided by Sconzo [154], Snort IDS logs from The Honeynet Project by Chuvakin [155], Snort IDS log dataset produced by the National CyberWatch Center in the Mid-Atlantic Collegiate Cyber Defense Competition [156] and syslog from a real-world honeypot installed in a RedHat-based Linux server were used [157]. The experimental results proved that ASLoC could be efficiently used to group the security logs and provides better clustering than others.

In 2021, Sahoo and Parida proposed an automatic clustering technique for brain tumour extraction from MRI images [158]. This work uses the Communication with Local Agents (CLA) [159] clustering technique and morphological post-processing methods to extract the tumour regions from the white matter regions of the human brain. In order to identify the meningiomas, gliomas and pituitary tumours present in MRI images [158–160], this work uses the intensity clustering approach. The algorithm performed well in meningioma tumour detection compared to the other two tumour detection methods. It was also found to be capable of extricating tumours from the nearest location of the skull. The outcome proved the effectiveness of the suggested algorithm over other comparable algorithms.

A detailed study of the contributions made in the domain of classical approaches to automatic clustering is demonstrated in Tables 1–3. The listed algorithms dating between 2004 and 2021, are compared based on the aim of the concerned works, used mechanisms, data specifications, merits and demerits. All the listed algorithms significantly outperform their competing algorithms.

Table 1. Classical approaches to automatic clustering.

Ref.	Aim of the Work	Mechanism Used	Data Specifications	Merits and Demerits
[126]	This paper presents an efficient automatic clustering technique referred to as the Similarity Index Fuzzy *C*-Means Clustering to generate a more optimal GRNN (Husain et al., 2004).	Used techniques include the conventional fuzzy *C*-means clustering algorithm and a similarity indexing technique.	Two benchmark problems, viz., the gas furnace data of Box and Jenkins and the MackeyGlass model for producing WBC, served as a simulation for the proposed approach.	Merits: 1. It is suitable for online dynamic GRNN-based modelling. Demerits: 1. Only two dynamic time series are considered for simulating the work.
[129]	This paper presents an automatic clustering and boundary detection algorithm referred to as ADACLUS (Nosovskiy et al., 2008).	It is based on a local-adaptive influence function which is not predefined as in DBSCAN and DENCLUE.	It has been applied to various two-dimensional datasets with arbitrary shapes and densities. Two shape features (circular/non-circular and concave/non-concave) of the clusters in the datasets were considered.	Merits: 1. It is suitable for large-scale real-time applications. 2. It can efficiently identify the clusters of datasets with arbitrary shapes and non-uniform densities on the run. 3. It can also easily detect the boundary of the clusters. 4. It is more robust to noise resistance than other competitive algorithms. Demerits: 1. It is not designed for model-based clustering and hence cannot distinguish overlapping clusters.

Table 1. *Cont*.

Ref.	Aim of the Work	Mechanism Used	Data Specifications	Merits and Demerits
[130]	This paper presents an automatic soft classifier to classify uncertain data in synthetic datasets (Li et al., 2011).	The proposed classifier includes fuzzy *C*-means with a fuzzy distance function and an evaluation function.	It has been applied to two synthetic datasets (Synthetic I, Synthetic II) and the sensor database (sensor).	Merits: 1. The proposed classifier performed effectively on different types of data containing uncertain data. 2. The required running time was found to be 50% less than others. Demerits: 1. No significant difference was achieved in the rate of error.
[131]	This paper presents a clustering method referred to as MDS (Zhang et al., 2015).	It uses the DPSC technique to automatically produce a summary from a given set of sentences by selecting the appropriate sentences from those documents.	The DUC2004 dataset has been used to conduct the experiments.	Merits: 1. It verifies that DPSC can effectively handle MDS. Demerits: 1. Sentence similarity matrix is required to improve the query-based multi-document summarisation method preparation.

Table 2. Classical approaches to automatic clustering.

Ref.	Aim of the Work	Mechanism Used	Data Specifications	Merits and Demerits
[143]	This study introduces the automatic clustering algorithm, STClu, which is based on external statistical testing on density metrics (Wang et al., 2016).	It introduces a local density evaluation metric *K*-density $\hat{\rho}$ which offers more robustness than detecting the clustering centre ρ from RLClu.	It has been applied to five groups of benchmark clustering datasets.	Merits: 1. STClu is efficient in identifying the clustering centres in most cases. Demerits: 1. No significant improvement in terms of the time complexity $(O(n^2.O(dist)))$ is found in STClu when compared with RLClu.
[145]	This paper presents an automatic clustering algorithm referred to as CRS (Chen et al., 2018).	It uses a region segmentation mechanism, which is unaffected by the parameter settings, the shape of the cluster and the density of the data.	For experimental purposes, six groups of synthetic datasets and seven real-world datasets were used.	Merits: 1. It can efficiently and automatically identify the optimal number of clusters and the clusters of the dataset. Demerits: 1. Before the execution, it requires an approximate value of the number of nearest neighbour *K*.
[150]	This study introduces a fuzzy clustering algorithm referred to as AP-FAFCM for automatic clustering (Yangyang et al., 2019).	In this algorithm, the number of clusters is initially estimated using the AP clustering algorithm. After that, the FCM algorithm receives the obtained number of clusters, and then the FA is used to optimise the clustering centre.	It has been applied to three randomly selected images.	Merits: 1. It not only resolves the issue of automatic segmentation but also significantly raises the level of segmentation quality while maintaining the effect of segmentation. Demerits: 1. Very few datasets have been chosen. 2. The effect of segmentation has not been considered for all the datasets.

Table 3. Classical approaches to automatic clustering.

Ref.	Aim of the Work	Mechanism Used	Data Specifications	Merits and Demerits
[152]	This paper presents a graph-theoretic approach for ASLoC (Studiawan et al., 2020).	This method incorporates three steps of operation to maximise the number of percolations and intensity threshold for clique percolation, starting with a graph-theoretic approach, including intensity threshold and ultimately using a simulated annealing process.	Five publicly available security log datasets were used for experimental purposes.	Merits: 1. It automatically determines the used parameters without the intervention of user input. 2. It provides more significant results than comparable algorithms in all scenarios. Demerits: 1. The deployment of event log clustering is not yet achieved for anomaly detection. 2. A multi-objective framework still needs to be addressed.
[158]	This study describes an effective way to extract brain tumours by employing the CLA clustering technique (Sahoo et al., 2021).	All image slices are subjected to skull stripping using a morphological operation and a histogram-based methodology.	Three different types of tumours, viz., meningiomas, gliomas and pituitary tumours, have been taken into account from a publicly available brain tumour dataset.	Merits: 1. It shows 99.64% accuracy and outperforms other comparable algorithms in finding meningioma tumours near the skull regions compared to other types of tumours. Demerits: 1. Among three types of tumour, viz., meningiomas, gliomas and pituitary tumours, it only performs better for the meningioma tumours.

6. Metaheuristic Approaches to Automatic Clustering

Generally, classical algorithms do not always guarantee global optimality as they rely on the local search strategy. Thus, the results often depend on the initial starting point. In real-world scenarios, the manual recognition and categorisation of data points are often difficult as the datasets generally contain unlabelled data points. This problem limits the performance of the classical algorithms. Furthermore, classical algorithms are mostly seen to be problem-specific, and they also struggle to deal with discontinuity issues.

Nature-inspired metaheuristic algorithms have evolved to overcome the limitations of classical algorithms. The nature-inspired metaheuristic algorithms are classified into two broad categories: single-objective and multi-objective [161,162]. In the case of single-objective optimisation problems, a single objective function is used to compute the fitness of the individuals belonging to a solution or population. In real-world scenarios, a single objective function can properly tackle most problems. In other types of optimisation, the problem can be handled using multi-objective optimisation techniques in which more than one objective function is simultaneously optimised [163–166]. In this case, a set of Pareto optimal solutions is generated. The search gradually converges to the true Pareto front by identifying the solutions from the frontier of the Pareto efficient set. Metaheuristic algorithms have often been used to solve the problems of automatic clustering [167,168].

The following subsections present an extensive investigation of the most important nature-inspired metaheuristic algorithms used to deal with the problem of automatic clustering.

6.1. Single-Objective Approaches

In these approaches, a single candidate solution is used, and over time, it uses a local search strategy to improve this solution. However, the solution obtained by exe-

cuting a single-objective-based metaheuristic may get stuck in the local optima [161,169]. Some examples of single-objective-based metaheuristics include simulated annealing [170], Tabu search (TS) [171], microcanonical annealing (MA) [172] and guided local search (GLS) [173].

This section presents some of the current metaheuristic algorithms for solving the problem of automatic clustering of different types of datasets by satisfying a single objective function.

The earliest attempt for automatic clustering based on GA was proposed by Tseng and Yang in 2001 [97]. The proposed algorithm CLUSTERING with a heuristic strategy can automatically find the exact number of clusters. The proposed algorithm has been compared with three other conventional algorithms, viz., *K*-means, Single link and Complete link [25,174,175]. All the experiments were performed on one real-life and two artificial datasets [176], which proved that CLUSTERING outperformed its counterparts.

In 2003, Merwe and Engelbret proposed two new PSO-based approaches for data clustering [177]. In the first approach, PSO was used to find the centroids of a user-specified number of clusters. In the second approach, PSO was used to refine the clusters formed by the *K*-means algorithm [25] to seed the initial swarm. In this paper, the proposed algorithms were executed on six datasets. To prove the efficiency of the proposed algorithms, they were contrasted with the *K*-means clustering algorithm.

In 2004, Garai and Chaudhuri proposed a Genetically Based Clustering Algorithm (GCA) to automatically find the correct number of clusters using a two-stage split-and-merge strategy [178]. GCA is composed of two algorithms, viz., the cluster decomposition algorithm (CDA) and the hierarchical cluster merging algorithm (HCMA) [179,180]. One real-life and nine artificial datasets were used for experimental purposes. A comparison was made between the proposed GCA [178] and the CURE [181], DBScan [30] and Chameleon [182] clustering methods to prove the superiority of GCA [178].

Das et al. [183] proposed an improved differential evolution algorithm in 2008 to automatically cluster real-life datasets. The *CS*-measure [123] and *DB*-index [116] were used as the cluster validity indices. The proposed algorithm was compared with two other state-of-the-art automatic clustering techniques, viz., GA [184] and PSO [185]. The comparison was made based on the accuracy of the final clustering results, convergence speed and robustness. The analysis of the experimental results revealed that the proposed algorithm performed better than all other competitive algorithms.

In 2011, Karaboga and Ozturk proposed the ABC [186] algorithm for data clustering on benchmark datasets. This algorithm is inspired by the foraging behaviour of honey bees, which has been used to solve numerical optimisation problems [45]. The ABC [186] algorithm was compared with nine other clustering methods, including PSO [187]. The experimental results showed that ABC [186] was the best-performing algorithm over others.

In 2018, Kapoor et al. proposed an automatic clustering technique using the GWO for satellite image segmentation [48]. The main objective of the work was to demonstrate how the infrastructure and urbanisation around New Delhi are expanding while the amount of greenery is declining. The performance of this algorithm was assessed by the *DB*-index [116], inter-cluster distance and intra-cluster distance [48]. The proposed algorithm was compared with three other well-known algorithms, viz., GA [184], DE [183] and PSO [188]. The experimental results proved the efficiency of the proposed algorithm.

In 2018, Pacheco et al. [189] introduced an automatic clustering algorithm called Anthill which was motivated by the collaborative intelligent behaviour of ants. The proposed algorithm addressed the problem of an automatic grouping which is admittedly considered an NP-difficult problem. In order to quickly derive the solutions, it uses an adaptive strategy. The quality of the clustering and effectiveness of the algorithm was measured with the help of two different criteria: the *SI* index and visual inspection. The outcome of the experiments indicated that the proposed algorithm was more effective than other compared algorithms.

An automatic clustering algorithm known as ASOSCA was proposed by Elaziz et al. in 2019 [190]. Its foundation lies in a hybridised version of the Atom Search Optimisation (ASO) [191] technique along with the Sine Cosine Algorithm (SCA) [192]. ASOSCA used SCA as a local search operator to enhance the performance of ASO. The proposed algorithm used different cluster validity indices, viz., *DI* [117], *SI* [119], *DB* [116] and *CH* indices [118] to validate the goodness of clustering. In this work, sixteen clustering datasets were used for all experiments. The experimental results showed that the ASOSCA is superior to comparable hybrid metaheuristics.

A hybrid metaheuristic algorithm known as the Firefly Particle Swarm Optimisation (FAPSO) [44] algorithm for the automatic clustering of real-life datasets was proposed by Agbaje et al. in 2019. The merits of the FA [44] and PSO [43] algorithms were incorporated into FAPSO to improve its performance. In this paper, the proposed algorithm was compared with six different algorithms, viz., Automatic Clustering DE (ACDE) [183], Genetic Clustering With an unknown Number of Clusters *K* (GCUK) [184], Dynamic Clustering Particle Swarm Optimisation (DCPSO) [185], Classical DE [42], Classical FA [44] and Classical PSO [43]. The *CS*-measure [123] and the *DB*-index [116] were used as the cluster validity indices. As per the experimental results, it can be claimed that the FAPSO [44] algorithm performed better than all other participating algorithms.

The ABC [186] algorithm suffers from balancing exploration and exploitation. In 2021, Al-rosan et al. proposed the mean artificial bee colony (MeanABC) optimisation algorithm [193] referred to as AC-MeanABC for the purpose of automatic clustering. It was designed to enhance the performance of the ABC [186] algorithm by effectively exploring the search space. The algorithm also possesses the capability to explore the entire search space ranging from positive to negative directions to determine the exact number of clusters. The balance between exploration and exploitation was efficiently addressed. The effectiveness of this algorithm was evaluated by experimenting on eleven benchmark real-life datasets, viz., Iris, Ecoli, Wisconsin, Wine, Dermatology, Glass, Aggregation, R15, D31, Libras movement, and Wholesaler Customers, and a set of natural images from the Berkeley1 segmentation dataset, viz., the Lena, Jet plane, MorroBay, Mandril and Pepper images [194,195]. The outcome indicated that the proposed algorithm outperformed competitive algorithms in the same domain.

A detailed study of the contributions made in the field of single-objective metaheuristic approaches to automatic clustering is presented in Tables 4 and 5. The listed algorithms dating between 2001 and 2021 were compared based on the aim of the concerned works, used mechanisms, data specifications, merits and demerits. All the listed algorithms significantly outperform their competing algorithms.

Table 4. Single-objective metaheuristic approaches to automatic clustering.

Ref.	Aim of the Work	Mechanism Used	Data Specifications	Merits and Demerits
[97]	This study introduces the CLUSTERING algorithm to automatically identify the exact number of clusters and to also simultaneously classify the objects into these clusters (Tseng et al., 2001).	This work proposes a genetic clustering algorithm in which, initially, the single-linkage algorithm is employed to decrease the size of the large dataset. After that, a heuristic strategy is used to select the appropriate clustering.	Spectral feature vectors derived from the TIMIT database were used for this study.	Merits: 1. Almost all types of data can be effectively clustered by CLUSTERING. Demerits: 1. A good clustering result is only achieved for some specified settings of parameters.
[177]	This study presents two new PSO-based approaches for clustering data vectors (Merwe et al., 2003).	A standard gbest PSO and a hybrid approach were used in which the swarm individuals are seeded by the result of the *K*-means algorithm.	Two artificial classification problems and four publicly available datasets, viz., Iris Plants Database, Wine, Breast Cancer and Automotives were used for experimental purposes.	Merits: 1. It is successful with respect to faster convergence to reduce quantisation errors and provides higher inter-cluster distances and lower intra-cluster distances. Demerits: 1. For early convergence, it gets stuck in local optima.
[178]	This study demonstrates a GCA to automatically recognise the correct number of clusters using a two-stage split-and-merge strategy (Garai et al., 2004).	GCA uses two algorithms, viz., the CDA and the HCMA.	One real-life and nine artificial datasets were used for this study.	Merits: 1. It provides simple codes for implementation. 2. It shows quite encouraging results compared with other performing algorithms. Demerits: 1. Flexibility is decreased due to the fixed values of crossover and mutation operators.
[183]	This study presents an improved differential evolution algorithm for automatically clustering real-life datasets (Das et al., 2008).	An improved version of the DE algorithm was implemented in this study.	The experimental datasets include the Iris Plants Database, Glass, Wisconsin Breast Cancer Dataset, Wine and Vowel Dataset. This algorithm was also used to segment five 256×256 grey-scale images.	Merits: 1. It is easy to implement. 2. It proves to be superior to other competitive algorithms. Demerits: 1. It may not outperform DCPSO or GCUK for every dataset.
[48]	This paper presents an automatic clustering technique referred to as GWO for satellite image segmentation (Kapoor et al., 2018).	The proposed work is based on the Grey Wolf Optimisation algorithm.	The dataset comprises two satellite images of New Delhi.	Merits: 1. It has good convergence speed. 2. It can avoid local optima. Demerits: 1. It is unable to explore the entire search space.

Table 5. Single-objective metaheuristic approaches to automatic clustering.

Ref.	Aim of the Work	Mechanism Used	Data Specifications	Merits and Demerits
[189]	This paper provides an automatic clustering algorithm referred to as Anthill, which is influenced by the collaborative intelligent behaviour of ants (Pacheco et al., 2018).	It is based on the Ant Colony Optimisation (ACO) algorithm. In Anthill, the solution is represented by a set of solutions produced by the entire colony, formed from the partial digraphs and obtained by acquiring the highly connected components.	The experimental datasets include the Wine Dataset, Iris Dataset, Breast Cancer Wisconsin (Original) Dataset, Pima Indians Diabetes Dataset and Haberman's Survival Dataset from the UCI machine learning database.	Merits: 1. It uses an iterated racing parameter calibrator for automatically configuring the algorithm. Demerits: 1. It has low convergence speed.
[190]	This study presents an automatic clustering algorithm known as ASOSCA to identify the exact number of centroids along with their positions on the run (Elaziz et al., 2019).	It uses the hybridisation of the ASO and the SCA.	Sixteen clustering datasets were used for all the experiments.	Merits: 1. It can achieve global optima. Demerits: 1. It is not suitable for all types of datasets.
[44]	This study introduces a hybrid metaheuristic algorithm, FAPSO algorithm, for the automatic clustering of real-life datasets (Agbaje et al., 2019).	FAPSO incorporates the basic features of the FA algorithm and PSO algorithms.	Twelve benchmark datasets have been taken from the UCI machine learning data repository of the University of California for this study.	Merits: 1. It can reach the global optima. 2. FAPSO performs significantly better than other state-of-the-art clustering algorithms. Demerits: 1. Convergence speed is low.
[193]	This paper proposes an automatic clustering algorithm referred to as AC-MeanABC, which incorporates an improved exploration process of the ABC algorithm (Alrosan et al., 2021).	The unsupervised data clustering method, AC-MeanABC, uses the MeanABC capability of balancing between exploration and exploitation and its capacity to explore the positive and negative directions in search space to determine the optimal result.	Eleven benchmark real-life datasets and natural images from the Berkeley1 segmentation dataset were considered for the experiments.	Merits: 1. It can explore and exploit the search space in both positive and negative directions. Demerits: 1. It has a lower convergence speed.

6.2. Multi-Objective Approaches

In a multi-objective optimisation problem [163–166], the decision variable vector $\bar{x}^* = \left[x_1^*, x_2^*, ..., x_n^*\right]^T$ generally optimises M number of objectives simultaneously by satisfying m inequality and n equality constraints as follows.

$$g_i(\bar{x}) \geq 0, i = 1, 2, ..., m \tag{41}$$

and

$$h_i(\bar{x}) \geq 0, i = 1, 2, ..., n \tag{42}$$

where the optimisation vector is given by

$$f(\bar{x}) = [f_1(\bar{x}), f_2(\bar{x}), ..., f_M(\bar{x})]^T \tag{43}$$

while considering a maximisation problem, a solution $\bar{x}_j$ is said to be dominated by a solution $\bar{x}_i$ if the following criteria are met.

$$\forall k \in 1, 2, ..., M, f_k(\bar{x}_i) \geq f_k(\bar{x}_j) \tag{44}$$

and

$$\exists k \in 1, 2, ..., M, s.t. f_k(\bar{x}_i) > f_k(\bar{x}_j) \tag{45}$$

The solutions that are not dominated by any member of the set of solutions comprise the non-dominated set of solutions. This non-dominated set of solutions is referred to as the Pareto optimal front.

This section presents a few metaheuristic algorithms based on the multi-objective optimisation framework for automatic clustering.

In 2009, Suresh et al. [196] presented a comparison between four multi-objective variants of DE [197–202]—viz., the Multi-Objective DE (MODE) [197], the Pareto DE (PDE) [198,200], DE for Multi-objective Optimisation (DEMO) [199], and the Non-Dominated Sorting DE (NSDE)—with Non-Dominated Sorting Genetic Algorithm (NSGA-II) [201] and the Multi-Objective Clustering with an Unknown Number of Clusters K (MOCK) [202]. All the experiments were conducted on six artificial datasets (Dataset_1 to Dataset_6) and four real-life datasets, viz., Iris, Wine, Breast Cancer and the Yeast Sporulation data with a varying range of complexities [203,204]. After analysing the experimental results, it was found that MODE [196] happened to be the best promising candidate algorithm among others for devising a multi-objective clustering framework for the automatic clustering of artificial and real-life datasets.

In 2009, Kundu et al. [205] proposed the hybrid multi-objective optimisation algorithm, GADE, for solving the automatic fuzzy clustering problem. The proposed algorithm is a hybridisation of GA [201] and DE [197,206,207] algorithms. In this work, two conflicting fuzzy validity indices, viz., XB [107] and fuzzy C-means (FCM) measure (J_m) [208], were simultaneously optimised to solve the automatic fuzzy clustering problem. All the experiments were conducted on six artificial datasets (Dataset_1–Dataset_6) and four real-life datasets, viz., Iris, Wine, Breast Cancer and the Yeast Sporulation data [203,204]. The computational results indicated that GADE performed better than two other compared algorithms, viz., NSGA-II [201] and MOCK [202].

In 2013, Saha and Bandyopadhyay [209] proposed an automatic clustering algorithm called GenClustMOO for the automatic clustering of artificial and real-life datasets. GenClustMOO uses a simulated annealing-based multi-objective framework, AMOSA [165], to identify the optimal number of clusters and the appropriate partitioning from datasets with a variety of cluster structures. All the experiments were conducted on nineteen artificial datasets, viz., symmetrically shaped clusters (Sym_5_2, Sym_3_2, Ellip_2_2, Ring_3_2, and Rect_3_2) [210], hyperspherical shaped clusters (Sph_5_2, Sph_3_4, Sph_6_2, Sph_10_2, and Sph_9_2) [184], well-separated clusters of different shapes, sizes and convexities (Pat1, Pat2, Long1, Sizes5, Spiral, Square1, Square4, Twenty and Forty) [211] and seven real-life datasets, viz., Iris, Cancer, Newthyroid, Wine, LiverDisorder, LungCancer and Glass [203,204]. The proposed algorithm, GenClustMOO, was compared with the multi-objective clustering technique, MOCK [197], and the single-objective clustering technique, VGAPS [210]. GenClustMOO was found to be superior to other comparable algorithms. The superiority of GenClustMOO was also established by performing the statistical superiority tests, viz., Friedman test [212] and Nemenyi's test [213].

In 2015, Abubaker et al. [214] proposed an automatic clustering algorithm referred to as the Multi-Objective Particle Swarm Optimisation and Simulated Annealing (MOPSOSA). MOPSOSA is designed by combining the principles of two multi-objective-based algorithms, viz. the multi-objective-based particle swarm optimisation (MPSO) [215] and the multi-objective simulated annealing (MOSA) [216]. In MOPSOSA, three different cluster validity indices, viz., DB-index [116], Sym-index [210] and Conn-index [110] were simultaneously used to optimise the obtained results. Experiments were conducted on fourteen artificial and five real-life datasets [217]. The proposed algorithm successfully identified the number of clusters from various overlapping and non-convex datasets with different shapes. The experimental results proved its superiority in all respects while comparing it to other multi-objective and single-objective algorithms.

In 2018, Paul and Shill [218] proposed two multi-objective automatic clustering methods, viz., Fuzzy Relational Clustering with NSGA-II (FRC-NSGA) and Improved FRC

with NSGA-II (IFRC-NSGA). FRC-NSGA incorporates the features of FRC [219] and NSGA-II [166]. This method can perform fuzzy clustering without knowing the cluster numbers beforehand. In FRC-NSGA, NSGA-II was used to simultaneously optimise the cluster validity indices of separation and cohesion, while FRC was employed to deal with the overlapping characteristics of clusters. IFRC-NSGA provides an enhanced version of FRC-NSGA by reducing the randomness of the initial membership values of FRC-NSGA. In this paper, the proposed methods were compared with three other existing multi-objective clustering methods, viz., VAMOSA [166], MOCK [197] and FCM-NSGA [220], and with a single-objective clustering method VGAPS [221]. All the experiments were conducted on two types of datasets: gene expression and non-gene expression data. The gene expression data included four microarray datasets, viz., Prostate Tumor [222], Leukemia [221], Colon Cancer [223] and the DLBCL datasets [224]. The Non-Gene Expression data included synthetic datasets, viz., AD_5_2 [224], AD_10_2 [223], Square-1, Square-4 and Long-1 [197], and real-life datasets, viz., Iris, Glass, Wine and Liver Disorders [225]. In contrast to the previously existing methods, the proposed methods efficiently identified well-separated, hyperspherical, non-compact and overlapping clusters. For the Colon dataset, Prostate Tumor dataset, Leukemia dataset and Lymphoma dataset, IFRC-NSGA achieved 90.0%, 94.11%, 94.44% and 97.0% accuracy. In contrast, FRC-NSGA showed accuracies of 90.0%, 93.14%, 94.44% and 97.0%, respectively, for the same dataset. Based on the experimental findings, the supremacy of the IFRC-NSGA algorithm over others was established.

In 2019, Dutta et al. [226] proposed a multi-objective-based automatic clustering algorithm called MOGA-KP for automatically determining the exact number of clusters from some real-life benchmark datasets having numeric or categorical features. This work is based on the working principle of a multi-objective Genetic Algorithm (MOGA) [227] in which, at every generation of MOGA, a single iteration of K-prototypes [228] is performed to conduct a local search. The performance of the proposed work was compared with seven algorithms, viz., K-Prototypes (KPs) [228], fuzzy C-means (FCMs) [229], Mean Shift (MS) [230], Hierarchical Clustering [231], Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [30], Self-Organising Map (SOM) [232] and the Single-Objective Genetic Algorithm with K-Means (SGA-KP) [233]. In this work, the performance of the proposed algorithm was measured by experimenting on twenty-five benchmark datasets taken from the University of California at Irvine (UCI) machine learning repository [234]. The MOGA-based automatic data clustering algorithm MOGA-KP is regarded as the first to handle both continuous and categorical information. Two statistical tests, viz., the Friedman's test [212] and a non-parametric equivalent of the repeated-measures ANOVA test [235], proved the superiority of the proposed algorithm over others.

In 2021, Qu et al. [236,237] proposed an automatic clustering algorithm based on MOGA referred to as NSGAII-GR. NSGAII-GR incorporates the features of the well-known Non-Dominated Sorting Genetic Algorithm-II (NSGAII) [166] along with a gene rearrangement technique. Two internal cluster validity indices, viz., the sum of generalised sample variance and CH index [118], were used in this paper to simultaneously optimise the results. The well-known DB-index determines the best result from the Pareto optimal front [116]. Finally, the SI coefficient [119] was used to judge the optimal result. Experiments were conducted on five two-dimensional artificial datasets, viz., T9_1, Tb_1 [197], Square1, Square4 and Long1 [238], five real-world datasets, viz., Iris, Wine, NewThyroid, WBC and WDBC from the UCI machine learning dataset [225], and ten datasets 10d-4c-No.(0–9) having the characteristics of 10 dimensions and 4 clusters, and ten datasets 10d-10c-No.(0–9) having the characteristics of 10 dimensions and 10 clusters [239]. This work presents a comparison between the proposed algorithm and the Fuzzy C-Means Non-Dominated Sorting Genetic Algorithm II (FCM-NSGA-II) [220] and the MOCK [197]. The experimental results indicate the superiority of the multi-objective optimisation method over all other single-objective optimisation methods while using the same objective function. The remarkable advantage

offered by the algorithm is that the gene rearrangement processes and the inter-cluster merging do not increase the time complexity.

A detailed study of the contributions made in the field of multi-objective metaheuristic approaches to automatic clustering is presented in Tables 6 and 7. The listed algorithms dating between 2001 and 2021 are compared based on the aim of the concerned works, used mechanisms, data specifications, merits and demerits. All the listed algorithms significantly outperform their competing algorithms.

Table 6. Multi-objective metaheuristic approaches to automatic clustering.

Ref.	Aim of the Work	Mechanism Used	Data Specifications	Merits and Demerits
[196]	This study presents a comparison between four multi-objective variants of DE for the automatic clustering of artificial and real-life datasets (Suresh et al., 2009).	The comparable algorithms include the MODE, the PDE and DE for Multi-Objective Optimisation (DEMO) and the NSDE.	In this study, six artificial and four real-life datasets were considered.	Merits: 1. It can handle multi-objective optimisation problems. Demerits: 1. Time and space complexity are higher.
[205]	This paper presents the hybrid multi-objective optimisation algorithm, GADE, for solving the automatic fuzzy clustering problem (Kundu et al., 2009).	It is a hybridisation of GA and DE algorithms.	Six artificial and four real-life datasets have been used.	Merits: 1. It is based on the multi-objective framework. Demerits: 1. Time and space complexity are higher.
[209]	This work presents an automatic clustering algorithm referred to as GenClustMOO for the automatic clustering of artificial and real-life datasets (Saha et al., 2013).	GenClustMOO uses a simulated annealing-based multi-objective framework, AMOSA, to identify the optimal number of clusters and the appropriate partitioning from datasets with various cluster structures.	All the experiments were conducted on nineteen artificial and seven real-life datasets.	Merits: 1. It is based on a multi-objective framework and generates a set of Pareto optimal fronts. Demerits: 1. Convergence rate is lower.
[214]	This study introduces an automatic clustering algorithm referred to as MOPSOSA for the automatic identification of the exact number of clusters in a dataset (Abubaker et al., 2015).	It combines the features of MPSO and the MOSA.	During the experiments, with fourteen artificial and five real-life datasets being considered.	Merits: 1. It can solve the multi-objective optimisation problem. 2. MOPSOSA outperforms all its competitors. Demerits: 1. Suffers from a low convergence rate.
[218]	This paper presents two multi-objective automatic clustering algorithms, viz., FRC-NSGA and IFRC-NSGA (Paul et al., 2018).	The proposed FRC-NSGA algorithm incorporates the features of the well-known FRC and NSGA-II algorithms. The proposed algorithm IFRC-NSGA is designed to improve the performance of FRC-NSGA.	Several Gene Expression data and Non-Gene Expression data were considered to conduct the experiments.	Merits: 1. Multi-objective framework is used to handle real-life problems. Demerits: 1. Space requirement is higher.

Table 7. Multi-objective metaheuristic approaches to automatic clustering.

Ref.	Aim of the Work	Mechanism Used	Data Specifications	Merits and Demerits
[226]	This paper introduces a multi-objective-based automatic clustering algorithm referred to as MOGA-KP (Dutta et al., 2019).	This work combines the features of MOGA and *K*-Prototypes (KPs) to automatically identify the exact number of clusters from some real-life benchmark datasets with multiple numeric or categorical features.	Twenty-five benchmark datasets from the UCI machine learning repository were used for experimental purposes.	Merits: 1. It can provide a set of Pareto optimal solutions. 2. It can handle different features, viz., continuous and categorical and missing feature values. 3. It can also deal with the missing features. Demerits: 1. It does not give priority to the features in clustering.
[237]	This study demonstrates a multi-objective-based automatic clustering algorithm referred to as NSGAII-GR for the automatic clustering of different types of real-life and artificial datasets (Qu et al., 2021).	It is based on the principle of the well-known Non-Dominated Sorting Genetic Algorithm-II (NSGAII) with a gene rearrangement technique.	The experiments have been conducted on five two-dimensional artificial datasets, five real-world datasets and twenty ten-dimensional datasets with various clusters.	Merits: 1. The remarkable advantage offered by the algorithm is that it prevents the time complexity from increasing while conducting the gene rearrangement process and inter-cluster merging process. Demerits: 1. It cannot properly handle uneven overlapping datasets.

7. Quantum-Inspired Metaheuristic Approaches for Automatic Clustering

In 1995, Narayanan and Moore introduced a new concept following the basic principles of quantum mechanics to obtain more efficient evolutionary methods [240]. Quantum computing principles are incorporated in these quantum-inspired metaheuristics with the backbone of several metaheuristics to enhance the performance of the existing ones. The first two pioneer works based on the concepts and principles of quantum computing are the Genetic Quantum Algorithm (GQA) [241] and Quantum-Inspired Evolutionary Algorithm (QEA) [242]. These algorithms use the concept of quantum bits and the superposition states. After that, several quantum-inspired metaheuristic algorithms have been developed to date to address different categories of optimisation issues, viz., Quantum-Inspired Particle Swarm Optimisation (QIPSO) [243], Quantum-Inspired Firefly Algorithm with PSO [244] and Quantum-Inspired Acromyremx Evolutionary Algorithm [245], to name a few.

A brief illustration of some metaheuristic algorithms influenced by quantum mechanisms, which automatically cluster different types of datasets using single- and multi-objective frameworks, is presented in this section.

7.1. Single-Objective Approaches

This section presents a few single-objective quantum-inspired metaheuristic algorithms for addressing the problem of the automatic clustering of datasets.

In 2010, Ramdane et al. proposed the Quantum-Inspired Evolutionary Algorithm for Data Clustering (QEAC) [246]. The proposed algorithm was applied to four synthetic and four real-world datasets. This work explored the applicability of the QEA to data clustering [184]. QEAC was compared with the GA designed by Maulik and Bandyopadhyay [247] for data clustering and QEA for gene expression data clustering

developed by Zhou et al. [248]. As per the experimental results, the applicability and effectiveness of QEAC were duly established.

In 2014, Dey et al. proposed an automatic clustering algorithm influenced by quantum mechanics for multi-level image thresholding [249]. In this work, the principles of quantum computing were incorporated with the well-known genetic algorithm [41] for automatically identifying the optimal number of clusters from an image dataset. This work used the *CS*-measure [123] as a fitness function. The proposed algorithm was compared with its classical counterparts. The supremacy of the proposed algorithm over its classical equivalent was proven by several metrics, viz., standard deviation and standard error.

In 2017, Dey et al. proposed a quantum-inspired automatic clustering algorithm for the automatic clustering of grey-scale images [250]. This work presented the Quantum-Inspired Particle Swarm Optimisation (QIPSO) algorithm and the Quantum-Inspired Differential Evolution (QIDE) algorithm and the comparison with their classical counterparts. These algorithms outperformed two other state-of-the-art classical clustering algorithms both computationally and statistically. According to the experimental results, the QIPSO algorithm was found to excel over all other competitive algorithms in performance.

In 2018, Dey et al. proposed a quantum-inspired automatic clustering algorithm for grey-scale images [251,252]. This work incorporated the quantum computing principle with the single-solution-based Simulated Annealing algorithm. The proposed algorithm was compared with its classical equivalent. The supremacy of the proposed algorithm over its classical equivalent was established with respect to mean fitness value, standard deviation, standard error and optimum computational time.

In 2018, Dey et al. proposed an automatic clustering algorithm called the Quantum-Inspired Automatic Clustering Technique using Ant Colony Optimisation algorithm [253]. This technique was developed to determine the appropriate number of grey-scale image clusters. The superiority of the proposed technique was established based on the optimal number of clusters, computed fitness value, computational time, the mean, standard deviation of the fitness value, standard errors and an unpaired two-tailed *t*-test [254].

In 2018, the Quantum Spider Monkey Optimisation (QSMO) algorithm was proposed by Bhattacharyya et al. for the automatic clustering of grey-scale images [255]. The efficiency of this proposed algorithm inspired the researchers to develop Quantum-Inspired Ageist Spider Monkey Optimisation (QIASMO) [256] algorithm for the same purpose. This work explored the QIPSO algorithm [256], Quantum-Inspired Spider Monkey Optimisation algorithm [256] and QIASMO algorithm [256] for the automatic clustering of grey-scale images. The quantum-inspired algorithms were compared with their classical counterparts. In all cases, the quantum-inspired algorithms outperformed the state-of-the-art algorithms. QIASMO [256] was found to be the most effective algorithm.

In 2019, the Quantum-Inspired BAT (QIBAT) algorithm was proposed by Dey et al. to solve the problem of automatic clustering of grey-scale images [257]. After that, this work was extended in 2020 [258] where two quantum-inspired metaheuristic algorithms, viz., QIBA and QIGA, were introduced. In all aspects, the quantum versions of these algorithms were found to be superior to the state-of-the-art algorithms. The computational results showed that QIBA was the best-performing algorithm.

In 2021, Dey et al. proposed a quantum-inspired metaheuristic algorithm for the automatic clustering of colour images [259]. In this paper, the Crow Search Optimisation Algorithm (CSOA) [260] and Intelligent Crow Search Optimisation Algorithm (ICSOA) [261] were used as the underlying metaheuristic algorithms. The experimental results proved that the quantum-inspired algorithms were superior to their corresponding classical alternatives in all aspects. Moreover, among all other algorithms, the quantum-inspired intelligent crow search optimisation algorithm was found to be the best-performing algorithm in terms of performance.

Dutta et al. [262] developed an Automatic Clustering-Based Qutrit version of Particle Swarm Optimisation (AC-QuPSO) in 2021 to automatically cluster hyperspectral images. The space complexity was reduced using the band fusion approach, as it minimised the number of

bands present in each image. Shannon entropy [263] was used to choose the minimum number of bands for implementing the modified improved subspace decomposition technique [263]. In this paper, the authors introduced a new concept called *qutrit* instead of *qubits*. The proposed work was compared with the classical PSO [43,264] algorithm based on four metrics, viz., peak signal-to-noise ratio (PSNR) [265], Jaccard Index [266], Sørensen–Dice Similarity Index [267] and the computational time. All the experiments were conducted on the Salinas Dataset [268]. All the outcomes of the experiments, along with the statistical superiority test, viz., the unpaired *t*-test [254] proved the supremacy of the proposed algorithm over its classical alternative. Moreover, greater progress was made in reducing space and time complexity using the concept of *qutrit*.

A Quantum-Inspired Manta Ray Foraging Optimisation (QIMRFO) algorithm for the automatic clustering of colour images [269] was proposed by Dey et al. in 2022. This work presented a comparison between the proposed algorithm and the classical version of the Manta Ray Foraging Optimisation (MRFO) algorithm [270] as well as with the well-known GA [41]. The PBM index [122] was used as an objective function to demonstrate the goodness of the clustering results. The computational results indicated that the QIMRFO quantitatively and qualitatively outperformed other comparative algorithms.

In 2022, Dey et al. proposed two quantum-inspired metaheuristic algorithms, viz., QIPSO algorithm and the Quantum-Inspired Enhanced Particle Swarm Optimisation (QIEPSO) algorithm, for the automatic clustering of colour images [271]. In this work, the proposed algorithms were compared with their corresponding classical counterparts as well as three different well-known classical algorithms, viz., ABC [272], DE [206] and Covariance Matrix Adaption Evolution Strategies (CMA-ES) [273]. In this work, the goodness of the clustering was measured using three different cluster validity indices, viz., PBM [122], CSM [123] and DI [117]. During the experiments, ten Berkeley colour images [274] and ten real-life colour images [275] with different sizes were used. According to the experimental results, QIEPSO proved to be a potential candidate for the automatic clustering of colour images.

A detailed study of the contributions made in the field of single-objective quantum-inspired metaheuristic approaches to automatic clustering is presented in Tables 8 and 9. The listed algorithms dating between 2001 and 2021 are compared based on the aim of the concerned works, used mechanisms, data specifications, merits and demerits. All the listed algorithms significantly outperform their competing algorithms.

7.2. Multi-Objective Approaches

Nowadays, researchers have intended to achieve multi-objective optimisation based on quantum-inspired frameworks to solve several types of optimisation problems, specifically clustering problems [276]. In quantum-inspired multi-objective optimisation techniques, the superposition of quantum states generates the search space, thereby providing a good population diversity and increasing the search capabilities. Examples of a few quantum-inspired multi-objective optimisation techniques include the Quantum-Inspired Multi-Objective Evolutionary Clustering (QMEC) algorithm [276], a self-organising PSO algorithm based on the principle of the quantum computing mechanism to handle multi-modal multi-objective optimisation problems (MMO_SO_QPSO) [277], a new quantum-based multi-objective simulated annealing technique for bi-level thresholding [278], and an optimal VSM model based on a multi-objective Quantum-Inspired Genetic Algorithm to retrieve web information [279], to name a few. This section presents a few quantum-inspired metaheuristic algorithms based on the multi-objective optimisation framework for automatic clustering.

Table 8. Single-objective quantum-inspired metaheuristic approaches to automatic clustering.

Ref.	Aim of the Work	Mechanism Used	Data Specifications	Merits and Demerits
[249]	This study introduces an automatic clustering algorithm referred to as QIAGA for multi-level image thresholding which is capable of automatically identifying the optimal number of clusters from an image dataset (Dey et al., 2014).	This work incorporates the quantum computing mechanism with the well-known GA.	The test images include four real-life grey-scale images.	Merits: 1. It has a good convergence rate. 2. Experimental results statistically prove the efficiency and effectiveness of the proposed algorithm. Demerits: 1. Time complexity is higher.
[252]	This study presents a quantum-inspired automatic clustering algorithm for grey-scale images (Dey et al., 2018).	It incorporates the quantum computing principles with the single solution-based Simulated Annealing algorithm.	For experimental purposes, four real-life grey-scale images and four Berkeley images of different dimensions were used.	Merits: 1. It has a higher convergence rate. 2. The supremacy of the proposed algorithm over its classical equivalent has been established with respect to some metrics. Demerits: 1. It suffers from premature convergence.
[253]	This study introduces an automatic clustering algorithm referred to as the Quantum-Inspired Automatic Clustering Technique Using the Ant Colony Optimisation algorithm for the automatic identification of the optimal number of clusters in a grey-scale image (Dey et al., 2018).	The Ant Colony Optimisation algorithm was chosen as the underlying algorithm to be incorporated in a quantum computing framework. The Xie–Beni cluster validity measure was used as the objective function for this study.	All the experiments were conducted on four real-life grey-scale images.	Merits: 1. The proposed technique is found to be superior to its classical counterpart with regard to several factors such as accuracy, stability, computational speed and standard errors. Demerits: 1. It does not always attain global optimum.
[258]	This study introduces two quantum-inspired metaheuristic algorithms, viz., QIBA and QIGA, for the automatic clustering of grey-scale images (Dey et al., 2019).	Quantum computing principles are incorporated in this work with the classical Bat optimisation algorithm and GA. *DB*-index was used to validate the clustering process.	All the experiments were performed on four Berkeley images and two real-life grey-scale images.	Merits: 1. QIBA can efficiently balance exploration and exploitation in the search space. 2. Computational results indicate that QIBA outperforms others. Demerits: 1. The time complexity is higher.
[256]	This study presents three automatic clustering algorithms for grey-scale images (Dey et al., 2020).	This work includes the QIPSO, Quantum-Inspired Spider Monkey Optimisation algorithm (QISMO) and QIASMO algorithm (QIASMO).	All the experiments were conducted on five Berkeley images, five real-life images and four mathematical benchmark functions.	Merits: 1. QIASMO has better convergence speed than the others. 2. The quantum-inspired algorithms outperform state-of-the-art algorithms in all cases. 3. QIASMO has been regarded as being the most effective algorithm out of these three. Demerits:1. QIPSO may get stuck in the local optima.

Table 9. Single-objective quantum-inspired metaheuristic approaches to automatic clustering.

Ref.	Aim of the Work	Mechanism Used	Data Specifications	Merits and Demerits
[259]	This study presents two quantum-inspired metaheuristic algorithms referred to as the Quantum-Inspired Crow Search Optimisation Algorithm (QICSOA) and Quantum-Inspired Intelligent Crow Search Optimisation Algorithm (QIICSOA) for automatically clustering colour images (Dey et al., 2021).	For this study, the underlying metaheuristic algorithms include the CSOA and ICSOA. This work uses four cluster validity indices, viz., *PBM*-index, *I*-index, Silhouette (SIL) index and *CS*-Measure (CSM).	Fifteen Berkeley colour images and five publicly available real-life colour images with varied dimensions were used for experimental purposes.	Merits: 1. All of them have good convergence speed. 2. QIICSOA is found to be the most promising algorithm in terms of performance. Demerits: 1. Time complexity is higher.
[262]	This study introduces an automatic clustering algorithm for hyper-spectral images, which is referred to as AC-QuPSO (Dutta et al., 2021).	A new concept called *qutrit* was introduced to reduce space and time complexity.	All the experiments were performed on the Salinas Dataset.	Merits: 1. It has good convergence speed. 2. The supremacy of AC-QuPSO over its conventional alternatives was realised by performing the unpaired *t*-test. Demerits: 1. It has difficulties in the placement of the controller.
[269]	This study presents the QIMRFO algorithm for clustering of colour images on the run (Dey et al., 2022).	Quantum computing framework was combined with the classical version of the MRFO algorithm.	Four Berkeley colour images and four publicly available real-life colour images were used for experimental purposes.	Merits: 1. It has a good convergence rate. 2. It is capable of efficiently exploring and exploiting the search space. 3. Experimental results indicate that the QIMRFO quantitatively and qualitatively outperforms other comparative algorithms. Demerits: 1. In QIMRFO, the space requirement is high.
[271]	This paper presents two algorithms, viz., QIPSO algorithm and QIEPSO algorithm, for the automatic clustering of colour images (Dey et al., 2022).	Quasi quantum operations were performed to achieve the goal.	All the experiments were conducted on ten Berkeley images and ten real-life colour images.	Merits: 1. QIEPSO has good convergence speed. 2. The QIEPSO algorithm has enough potential to be a viable candidate for the automatic clustering of colour images. Demerits: 1. QIPSO does not always find an optimal solution.

In 2019, the Automatic Clustering Using Multi-Objective Emperor Penguin Optimiser (ACMOEPO) algorithm was developed by Kumar et al. [280] to automatically determine the optimal number of clusters from real-life datasets. Furthermore, this algorithm was also applied to four colour images, viz., Mandrill, Airplane, Peppers and House, for segmentation purposes. Two cluster validity indices were used to design a novel fitness function to maintain the balance between inter-cluster and intra-cluster distances. The superiority of this algorithm was established by comparing it with four other multi-objective clustering algorithms, viz., GenClustMOO [209], Multi-Objective Invasive Weed Optimisation algorithm for clustering (MOIWO) [281], automatic clustering using MOPSOSA [214] and an evolutionary approach to MOCK [197]. The performance of the participating algorithms was measured with respect to the obtained number of clusters, mean and standard devia-

tion. Finally, the superiority of the algorithm was established by performing the unpaired *t*-test [254] on all the participating algorithms.

In 2022, the Quantum-Inspired Multi-Objective NSGA-II algorithm for the Automatic Clustering of Grey-Scale Images (QIMONSGA-II) was proposed by Dey et al. [282] to automatically determine the exact number of clusters in a grey-scale image. This work compared the proposed algorithm and the well-known Non-Dominated Sorting Genetic Algorithm II (NSGA-II) [166,283]. QIMONSGA-II performs quasi-quantum computation to optimise two different objectives, viz., *CS*-Measure (CSM) [123] and *DB*-index [116] simultaneously. All the experiments were conducted over six Berkeley [274] grey-scale images of different sizes. In most cases, the experimental outcome achieved better results for the proposed QIMONSGA-II [282] algorithm. Finally, the superiority of the proposed algorithm was proved using the Minkowski score [284] and *SI* index [119].

A detailed study of the contributions made in the field of multi-objective quantum-inspired metaheuristic approaches to automatic clustering is presented in Table 10. The listed algorithms dating between 2001 and 2021 are compared based on the aim of the concerned works, used mechanisms, data specifications, merits and demerits. All the listed algorithms significantly outperform their competing algorithms.

Table 10. Multi-objective quantum-inspired metaheuristic approaches to automatic clustering.

Ref.	Aim of the Work	Mechanism Used	Data Specifications	Merits and Demerits
[280]	This paper presents the Automatic Clustering using Multi-Objective Emperor Penguin Optimiser (ACMOEPO) algorithm to automatically determine the optimal number of clusters from real-life datasets (Kumar et al., 2019).	To maintain the balance between inter-cluster and intra-cluster distances, a unique fitness function is suggested, which comprises multiple cluster validity indices.	The test dataset includes nine real-life benchmark datasets.	Merits: 1. It can handle very large datasets. 2. This type of algorithm is helpful in data mining applications. 3. The superiority of ACMOEPO was proved by performing the unpaired *t*-test among all the participating algorithms. Demerits: 1. Space requirement is higher.
[282]	This paper introduces the QIMONSGA-II for the automatic clustering of grey-scale images (Dey et al., 2022).	QIMONSGA-II performs quasi-quantum computation and simultaneously optimises two objectives, viz., the *CS*-Measure (CSM) and *DB*-index.	All the experiments were conducted over six Berkeley grey-scale images of varied dimensions.	Merits: 1. It can identify optimal results in a multi-objective environment. 2. The superiority of QIMONSGA-II was proven by utilising the Minkowski score and *SI* Demerits: 1. Space requirement is higher.

8. Conclusions

This paper presents an overview of metaheuristic algorithms for automatic clustering. A brief overview of the automatic clustering process is presented. The basics of the quantum computing paradigm have also been highlighted for the sake of self-sufficiency. Different state-of-the-art nature-inspired metaheuristic and automatic clustering algorithms based on the quantum-inspired framework were reviewed to identify common inspiring sources, classifications, merits and demerits. The major objective of this study was to present a clear view towards different clustering algorithms and their application areas. From the study, it is quite evident that the quantum-inspired metaheuristics prove to be superior to their classical counterparts when it comes to the automatic clustering of datasets. Moreover, these algorithms seem to be robust, time-efficient and fail-safe compared to their classical counterparts. The focus of this study was limited to providing examples of the most widely used clustering techniques. Further research in this direction would entail higher-level quantum states for designing higher-level quantum-inspired metaheuristics.

Abbreviations

The following abbreviations are used in this manuscript:

ABC	Artificial Bee Colony
ACDE	Automatic Clustering DE
ACMOEPO	Automatic Clustering Using Multi-Objective Emperor Penguin Optimiser
AC-QuPSO	Automatic Clustering-Based Qutrit version of Particle Swarm Optimisation
AFCH	Automatic Fuzzy *C*-Means with selected Initial Centres by Histogram
AFCR	Automatic Fuzzy *C*-Means with randomly selected Initial Centres
AP	Affinity Propagation
ASLoC	Automatic Security Log Clustering
ASO	Atom Search Optimisation
BA	Bat Algorithm
BEA	Bacterial Evolutionary Algorithm
BQANA	Binary Quantum-Based Avian Navigation Optimiser
BSMO	Binary Starling Murmuration Optimiser
CDA	Cluster Decomposition Algorithm
CH	Calinski–Harabasz
CLA	Communication with Local Agents
CLIQUE	Clustering In QUEst
CMA-ES	Covariance Matrix Adaption Evolution Strategies
CS	Cuckoo Search
CSM	CS-Measure
CSOA	Crow Search Optimisation Algorithm
CVI	Cluster Validity Index
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DB	Davies–Bouldin index
DCPSO	Dynamic Clustering Particle Swarm Optimisation
DE	Differential Evolution
DEMO	DE for Multi-Objective Optimisation
DI	Dunn Index
DPSC	Density Peaks Sentence Clustering
EM	Expectation Maximisation
FA	Firefly Algorithm
FAPSO	Firefly Particle Swarm Optimisation
FCM	Fuzzy C-Means
FCM-NSGA-II	Fuzzy *C*-Means-Non-Dominated Sorting Genetic Algorithm II
FCS	Fuzzy C-Shells
FRC-NSGA	Fuzzy Relational Clustering with NSGA-II
GA	Genetic Algorithm
GCA	Genetically Based Clustering Algorithm
GCUK	Genetic Clustering with an Unknown Number of Clusters *K*
GLS	Guided Local Search
GMDD	Gaussian Mixture Density Modelling Decomposition
GQA	Genetic Quantum Algorithm
GRNN	Generalised Regression Neural Network
GWO	Grey Wolf Optimiser

HCMA	Hierarchical Cluster Merging Algorithm
I	I Index
ICSOA	Intelligent Crow Search Optimisation Algorithm
IFRC-NSGA	Improved FRC with NSGA-II
KP	*K*-Prototypes
LCCV	Local Cores-Based Cluster Validity index
MA	Microcanonical Annealing
MAFIA	Merging of Adaptive Intervals Approach to Spatial Data Mining
MDS	Multi-Document Summarisation
MODE	Multi-Objective DE
MOCK	Multi-Objective Clustering with an Unknown Number of Clusters *K*
MOGA	Multi-Objective Genetic Algorithm
MPSO	Multi-objective-Based Particle Swarm Optimisation
MOPSOSA	Multi-Objective Particle Swarm Optimisation and Simulated Annealing
Mo-QIGA	Multi-Objective Quantum-Inspired Genetic Algorithm
MOSA	Multi-Objective Simulated Annealing
MOIWO	Multi-Objective Invasive Weed Optimisation
MQPSO	Many-Objective Quantum-Inspired Particle Swarm Optimisation Algorithm
MRFO	Manta Ray Foraging Optimisation
MS	Mean Shift
NSDE	Non-Dominated Sorting DE
NSGA-II	Non-Dominated Sorting Genetic Algorithm
OPTICS	Ordering Points To Identify the Clustering Structure
PBM	PBM Index
PDE	Pareto DE
PSNR	Peak Signal-to-Noise Ratio
PSO	Particle Swarm Optimisation
QANA	Quantum-Based Avian Navigation Optimiser Algorithm
QEA	Quantum Evolutionary Algorithm
QEAC	Quantum-Inspired Evolutionary Algorithm for Data Clustering
QIASMO	Quantum-Inspired Ageist Spider Monkey Optimisation
QIBAT	Quantum-Inspired BAT
QIGA	Quantum-Inspired Genetic Algorithm
QIMRFO	Quantum-Inspired Manta Ray Foraging Optimisation
QIEPSO	Quantum-Inspired Enhanced Particle Swarm Optimisation
QIMONSGA-II	Quantum-Inspired Multi-Objective NSGA-II Algorithm for Automatic Clustering of Grey-Scale Images
QIPSO	Quantum-Inspired Particle Swarm Optimisation
QSMO	Quantum Spider Monkey Optimisation
QMEC	Quantum-Inspired Multi-Objective Evolutionary Clustering
QTM	Quantum Turing Machine
SCA	Sine Cosine Algorithm
SI	Silhouette Index
SMO	Starling Murmuration Optimiser
SOM	Self-Organising Map
SOS	Symbiotic Organism Search
STING	Statistical Information Grid
STClu	Statistical Test-Based Clustering
SGA-KP	Single-Objective Genetic Algorithm with *K*-Means
TLBO	Teaching Learning-Based Optimisation
TS	Tabu Search
UCI	University of California at Irvine
UCDP	Uppsala Conflict Data Program
XB	Xie–Beni index

References

1. Jain, A.K.; Dubes, R.C. *Algorithms for Clustering Data*; Prentice-Hall, Inc.: Hoboken, NJ, USA, 1988.
2. Jain, A.K.; Murty, M.N.; Flynn, P.J. Data Clustering: A Review. *ACM Comput. Surv.* **1999**, *31*, 264–323. [CrossRef]
3. Roberts, S.J. Parametric and non-parametric unsupervised cluster analysis. *Pattern Recognit.* **1997**, *30*, 261–272. [CrossRef]
4. Gan, G.; Ma, C.; Wu, J. *Data Clustering: Theory, Algorithms, and Applications*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2007.
5. Faizan, M.; Zuhairi, M.F.; Ismail, S.; Sultan, S. Applications of Clustering Techniques in Data Mining: A Comparative Study. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 146–153. [CrossRef]
6. Dziechciarz-Duda, M. Marketing applications of cluster analysis to durables market segmentation. Klasyfikacja i analiza danych–teoria i zastosowania. *Taksonomia* **2007**, *14*, 523–532.
7. Karataş, A.; Şahin, S. Application Areas of Community Detection: A Review. In Proceedings of the International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT), Ankara, Turkey, 3–4 December 2018; pp. 65–70.
8. Yin, S.; Gan, G.; Valdez, E.A.; Vadiveloo, J. Applications of Clustering with Mixed Type Data in Life Insurance. *Risks* **2021**, *9*, 47. [CrossRef]
9. Tanwar, B. Clustering Techniques for Digital Image Segmentation. *Int. J. Sci. Eng. Res.* **2016**, *7*, 55.
10. Mittal, H.; Pandey, A.C.; Saraswat, M.; Kumar, S.; Pal, R.; Modwel, G. A comprehensive survey of image segmentation: Clustering methods, performance parameters, and benchmark datasets. *Multimed. Tools Appl.* **2021**, *81*, 35001–35026. [CrossRef]
11. Ramadas, M.; Abraham, A. Metaheuristics for Data Clustering and Image Segmentation. In *Proceedings of the Intelligent Systems Reference Library*; Springer: Berlin/Heidelberg, Germany, 2019.
12. Singh, S.; Srivastava, S. Review of Clustering Techniques in Control System: Review of Clustering Techniques in Control System. *Procedia Comput. Sci.* **2020**, *173*, 272–280. [CrossRef]
13. Gandhi, G.; Srivastava, R. Review Paper: A Comparative Study on Partitioning Techniques of Clustering Algorithms. *Int. J. Comput. Appl.* **2014**, *87*, 10–13. [CrossRef]
14. Saxena, A.; Prasad, M.; Gupta, A.; Bharill, N.; Patel, O.P.; Tiwari, A.; Er, M.J.; Ding, W.; Lin, C.T. A Review of Clustering Techniques and Developments. *Neurocomputing* **2017**, *267*, 664–681. [CrossRef]
15. Indhu, R.; Porkodi, R. Comparison of Clustering Algorithm. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol. (IJSRCSEIT)* **2018**, *3*, 218–223.
16. Abu Abbas, O. Comparisons Between Data Clustering Algorithms. *Int. Arab. J. Inf. Technol.* **2008**, *5*, 320–325.
17. Johnson, S. Hierarchical clustering schemes. *Psychometrika* **1967**, *32*, 241–254. [CrossRef]
18. Xu, D.; Tian, Y. A Comprehensive Survey of Clustering Algorithms. *Ann. Data Sci.* **2015**, *2*, 165–193. [CrossRef]
19. Murtagh, F. A Survey of Recent Advances in Hierarchical Clustering Algorithms. *Comput. J.* **1983**, *26*, 354–359. [CrossRef]
20. Gil-García, R.; Pons-Porrata, A. Dynamic hierarchical algorithms for document clustering. *Pattern Recognit. Lett.* **2010**, *31*, 469–477. [CrossRef]
21. Feng, L.; Qiu, M.H.; Wang, Y.X.; Xiang, Q.L.; Yang, Y.F.; Liu, K. A fast divisive clustering algorithm using an improved discrete particle swarm optimizer. *Pattern Recognit. Lett.* **2010**, *31*, 1216–1225. [CrossRef]
22. Wang, W.; Zhang, Y.; Li, y.; Zhang, X. The Global Fuzzy C-Means Clustering Algorithm. In Proceedings of the 6th World Congress on Intelligent Control and Automation, Dalian, China, 21–23 June 2006; Volume 1, pp. 3604–3607.
23. Dave, R.N.; Bhaswan, K. Adaptive fuzzy c-shells clustering and detection of ellipses. *IEEE Trans. Neural Networks* **1992**, *3*, 643–662. [CrossRef] [PubMed]
24. Yager, R.R.; Filev, D.P. Approximate clustering via the mountain method. *IEEE Trans. Syst. Man. Cybern.* **1994**, *24*, 1279–1284. [CrossRef]
25. MacQueen, J. Some methods for classification and analysis of multivariate observations. *Proc. Fifth Berkeley Symp. Math. Stat. Probab.* **1967**, *1*, 281–297.
26. Park, H.S.; Jun, C.H. A simple and fast algorithm for K-medoids clustering. *Expert Syst. Appl.* **2009**, *36*, 3336–3341. [CrossRef]
27. Kaufman, L.; Rousseeuw, P. *Finding Groups in Data: An Introduction to Cluster Analysis*; Wiley: New York, USA, 1990; pp. 126–163.
28. Ng, Raymond, T.; Han, J. CLARANS: A method for clustering objects for spatial data mining. *IEEE Trans. Knowl. Data Eng.* **2002**, *14*, 1003–1016. [CrossRef]
29. Kriegel, H.P.; Kröger, P.; Sander, J.; Zimek, A. Density-based Clustering. *Wiley Interdisc. Rew. Data Min. Knowl. Discov.* **2011**, *1*, 231–240. [CrossRef]
30. Ester, M.; Kriegel, H.P.; Sander, J.; Xu, X. *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*; AAAI Press: Palo Alto, CA, USA, 1996.
31. Ankerst, M.; Breunig, M.M.; Kriegel, H.P.; Sander, J. OPTICS: Ordering Points to Identify the Clustering Structure. *Acm Sigmod Rec.* **1999**, *28*, 49–60. [CrossRef]
32. Comaniciu, D.; Meer, P. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 603–619. [CrossRef]
33. Agrawal, R.; Gehrke, J.; Gunopulos, D.; Raghavan, P. Automatic subspace clustering of high dimensional data for data mining applications. In Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD '98), Seattle, WA, USA, 1–4 June 1998; pp. 94–105.

34. Wang, W.; Yang, J.; Muntz, R.R. STING: A Statistical Information Grid Approach to Spatial Data Mining. In Proceedings of the VLDB, Athens, Greece, 26–29 August 1997.

35. Goil, S.; Nagesh, H.; Choudhary, A. *MAFIA: Efficient and Scalable Subspace Clustering for Very Large Data Sets*; Technical Report NumberCPDC-TR-9906-019; Center for Parallel and Distributed Computing, Northwestern University: Evanston, IL, USA, 1999.

36. Sheikholeslami, G.; Chatterjee, S.; Zhang, A. Wavecluster: A multi-resolution clustering approach for very large spatial databases. In Proceedings of the VLDB, New York, NY, USA, 24–27 August 1998; pp. 428–439.

37. Si, Y.; Liu, P.; Li, P.; Brutnell, T.P. Model-based clustering for RNA-seq data. *Bioinformatics* **2013**, *30*, 197–205. [CrossRef]

38. Zhuang, X.; Huang, Y.; Palaniappan, K.; Zhao, Y. Gaussian mixture density modeling, decomposition, and applications. *IEEE Trans. Image Process.* **1996**, *5*, 1293–1302. [CrossRef]

39. Barbará, D.; Li, Y.; Couto, J. COOLCAT: An entropy-based algorithm for categorical clustering. In Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM '02), McLean, VA, USA, 4–9 November 2002; pp. 582–589.

40. Bay, S.D.; Pazzani, M.J. Detecting change in categorical data: Mining contrast sets. In Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '99), San Diego, CA, USA, 15–18 August 1999; pp. 302–306.

41. Goldberg, D.; Holland, J. Genetic Algorithms and Machine Learning. *Mach. Learn.* **1988**, *3*, 95–99. [CrossRef]

42. Storn, R.; Price, K.V. Differential Evolution—A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]

43. Eberhart, R.; Kennedy, J. A new optimizer using particle swarm theory. In Proceedings of the MHS'95—6th International Symposium on Micro Machine and Human Science, Nagoya, Japan, 4–6 October 1995; pp. 39–43.

44. Agbaje, M.; Ezugwu, A.; Els, R. Automatic Data Clustering Using Hybrid Firefly Particle Swarm Optimization Algorithm. *IEEE Access* **2019**, *7*, 184963–184984. [CrossRef]

45. Karaboga, D. *An Idea Based on Honey Bee Swarm for Numerical Optimization*; Technical Report—TR06; Erciyes University: Kayseri, Turkey, 2005; pp. 1–10.

46. Rajah, V.; Ezugwu, A.E. Hybrid Symbiotic Organism Search algorithms for Automatic Data Clustering. In Proceedings of the Conference on Information Communications Technology and Society (ICTAS), Virtual, 9–10 March 2020; pp. 1–9.

47. Das, S.; Chowdhury, A.; Abraham, A. A Bacterial Evolutionary Algorithm for automatic data clustering. In Proceedings of the Congress on Evolutionary Computation, Trondheim, Norway, 18–21 May 2009; pp. 2403–2410.

48. Kapoor, S.; Zeya, I.; Singhal, C.; Nanda, S. A Grey Wolf Optimizer Based Automatic Clustering Algorithm for Satellite Image Segmentation. *Procedia Comput. Sci.* **2017**, *115*, 415–422. [CrossRef]

49. Jensi, R.; Jiji, G.W. MBA-IF:A New Data Clustering Method Using Modified Bat Algorithm and Levy Flight. In Proceedings of the SOCO 2015, Burgos, Spain, 15–17 June 2015; pp. 15–17.

50. Goel, S.; Sharma, A.; Bedi, P. Cuckoo Search Clustering Algorithm: A novel strategy of biomimicry. In Proceedings of the World Congress on Information and Communication Technologies, Mumbai, India, 11–14 December 2011; pp. 916–921.

51. Rao, R.; Savsani, V.; Vakharia, D. Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems. *Comput. Aided Des.* **2011**, *43*, 303–315. [CrossRef]

52. Zamani, H.; Nadimi-Shahraki, M.H.; Gandomi, A.H. Starling murmuration optimizer: A novel bio-inspired algorithm for global and engineering optimization. *Comput. Methods Appl. Mech. Eng.* **2022**, *392*, 114616. [CrossRef]

53. Nadimi-Shahraki, M.H.; Asghari Varzaneh, Z.; Zamani, H.; Mirjalili, S. Binary Starling Murmuration Optimizer Algorithm to Select Effective Features from Medical Data. *Appl. Sci.* **2023**, *13*, 564. [CrossRef]

54. Montiel Ross, O.H. A Review of Quantum-Inspired Metaheuristics: Going From Classical Computers to Real Quantum Computers. *IEEE Access* **2020**, *8*, 814–838. [CrossRef]

55. Mani, N.; Srivastava, G.; Mani, A. Solving Combinatorial Optimization problems with Quantum inspired Evolutionary Algorithm Tuned using a Novel Heuristic Method. *arXiv* **2016**, arXiv:1612.08109.

56. Abs da Cruz, A.; Barbosa, C.; Pacheco, M.; Vellasco, M. Quantum-inspired evolutionary algorithms and its application to numerical optimization problems. *Lect. Notes Comput. Sci.* **2004**, *3316*, 212–217.

57. DBLP-Citation-Network V12. Available online: https://www.aminer.org/citation (accessed on 7 October 2021).

58. Amami, M.; Pasi, G.; Stella, F.; Faiz, R. An LDA-Based Approach to Scientific Paper Recommendation. In *Natural Language Processing and Information Systems*; Springer: Berlin/Heidelberg, Germany, 2016; Volume 9612, pp. 200–210.

59. Saha, R.; Tariq, M.T.; Hadi, M.; Xiao, Y. Pattern Recognition Using Clustering Analysis to Support Transportation System Management, Operations, and Modeling. *J. Adv. Transp.* **2019**, *2019*, 1–12. [CrossRef]

60. Jardine, N.; van Rijsbergen, C.J. The use of hierarchic clustering in information retrieval. *Inf. Storage Retr.* **1971**, *7*, 217–240. [CrossRef]

61. Oyelade, J.; Isewon, I.; Oladipupo, F.; Aromolaran, O.; Uwoghiren, E.; Ameh, F.; Achas, M.; Adebiyi, E. Clustering Algorithms: Their Application to Gene Expression Data. *Bioinform. Biol. Insights* **2016**, *10*, 237–253. [CrossRef]

62. Seetharaman, S.K.; Thouheed Ahmed, S.; Gunashree; Bhumika, P.; Ishwarya; Anusha, B. A Generalized Study on Data Mining and Clustering Algorithm. In *New Trends in Computational Vision and Bio-inspired Computing*; Springer: Berlin/Heidelberg, Germany, 2018.

63. Xu, J.; Liu, H. Web user clustering analysis based on K-Means algorithm. In Proceedings of the International Conference on Information, Networking and Automation (ICINA), Kunming, China, 18–19 October 2010; Volume 2, pp. 6–9.

64. Montanaro, A. Quantum algorithms: An overview. *Npj Quantum Inf.* **2016**, *2*, 15023. [CrossRef]
65. Jordan, S. The Quantum Algorithm Zoo. Available online: https://quantumalgorithmzoo.org/ (accessed on 7 October 2021).
66. Deutsch, D. Quantum theory, the Church–Turing principle and the universal quantum computer. *Proc. R. Soc. London. A. Math. Phys. Sci.* **1985**, *400*, 117–197.
67. Pour-El, M.B.; Richards, I. The wave equation with computable initial data such that its unique solution is not computable. *Adv. Math.* **1981**, *39*, 215–239. [CrossRef]
68. Benioff, P. Quantum Mechanical Models of Turing Machines That Dissipate No Energy. *Phys. Rev. Lett.* **1982**, *48*, 1581–1585. [CrossRef]
69. Feynman, R.P. Simulating physics with computers. *Int. J. Theor. Phys.* **1982**, *21*, 467–488. [CrossRef]
70. Nourbakhsh, A.; Jones, M.; Kristjuhan, K.; Carberry, D.; Karon, J.; Beenfeldt, C.; Shahriari, K.; Andersson, M.; Jadidi, M.; Mansouri, S. Quantum Computing: Fundamentals, Trends and Perspectives for Chemical and Biochemical Engineers. *arXiv* **2022**, arXiv:2201.02823.
71. Aung, D.M.M.; Aye, K.T.K.; Aung, T.M. On the Study of Quantum Computing. In Proceedings of the Conference on Science and Technology Development (CSTD-2019), Pyin Oo Lwin, Myanmar, 31 October–1 November 2019.
72. Deutsch, D.; Jozsa, R. Rapid Solution of Problems by Quantum Computation. *Proc. R. Soc. Lond. Ser. A* **1992**, *439*, 553–558.
73. Simon, D.R. On the power of quantum computation. In Proceedings of the 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, USA, 20–22 November 1994; pp. 116–123.
74. Shor, P.W. Algorithms for quantum computation: Discrete logarithms and factoring. In Proceedings of the 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, USA, 20–22 November 1994; pp. 124–134.
75. Grover, L.K. A fast quantum mechanical algorithm for database search. In Proceedings of the 28th Annual ACM Symposium on Theory of Computing, Philadelphia, PA, USA, 22–24 May 1996; pp. 212–219.
76. Hey, T. Quantum computing: An introduction. *Comput. Control Eng. J.* **1999**, *10*, 105–112. [CrossRef]
77. Blatt, R.; Häffner, H.; Roos, C.F.; Becher, C.; Schmidt-Kaler, F. Course 5—Quantum Information Processing in Ion Traps I. In *Quantum Entanglement and Information Processing*; Estève, D., Raimond, J.M., Dalibard, J., Eds.; Elsevier: Amsterdam, The Netherlands, 2004; Volume 79, pp. 223–260.
78. Vedral, V. Quantum entanglement. *Nat. Phys.* **2014**, *10*, 256–258. [CrossRef]
79. Li, B.; Yu, Z.H.; Fei, S.M. Geometry of Quantum Computation with Qutrits. *Sci. Rep.* **2013**, *3*, 2594. [CrossRef]
80. Gokhale, P.; Baker, J.M.; Duckering, C.; Chong, F.T.; Brown, N.C.; Brown, K.R. Extending the Frontier of Quantum Computers With Qutrits. *IEEE Micro* **2020**, *40*, 64–72. [CrossRef]
81. Chi, Y.; Huang, J.; Zhang, Z.; Mao, J.; Zhou, Z.; Chen, X.; Zhai, C.; Bao, J.; Dai, T.; Yuan, H.; et al. A programmable qudit-based quantum processor. *Nat. Commun.* **2022**, *13*, 1166. [CrossRef] [PubMed]
82. Luo, M.; Wang, X. Universal quantum computation with qudits. *Sci. China Physics Mech. Astron.* **2014**, *57*, 1712–1717. [CrossRef]
83. Wang, Y.; Hu, Z.; Sanders, B.C.; Kais, S. Qudits and high-dimensional quantum computing. *Front. Phys.* **2020**, *8*, 589504. [CrossRef]
84. Brylinski, J.L.; Brylinski, R. Universal quantum gates. In *Proceedings of the Mathematics of Quantum Computation*; American Mathematical Society: New York, NY, USA, 2002; p. 117.
85. Nielsen, M.A.; Chuang, I.L. *Quantum Computation and Quantum Information: 10th Anniversary Edition*; Cambridge University Press: Cambridge, UK, 2010.
86. Gómez, F.J.O.; Lopez, G.O.; Garzon, E.M. A Faster Half Subtractor Circuit Using Reversible Quantum Gates. *Balt. J. Mod. Comput.* **2019**, *7*, 99–111.
87. Fahdil, M.A.; Al-Azawi, A.F.; Said, S. Operations Algorithms on Quantum Computer. *IJCSNS Int. J. Comput. Sci. Netw. Secur.* **2010**, *10*, 85.
88. Acampora, G.; Vitiello, A. Implementing evolutionary optimization on actual quantum processors. *Inf. Sci.* **2021**, *575*, 542–562. [CrossRef]
89. Yang, S.; Wang, M.; Jiao, L. A novel quantum evolutionary algorithm and its application. In Proceedings of the Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753), Portland, OR, USA, 19–23 June 2004; Volume 1, pp. 820–826.
90. Narayanan, A.; Moore, M. Quantum-inspired genetic algorithms. In Proceedings of the International Conference on Evolutionary Computation, Nayoya, Japan, 20–22 May 1996; pp. 61–66.
91. Konar, D.; Sharma, K.; Sarogi, V.; Bhattacharyya, S. A Multi-Objective Quantum-Inspired Genetic Algorithm (Mo-QIGA) for Real-Time Tasks Scheduling in Multiprocessor Environment. *Procedia Comput. Sci.* **2018**, *131*, 591–599. [CrossRef]
92. Balicki, J. Many-Objective Quantum-Inspired Particle Swarm Optimization Algorithm for Placement of Virtual Machines in Smart Computing Cloud. *Entropy* **2022**, *24*, 58. [CrossRef]
93. Zamani, H.; Nadimi-Shahraki, M.H.; Gandomi, A.H. QANA: Quantum-based avian navigation optimizer algorithm. *Eng. Appl. Artif. Intell.* **2021**, *104*, 104314. [CrossRef]
94. Nadimi-Shahraki, M.H.; Fatahi, A.; Zamani, H.; Mirjalili, S. Binary Approaches of Quantum-Based Avian Navigation Optimizer to Select Effective Features from High-Dimensional Medical Data. *Mathematics* **2022**, *10*, 2770. [CrossRef]
95. Platt, J.C.; Czerwinski, M.; Field, B.A. PhotoTOC: Automatic clustering for browsing personal photographs. In Proceedings of the 4th International Conference on Information, Communications and Signal Processing, 2003 and the 4th Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint, Singapore, 15–18 December 2003; Volume 1, pp. 6–10.

96. Lei, T.; Liu, P.; Jia, X.; Zhang, X.; Meng, H.; Nandi, A.K. Automatic Fuzzy Clustering Framework for Image Segmentation. *IEEE Trans. Fuzzy Syst.* **2020**, *28*, 2078–2092. [CrossRef]

97. Tseng, L.Y.; Bien Yang, S. A genetic approach to the automatic clustering problem. *Pattern Recognit.* **2001**, *34*, 415–424. [CrossRef]

98. Azhir, E.; Navimipour, N.J.; Hosseinzadeh, M.; Sharifi, A.; Darwesh, A. An automatic clustering technique for query plan recommendation. *Inf. Sci.* **2021**, *545*, 620–632. [CrossRef]

99. Chen, J.H.; Chang, Y.C.; Hung, W.L. A robust automatic clustering algorithm for probability density functions with application to categorizing color images. *Commun. Stat. Simul. Comput.* **2018**, *47*, 2152–2168. [CrossRef]

100. Geraud, T.; Strub, P.; Darbon, J. Color image segmentation based on automatic morphological clustering. In Proceedings of the International Conference on Image Processing (Cat. No.01CH37205), Thessaloniki, Greece, 7–10 October 2001; Volume 3, pp. 70–73.

101. Zhu, S.; Xu, L.; Cao, L. A Study of Automatic Clustering Based on Evolutionary Many-Objective Optimization. In Proceedings of the Genetic and Evolutionary Computation Conference Companion, Kyoto, Japan, 15–19 July 2018; pp. 173–174.

102. Binu, D. Cluster analysis using optimization algorithms with newly designed objective functions. *Expert Syst. Appl.* **2015**, *42*, 5848–5859. [CrossRef]

103. Wang, C.W.; Hwang, J.I. Automatic clustering using particle swarm optimization with various validity indices. In Proceedings of the 5th International Conference on BioMedical Engineering and Informatics, Chongqing, China, 16–18 October 2012; pp. 1557–1561.

104. Tsai, C.W.; Liao, Y.H.; Chiang, M.C. A quantum-inspired evolutionary clustering algorithm. In Proceedings of the International Conference on Fuzzy Theory and Its Applications (iFUZZY), Taipei, Taiwan, 6–8 December 2013; pp. 305–310.

105. Li, Y.; Shi, H.; Gong, M.; Shang, R. Quantum-Inspired Evolutionary Clustering Algorithm Based on Manifold Distance. In Proceedings of the 1st ACM/SIGEVO Summit on Genetic and Evolutionary Computation, Shanghai, China, 12–14 June 2009; pp. 871–874.

106. Theodoridis, S.; Koutroumbas, K. *Pattern Recognition*, 4th ed.; Academic Press: Cambridge, MA, USA, 2009.

107. Xie, X.; Beni, G. A validity measure for fuzzy clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* **1991**, *13*, 841–847. [CrossRef]

108. Kim, M.; Ramakrishna, R. New indices for cluster validity assessment. *Pattern Recognit. Lett.* **2005**, *26*, 2353–2363. [CrossRef]

109. Zhou, M.; Şenol, A. VIASCKDE Index: A Novel Internal Cluster Validity Index for Arbitrary-Shaped Clusters Based on the Kernel Density Estimation. *Comput. Intell. Neurosci.* **2022**, *2022*, 1687–5265.

110. Saha, S.; Bandyopadhyay, S. Some connectivity based cluster validity indices. *Appl. Soft Comput.* **2012**, *12*, 1555–1565. [CrossRef]

111. Arbelaitz, O.; Gurrutxaga, I.; Muguerza, J.; Pérez, J.; Perona, I. An extensive comparative study of cluster validity indices. *Pattern Recognit.* **2013**, *46*, 243–256. [CrossRef]

112. José-García, A.; Gómez-Flores, W. A Survey of Cluster Validity Indices for Automatic Data Clustering Using Differential Evolution. In Proceedings of Genetic and Evolutionary Computation Conference, Lille, France, 10–14 July 2021; pp. 314–322.

113. Liu, Y.; Li, Z.; Xiong, H.; Gao, X.; Wu, J.; Wu, S. Understanding and Enhancement of Internal Clustering Validation Measures. *IEEE Trans. Cybern.* **2013**, *43*, 982–994.

114. Hu, L.; Zhong, C. An Internal Validity Index Based on Density-Involved Distance. *IEEE Access* **2019**, *7*, 40038–40051. [CrossRef]

115. Li, Q.; Yue, S.; Wang, Y.; Ding, M.; Li, J. A New Cluster Validity Index Based on the Adjustment of Within-Cluster Distance. *IEEE Access* **2020**, *8*, 202872–202885. [CrossRef]

116. Davies, D.L.; Bouldin, D.W. A Cluster Separation Measure. *IEEE Trans. Pattern Anal. Mach. Intell.* **1979**, *PAMI-1*, 224–227. [CrossRef]

117. Dunn, J.C. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *J. Cybern.* **1973**, *3*, 32–57. [CrossRef]

118. Caliński, T.; Harabasz, J. A dendrite method for cluster analysis. *Commun. Stat.* **1974**, *3*, 1–27.

119. Rousseeuw, P. Rousseeuw, P.J. Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. *J. Comput. Appl. Math.* **1987**, *20*, 53–65. [CrossRef]

120. Bezdek, J.C. Cluster Validity. In *Pattern Recognition with Fuzzy Objective Function Algorithms*; Springer: Boston, MA, USA, 1981; pp. 95–154.

121. Halkidi, M.; Vazirgiannis, M. Clustering Validity Assessment: Finding the optimal partitioning of a data set. In Proceedings of the International Conference on Data Mining, San Jose, CA, USA, 29 November–2 December 2001; pp. 187–194.

122. Maulik, U.; Bandyopadhyay, S. Performance evaluation of some clustering algorithms and validity indices. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 1650–1654. [CrossRef]

123. Chou, C.H.; Su, M.C.; Lai, E. A new cluster validity measure and its application to image compression. *Pattern Anal. Appl.* **2004**, *7*, 205–220. [CrossRef]

124. Pakhira, M.K.; Bandyopadhyay, S.; Maulik, U. Validity index for crisp and fuzzy clusters. *Pattern Recognit.* **2004**, *37*, 487–501. [CrossRef]

125. Cheng, D.; Zhu, Q.; Huang, J.; Wu, Q.; Yang, L. A Novel Cluster Validity Index Based on Local Cores. *IEEE Trans. Neural Networks Learn. Syst.* **2019**, *30*, 985–999. [CrossRef] [PubMed]

126. Husain, H.; Khalid, M.; Yusof, R. Automatic clustering of generalized regression neural network by similarity index based fuzzy c-means clustering. In Proceedings of the Region 10 Conference TENCON 2004, Chiang Mai, Thailand, 24 November 2004; Volume 2, pp. 302–305.

127. Specht, D.F.; Shapiro, P.D. Generalization accuracy of probabilistic neural networks compared with backpropagation networks. In Proceedings of the IJCNN-91-Seattle International Joint Conference on Neural Networks,Seattle, WA, USA, 8–12 July 1991; Volume 1, pp. 887–892.

128. Specht, D. A General Regression Neural Network. *IEEE Trans. Neural Networks* **1991**, *2*, 568–578. [CrossRef] [PubMed]

129. Nosovskiy, G.V.; Liu, D.; Sourina, O. Automatic clustering and boundary detection algorithm based on adaptive influence function. *Pattern Recognit.* **2008**, *41*, 2757–2776. [CrossRef]

130. Li, L.; Yu, Z.; Feng, Z.; Zhang, X. Automatic classification of uncertain data by soft classifier. In Proceedings of the International Conference on Machine Learning and Cybernetics, Guilin, China, 10–13 July 2011; Volume 2, pp. 679–684.

131. Zhang, Y.; Xia, Y.; Liu, Y.; Wang, W. Clustering Sentences with Density Peaks for Multi-document Summarization. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, CO, USA, 31 May–5 June 2015; pp. 1262–1267.

132. Rodriguez, A.; Laio, A. Clustering by fast search and find of density peaks. *Science* **2014**, *344*, 1492–1496. [CrossRef] [PubMed]

133. DUC 2004: Documents, Tasks, and Measures (Some Comparisons to DUC 2003). Available online: https://duc.nist.gov/duc2004/ (accessed on 7 October 2021).

134. Conroy, J.M.; Schlesinger, J.D.; Goldstein, J.; O'Leary, D.P. Left-Brain/Right-Brain Multi-Document Summarization. In Proceedings of the Document Understanding Conference (DUC 2004), Boston, MA, USA, 6–7 May 2004.

135. Radev, D.R.; Jing, H.; Styś, M.; Tam, D. Centroid-based summarization of multiple documents. *Inf. Process. Manag.* **2004**, *40*, 919–938. [CrossRef]

136. Wan, X.; Yang, J. Multi-document summarization using cluster-based link analysis. In Proceedings of the 31st Annual international ACM SIGIR Conference on Research and Development in Information Retrieval, Singapore, 20–24 July 2008; pp. 299–306.

137. Wang, D.; Li, T.; Zhu, S.; Ding, C. Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization. In Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Singapore, 20–24 July 2008; pp. 307–314.

138. Cai, X.; Li, W. Ranking Through Clustering: An Integrated Approach to Multi-Document Summarization. *IEEE Trans. Audio Speech Lang. Process.* **2013**, *21*, 1424–1433.

139. Wang, D.; Zhu, S.; Li, T.; Chi, Y.; Gong, Y. Integrating Document Clustering and Multidocument Summarization. *TKDD* **2011**, *5*, 14. [CrossRef]

140. Erkan, G.; Radev, D.R. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.* **2004**, *22*, 457–479. [CrossRef]

141. Lin, H.; Bilmes, J. A Class of Submodular Functions for Document Summarization. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, OR, USA, 19–24 June 2011; Volume 1, pp. 510–520.

142. Wang, D.; Li, T. Weighted consensus multi-document summarization. *Inf. Process. Manag.* **2012**, *48*, 513–523. [CrossRef]

143. Wang, G.; Song, Q. Automatic Clustering via Outward Statistical Testing on Density Metrics. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 1971–1985. [CrossRef]

144. Fränti, P.; Sieranoja, S. K-means properties on six clustering benchmark datasets. *Appl. Intell.* **2018**, *48*, 4743–4759. [CrossRef]

145. Chen, Z.; Chang, D.; Zhao, Y. An Automatic Clustering Algorithm Based on Region Segmentation. *IEEE Access* **2018**, *6*, 74247–74259. [CrossRef]

146. Cassisi, C.; Ferro, A.; Giugno, R.; Pigola, G.; Pulvirenti, A. Enhancing density-based clustering: Parameter reduction and outlier detection. *Inf. Syst.* **2013**, *38*, 317–330. [CrossRef]

147. Cheng, Q.; Lu, X.; Liu, Z.; Huang, J.; Cheng, G. Spatial clustering with Density-Ordered tree. *Phys. A Stat. Mech. Its Appl.* **2016**, *460*, 188–200. [CrossRef]

148. Ram, A.; Sharma, A.; Jalal, A.S.; Agrawal, A.; Singh, R. An enhanced density based spatial clustering of applications with noise. In Proceedings of the International Advance Computing Conference, Patiala, India, 6–7 March 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 1475–1478.

149. Sundberg, R.; Melander, E. Introducing the UCDP georeferenced event dataset. *J. Peace Res.* **2013**, *50*, 523–532. [CrossRef]

150. Yangyang, H.; Zengli, L. Fuzzy clustering algorithm for automatically determining the number of clusters. In Proceedings of the International Conference on Signal Processing, Communications and Computing (ICSPCC), Dalian, China, 20–22 September 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–4.

151. Wang, L.; Zheng, K.; Tao, X.; Han, X. Affinity propagation clustering algorithm based on large-scale data-set. *Int. J. Comput. Appl.* **2018**, *40*, 1–6. [CrossRef]

152. Studiawan, H.; Payne, C.; Sohel, F. Automatic Graph-Based Clustering for Security Logs. In *Advanced Information Networking and Applications*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 914–926.

153. Hofstede, R.; Hendriks, L.; Sperotto, A.; Pras, A. SSH Compromise Detection Using NetFlow/IPFIX. *SIGCOMM Comput. Commun. Rev.* **2014**, *44*, 20–26. [CrossRef]

154. Sconzo, M. SecRepo.com: Security Data Samples Repository. 2014. Available online: http://www.secrepo.com (accessed on 7 October 2021).

155. Chuvakin, A. Scan 34 2005 from The Honeynet Project. 2005. Available online: https://seclists.org/focus-ids/2005/Apr/21 (accessed on 7 October 2021).

156. National CyberWatch Center. *Snort Fast Alert Logs from The U.S. National Cyber-Watch (MACCDC)*; National CyberWatch Center: Largo, MD, USA, 2012.

157. Chuvakin, A. Free Honeynet Log Data for Research. Available online: http://honeynet.org/node/456/ (accessed on 7 October 2021).

158. Sahoo, A.; Parida, P. Automatic clustering based approach for brain tumor extraction. *J. Physics Conf. Ser.* **2021**, *1921*, 012007. [CrossRef]

159. Wang, Z.; Yu, Z.; Chen, C.; You, J.; Gu, T.; Wong, H.; Zhang, J. Clustering by Local Gravitation. *IEEE Trans Cybern* **2017**, *48*, 1383–1396. [CrossRef]

160. Ruba, T.; Beham, D.M.P.; Tamilselvi, R.; Rajendran, T. Accurate Classification and Detection of Brain Cancer Cells in MRI and CT Images using Nano Contrast Agents. *Biomed. Pharmacol. J.* **2020**, *13*, 1227–1237. [CrossRef]

161. Chahar, V.; Katoch, S.; Chauhan, S. A Review on Genetic Algorithm: Past, Present, and Future. *Multimed. Tools Appl.* **2021**, *80*, 8091–8126.

162. Beheshti, Z.; Shamsuddin, S.M. A review of population-based meta-heuristic algorithm. *Int. J. Adv. Soft Comput. Its Appl.* **2013**, *5*, 1–35.

163. Talbi, E.G.; Basseur, M.; Nebro, A.; Alba, E. Multi-objective optimization using metaheuristics: Non-standard algorithms. *Int. Trans. Oper. Res.* **2012**, *19*, 283–305. [CrossRef]

164. Suresh, K.; Kundu, D.; Ghosh, S.; Das, S.; Abraham, A. Data Clustering Using Multi-objective Differential Evolution Algorithms. *Fundam. Inform.* **2009**, *97*, 381–403. [CrossRef]

165. Bandyopadhyay, S.; Saha, S.; Maulik, U.; Deb, K. A Simulated Annealing-Based Multiobjective Optimization Algorithm: AMOSA. *IEEE Trans. Evol. Comput.* **2008**, *12*, 269–283. [CrossRef]

166. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]

167. José-García, A.; Gómez-Flores, W. Automatic clustering using nature-inspired metaheuristics: A survey. *Appl. Soft Comput.* **2016**, *41*, 192–213. [CrossRef]

168. Ezugwu, A.; Shukla, A.; Agbaje, M.; José-García, A.; Olaide, O.; Agushaka, O. Automatic clustering algorithms: A systematic review and bibliometric analysis of relevant literature. *Neural Comput. Appl.* **2021**, *33*, 6247–6306. [CrossRef]

169. Talbi, E.G. Single-Solution Based Metaheuristics. In *Metaheuristics: From Design to Implementation*; Wiley: Hoboken, NJ, USA, 2009; Volume 74, pp. 87–189.

170. Van Laarhoven, P.J.M.; Aarts, E.H.L. Simulated annealing. In *Simulated Annealing: Theory and Applications*; Springer: Berlin/Heidelberg, Germany, 1987; pp. 7–15.

171. Glover, F. Tabu search—Part I. *ORSA J. Comput.* **1989**, *1*, 190–206. [CrossRef]

172. Linhares, A.; Torreão, J.R.A. Microcanonical optimization applied to the traveling salesman problem. *Int. J. Mod. Phys. C* **1998**, *9*, 133–146. [CrossRef]

173. Voudouris, C.; Tsang, E.; Alsheddy, A. Guided Local Search. In *Handbook of Metaheuristics*; Springer: Boston, MA, USA, 2010; pp. 321–361.

174. Dubes, R.; Jain, A.K. Clustering techniques: The user's dilemma. *Pattern Recognit.* **1976**, *8*, 247–260. [CrossRef]

175. Defays, D. An efficient algorithm for a complete link method. *Comput. J.* **1977**, *20*, 364–366. [CrossRef]

176. Garofolo, J.S.; Lamel, L.F.; Fisher, W.M.; Fiscus, J.G.; Pallett, D.S. DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1. *NASA STI/Recon Tech. Rep. N* **1993**, *93*, 27403.

177. Van der Merwe, D.; Engelbrecht, A. Data clustering using particle swarm optimization. In Proceedings of the Congress on Evolutionary Computation, CEC '03, Canberra, ACT, Australia, 8–12 December 2003; Volume 1, pp. 215–220.

178. Garai, G.; Chaudhuri, B. A novel genetic algorithm for automatic clustering. *Pattern Recognit. Lett.* **2004**, *25*, 173–187. [CrossRef]

179. Duda, R.O.; Hart, P.E. *Pattern Classification and Scene Analysis*; Wiley: Hoboken, NJ, USA, 1974.

180. Fisher, R. The use of multiple measurements in taxonomic problems. *Ann. Eugen.* **1936**, *7*, 179–188. [CrossRef]

181. Guha, S.; Rastogi, R.; Shim, K. CURE: An efficient clustering algorithm for large databases. *ACM Sigmod Rec.* **1998**, *27*, 73–84. [CrossRef]

182. Karypis, G.; Han, E.H.; Kumar, V. Chameleon: Hierarchical clustering using dynamic modeling. *Computer* **1999**, *32*, 68–75. [CrossRef]

183. Das, S.; Abraham, A.; Konar, A. Automatic Clustering Using an Improved Differential Evolution Algorithm. *IEEE Trans. Syst. Man, Cybern. Part A Syst. Humans* **2008**, *38*, 218–237. [CrossRef]

184. Bandyopadhyay, S.; Maulik, U. Genetic clustering for automatic evolution of clusters and application to image classification. *Pattern Recognit.* **2002**, *35*, 1197–1208. [CrossRef]

185. Omran, M.; Engelbrecht, A.; Salman, A. Dynamic Clustering using Particle Swarm Optimization with Application in Unsupervised Image Classification. In Proceedings of the 5th World Enformatika Conference (ICCI 2005), Prague, Czech Republic, 2005; pp. 199–204.

186. Karaboga, D.; Ozturk, C. A novel clustering approach: Artificial Bee Colony (ABC) algorithm. *Appl. Soft Comput.* **2011**, *11*, 652–657. [CrossRef]

187. De Falco, I.; Della Cioppa, A.; Tarantino, E. Facing classification problems with Particle Swarm Optimization. *Appl. Soft Comput.* **2007**, *7*, 652–658. [CrossRef]

188. Chen, C.Y.; Ye, F. Particle swarm optimization algorithm and its application to clustering analysis. In Proceedings of the 17th Conference on Electrical Power Distribution, Tehran, Iran, 2–3 May 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 789–794.

189. Pacheco, T.M.; Gonçalves, L.B.; Ströele, V.; Soares, S.S.R. An ant colony optimization for automatic data clustering problem. In Proceedings of the Congress on Evolutionary Computation (CEC), Rio de Janeiro, Brazil, 8–13 July 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–8.

190. Abd Elaziz, M.; Nabil, N.; Ewees, A.A.; Lu, S. Automatic data clustering based on hybrid atom search optimization and sine-cosine algorithm. In Proceedings of the Congress on evolutionary computation (CEC), Wellington, New Zealand, 10–13 June 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 2315–2322.

191. Zhao, W.; Wang, L.; Zhang, Z. Atom search optimization and its application to solve a hydrogeologic parameter estimation problem. *Knowl. Based Syst.* **2019**, *163*, 283–304. [CrossRef]

192. Mirjalili, S. SCA: A Sine Cosine Algorithm for solving optimization problems. *Knowl. Based Syst.* **2016**, *96*, 120–133. [CrossRef]

193. Alrosan, A.; Alomoush, W.; Alswaitti, M.; Alissa, K.; Sahran, S.; Makhadmeh, S.N.; Alieyan, k. Automatic Data Clustering Based Mean Best Artificial Bee Colony Algorithm. *Comput. Mater. Contin.* **2021**, *68*, 1575–1593. [CrossRef]

194. Ozturk, C.; Hancer, E.; Karaboga, D. Dynamic clustering with improved binary artificial bee colony algorithm. *Appl. Soft Comput.* **2015**, *28*, 69–80. [CrossRef]

195. Martin, D.; Fowlkes, C.; Tal, D.; Malik, J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In Proceedings of the 8th IEEE International Conference on Computer Vision, Vancouver, BC, Canada, 7–14 July 2001; IEEE: Piscataway, NJ, USA, 2001; Volume 2, pp. 416–423.

196. Suresh, K.; Kundu, D.; Ghosh, S.; Das, S.; Abraham, A. Automatic clustering with multi-objective Differential Evolution algorithms. In Proceedings of the Congress on Evolutionary Computation, Trondheim, Norway, 18–21 May 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 2590–2597.

197. Handl, J.; Knowles, J. An Evolutionary Approach to Multiobjective Clustering. *IEEE Trans. Evol. Comput.* **2007**, *11*, 56–76. [CrossRef]

198. Xue, F.; Sanderson, A.C.; Graves, R.J. Pareto-based multi-objective differential evolution. In Proceedings of the The Congress on Evolutionary Computation, Canberra, ACT, Australia, 8–12 December 2003; pp. 862–869.

199. Tusar, T.; Filipic, B. *DEMO: Differential Evolution for Multiobjective Optimization*; Institut Jozef Stefan: Ljubljana, Slovenia, 2005; pp. 520–533.

200. Abbass, H.; Sarker, R. The Pareto Differential Evolution Algorithm. *Int. J. Artif. Intell. Tools* **2002**, *11*, 531–552. [CrossRef]

201. Bandyopadhyay, S.; Maulik, U.; Mukhopadhyay, A. Multiobjective Genetic Clustering for Pixel Classification in Remote Sensing Imagery. *IEEE Trans. Geosci. Remote Sens.* **2007**, *45*, 1506–1511. [CrossRef]

202. Matake, N.; Hiroyasu, T.; Miki, M.; Senda, T. Multiobjective clustering with automatic k-determination for large-scale data. In Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, London, UK, 7–11 July 2007; pp. 861–868.

203. Blake, C.; Keough, E.; Merz, C.J. UCI Repository of Machine Learning Database. Available online: http://www.ics.uci.edu/~mlearn/MLrepository.html (accessed on 7 October 2021).

204. Sporulation Dataset. Available online: http://cmgm.stanford.edu/pbrown/sporulation (accessed on 7 October 2021).

205. Kundu, D.; Suresh, K.; Ghosh, S.; Das, S.; Abraham, A.; Badr, Y. Automatic Clustering Using a Synergy of Genetic Algorithm and Multi-objective Differential Evolution. In *Proceedings of the Hybrid Artificial Intelligence Systems*; Corchado, E., Wu, X., Oja, E., Herrero, Á., Baruque, B., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; pp. 177–186.

206. Storn, R.; Price, P. Differential Evolution—A simple and efficient adaptive scheme for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 347. [CrossRef]

207. Price, K.; Storn, R.M.; Lampinen, J.A. *Differential Evolution: A Practical Approach to Global Optimization*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2006.

208. Bezdek, J.C. Cluster validity with fuzzy sets. *J. Cybern.* **1973**, *3*, 58–72. [CrossRef]

209. Saha, S.; Bandyopadhyay, S. A generalized automatic clustering algorithm in a multiobjective framework. *Appl. Soft Comput.* **2013**, *13*, 89–108. [CrossRef]

210. Bandyopadhyay, S.; Saha, S. A Point Symmetry-Based Clustering Technique for Automatic Evolution of Clusters. *IEEE Trans. Knowl. Data Eng.* **2008**, *20*, 1441–1457. [CrossRef]

211. Pal, S.K.; Mitra, S. Fuzzy versions of Kohonen's net and MLP-based classification: Performance evaluation for certain nonconvex decision regions. *Inf. Sci.* **1994**, *76*, 297–337. [CrossRef]

212. Friedman, M. The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance. *Am. Stat. Assoc.* **1937**, *32*, 675–701. [CrossRef]

213. Nemenyi, P.B. Distribution-free Multiple Comparisons. Ph.D. Thesis, Princeton University, Princeton, NJ, USA, 1963.

214. Abubaker, A.; Baharum, A.; Alrefaei, M. Automatic Clustering Using Multi-objective Particle Swarm and Simulated Annealing. *PLoS ONE* **2015**, *10*, e0130995. [CrossRef]

215. Shieh, H.L.; Kuo, C.C.; Chiang, C.M. Modified particle swarm optimization algorithm with simulated annealing behavior and its numerical verification. *Appl. Math. Comput.* **2011**, *218*, 4365–4383. [CrossRef]

216. Ulungu, B.; Teghem, J.; Fortemps, P. Heuristic for multi-objective combinatorial optimization problems by simulated annealing. In *Proceedings of the MCDM: Theory and Applications*; SciTech: Encinitas, CA, USA, 1995; pp. 229–238.

217. Lichman, M. *UCI Machine Learning Repository*; University of California: Irvine, CA, USA, 2013.

218. Paul, A.K.; Shill, P.C. New automatic fuzzy relational clustering algorithms using multi-objective NSGA-II. *Inf. Sci.* **2018**, *448–449*, 112–133. [CrossRef]
219. Skabar, A.; Abdalgader, K. Clustering Sentence-Level Text Using a Novel Fuzzy Relational Clustering Algorithm. *IEEE Trans. Knowl. Data Eng.* **2013**, *25*, 62–75. [CrossRef]
220. Wikaisuksakul, S. A multi-objective genetic algorithm with fuzzy c-means for automatic data clustering. *Appl. Soft Comput.* **2014**, *24*, 679–691. [CrossRef]
221. Alon, U.; Barkai, N.; Notterman, D.A.; Gish, K.; Ybarra, S.; Mack, D.; Levine, A.J. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natl. Acad. Sci. USA* **1999**, *96*, 6745–6750. [CrossRef] [PubMed]
222. Singh, D.; Febbo, P.G.; Ross, K.; Jackson, D.G.; Manola, J.; Ladd, C.; Tamayo, P.; Renshaw, A.A.; D'Amico, A.V.; Richie, J.P. Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell* **2002**, *1*, 203–209. [CrossRef] [PubMed]
223. O'Neill, M.C.; Song, L. Neural network analysis of lymphoma microarray data: Prognosis and diagnosis near-perfect. *BMC Bioinform.* **2003**, *4*, 13.
224. Golub, T.R.; Slonim, D.K.; Tamayo, P.; Huard, C.; Gaasenbeek, M.; Mesirov, J.P.; Coller, H.; Loh, M.L.; Downing, J.R.; Caligiuri, M.A.; et al. Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. *Science* **1999**, *286*, 531–537. [CrossRef]
225. Real Life Data Set. Available online: https://archive.ics.uci.edu/ml/machine-learning-databases (accessed on 23 September 2020).
226. Dutta, D.; Sil, J.; Dutta, P. Automatic Clustering by Multi-Objective Genetic Algorithm with Numeric and Categorical Features. *Expert Syst. Appl.* **2019**, *137*, 357–379. [CrossRef]
227. Chen, E.; Wang, F. Dynamic Clustering Using Multi-objective Evolutionary Algorithm. In *Proceedings of the Computational Intelligence and Security*; Hao, Y., Liu, J., Wang, Y., Cheung, Y.m., Yin, H., Jiao, L., Ma, J., Jiao, Y.C., Eds.; Springer: Berlin/Heidelberg, Germany, 2005; pp. 73–80.
228. Huang, Z. Clustering Large Data Sets with Mixed Numeric and Categorical Values. In Proceedings of the 1st Pacific-Asia Conference on Knowledge Discovery and Data Mining, (PAKDD), Singapore, 23–24 February 1997; pp. 21–34.
229. Bezdek, J.C.; Ehrlich, R.; Full, W. FCM: The fuzzy c-means clustering algorithm. *Comput. Geosci.* **1984**, *10*, 191–203. [CrossRef]
230. Cheng, Y. Mean shift, mode seeking, and clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* **1995**, *17*, 790–799. [CrossRef]
231. King, B. Step-Wise Clustering Procedures. *J. Am. Stat. Assoc.* **1967**, *62*, 86–101. [CrossRef]
232. Kohonen, T. The self-organizing map. *Proc. IEEE* **1990**, *78*, 1464–1480. [CrossRef]
233. Rahman, M.A.; Islam, M.Z. A hybrid clustering technique combining a novel genetic algorithm with K-Means. *Knowl. Based Syst.* **2014**, *71*, 345–365. [CrossRef]
234. Asuncion, A.; Newman, D. UCI Machine Learning Repository. 2007. Available online: https://archive.ics.uci.edu/ml/index.php (accessed on 23 September 2020).
235. Fisher, R. Statistical Methods and Scientific Induction. *J. R. Stat. Society. Ser. B (Methodological)* **1955**, *17*, 69–78. [CrossRef]
236. Qu, H.; Yin, L. An Automatic Clustering Algorithm Using NSGA-II with Gene Rearrangement. In Proceedings of the 10th International Conference on Intelligent Systems (IS), Varna, Bulgaria, 28–30 August 2020; pp. 503–509.
237. Qu, H.; Yin, L.; Tang, X. An automatic clustering method using multi-objective genetic algorithm with gene rearrangement and cluster merging. *Appl. Soft Comput.* **2021**, *99*, 106929. [CrossRef]
238. Lin, H.J.; Yang, F.W.; Kao, Y.T. An efficient GA-based clustering technique. *J. Appl. Sci. Eng.* **2005**, *8*, 113–122.
239. Artificial Data Sets. Available online: https://research.manchester.ac.uk/en/publications/an-evolutionary-approach-to-multiobjective-clustering (accessed on 23 September 2020).
240. Moore, M.; Narayanan, A. *Quantum-Inspired Computing*; University of Exeter: Exeter, UK, 1995.
241. Han, K.H.; Kim, J.H. Genetic quantum algorithm and its application to combinatorial optimization problem. In Proceedings of the Congress on Evolutionary Computation, CEC00 (Cat. No. 00TH8512), La Jolla, CA, USA, 16–19 July 2000; IEEE: Piscataway, NJ, USA, 2000; Volume 2, pp. 1354–1360.
242. Han, K.H.; Kim, J.H. Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. *IEEE Trans. Evol. Comput.* **2002**, *6*, 580–593. [CrossRef]
243. Wang, Y.; Feng, X.Y.; Huang, Y.; Pu, D.B.; Zhou, W.; Liang, Y.C.; Zhou, C.G. A novel quantum swarm evolutionary algorithm and its applications. *Neurocomputing* **2007**, *70*, 633–640. [CrossRef]
244. Zouache, D.; Nouioua, F.; Moussaoui, A. Quantum Inspired Firefly Algorithm with Particle Swarm Optimization for Discrete Optimization Problems. *Soft Comput.* **2015**, *20*, 2781–2799. [CrossRef]
245. Moore, P.; Venayagamoorthy, G.K. Evolving combinational logic circuits using a hybrid quantum evolution and particle swarm inspired algorithm. In Proceedings of the NASA/DoD Conference on Evolvable Hardware (EH'05), Washington, DC, USA, 29 June–1 July 2005; pp. 97–102.
246. Ramdane, C.; Meshoul, S.; Batouche, M.; Kholladi, M.K. A quantum evolutionary algorithm for data clustering. *IJDMMM* **2010**, *2*, 369–387. [CrossRef]
247. Maulik, U.; Bandyopadhyay, S. Genetic algorithm-based clustering technique. *Pattern Recognit.* **2000**, *33*, 1455–1465. [CrossRef]
248. Zhou, W.; Zhou, C.; Huang, Y.; Wang, Y. Analysis of gene expression data: Application of quantum-inspired evolutionary algorithm to minimum sum-of-squares clustering. In *Proceedings of the International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 383–391.

249. Dey, S.; Bhattacharyya, S.; Maulik, U. Quantum Inspired Automatic Clustering for Multi-level Image Thresholding. In Proceedings of the International Conference on Computational Intelligence and Communication Networks, Bhopal, India, 14–16 November 2014; pp. 247–251.

250. Dey, S.; Bhattacharyya, S.; Snasel, V.; Dey, A.; Sarkar, S. PSO and DE based novel quantum inspired automatic clustering techniques. In Proceedings of the 3rd International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata, India, 3–5 November 2017; pp. 285–290.

251. Dey, A.; Dey, S.; Bhattacharyya, S.; Snasel, V.; Hassanien, A.E. *Simulated Annealing Based Quantum Inspired Automatic Clustering Technique*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 73–81.

252. Dey, A.; Bhattacharyya, S.; Dey, S.; Snasel, V.; Hassanien, A.E. 7. Quantum inspired simulated annealing technique for automatic clustering. In *Intelligent Multimedia Data Analysis*; Bhattacharyya, S., Pan, I., Das, A., Gupta, S., Eds.; De Gruyter: Berlin, Germany, 2019; pp. 145–166.

253. Dey, S.; Bhattacharyya, S.; Maulik, U. Quantum-inspired automatic clustering technique using ant colony optimization algorithm. In *Quantum-Inspired Intelligent Systems for Multimedia Data Analysis*; IGI Global: Hershey, PA, USA, 2018; pp. 27–54.

254. Flury, B. *A First Course in Multivariate Statistics*; Springer: Berlin/Heidelberg, Germany, 2013.

255. Bhattacharyya, S.; Snasel, V.; Dey, A.; Dey, S.; Konar, D. Quantum Spider Monkey Optimization (QSMO) Algorithm for Automatic Gray-Scale Image Clustering. In Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI), Bangalore, India, 19–22 September 2018; pp. 1869–1874.

256. Dey, A.; Dey, S.; Bhattacharyya, S.; Platos, J.; Snasel, V. Novel quantum inspired approaches for automatic clustering of gray level images using Particle Swarm Optimization, Spider Monkey Optimization and Ageist Spider Monkey Optimization algorithms. *Appl. Soft Comput.* **2020**, *88*, 106040. [CrossRef]

257. Dey, A.; Bhattacharyya, S.; Dey, S.; Platos, J.; Snasel, V. Quantum-Inspired Bat Optimization Algorithm for Automatic Clustering of Grayscale Images. In *Recent Trends in Signal and Image Processing*; Springer: Singapore, 2019; pp. 89–101.

258. Dey, A.; Dey, S.; Bhattacharyya, S.; Platos, J. 5. Quantum Inspired Automatic Clustering Algorithms: A Comparative Study of Genetic Algorithm and Bat Algorithm. In *Quantum Machine Learning*; Bhattacharyya, S., Pan, I., Mani, A., De, S., Behrman, E., Chakraborti, S., Eds.; De Gruyter: Berlin, Germany, 2020; pp. 89–114.

259. Dey, A.; Dey, S.; Bhattacharyya, S.; Platos, J.; Snasel, V. Quantum Inspired Meta-Heuristic Approaches for Automatic Clustering of Colour Images. *Int. J. Intell. Syst.* **2021**, *36*, 4852–4901. [CrossRef]

260. Askarzadeh, A. A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm. *Comput. Struct.* **2016**, *169*, 1–12. [CrossRef]

261. Shekhawat, S.; Saxena, A. Development and applications of an intelligent crow search algorithm based on opposition based learning. *ISA Trans* **2020**, *99*, 210–230. [CrossRef] [PubMed]

262. Dutta, T.; Bhattacharyya, S.; Mukhopadhyay, S. Automatic Clustering of Hyperspectral Images Using Qutrit Exponential Decomposition Particle Swarm Optimization. In Proceedings of the International India Geoscience and Remote Sensing Symposium (InGARSS), Ahmedabad, India, 6–10 December 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 289–292.

263. Xie, F.; Li, F.; Lei, C.; Yang, J.; Zhang, Y. Unsupervised band selection based on artificial bee colony algorithm for hyperspectral image classification. *Appl. Soft Comput.* **2019**, *75*, 428–440. [CrossRef]

264. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95—International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; pp. 1942–1948.

265. Hore, A.; Ziou, D. Image quality metrics: PSNR vs. SSIM. In Proceedings of the 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 2366–2369.

266. Fletcher, S.; Islam, M.Z. Comparing sets of patterns with the Jaccard index. *Australas. J. Inf. Syst.* **2018**, *22*, 1–17. [CrossRef]

267. Dice, L.R. Measures of the amount of ecologic association between species. *Ecology* **1945**, *26*, 297–302. [CrossRef]

268. Hyperspectral Remote Sensing Scenes—Grupo de Inteligencia Computacional (GIC). Available online: http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes (accessed on 7 October 2019).

269. Dey, A.; Bhattacharyya, S.; Dey, S.; Platos, J.; Snasel, V. Quantum Inspired Manta Ray Foraging. Optimization Algorithm for Automatic Clustering of Colour Images. In *Quantum Machine Intelligence*; CRC Press: Boca Raton, FL, USA, 2022; pp. 95–116.

270. Zhao, W.; Zhang, z.; Wang, L. Manta ray foraging optimization: An effective bio-inspired optimizer for engineering applications. *Eng. Appl. Artif. Intell.* **2020**, *87*, 103300. [CrossRef]

271. Dey, A.; Bhattacharyya, S.; Dey, S.; Platos, J.; Snasel, V. Automatic clustering of colour images using quantum inspired meta-heuristic algorithms. *Appl. Intell.* **2022**, 1–23. [CrossRef]

272. Xu, Y.; Fan, P.; Yuan, L. A Simple and Efficient Artificial Bee Colony Algorithm. *Math. Probl. Eng.* **2013**, *2013*, 9. [CrossRef]

273. Biedrzycki, R. On equivalence of algorithm's implementations: The CMA-ES algorithm and its five implementations. In Proceedings of the Genetic and Evolutionary Computation Conference Companion, Prague, Czech Republic, 13–17 July 2019; pp. 247–248.

274. Berkeley Images. Available online: www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/BSDS300/html/dataset/images.html (accessed on 1 May 2017).

275. Real Life Images. Available online: www.hlevkin.com/06testimages.htm (accessed on 1 February 2018).

276. Li, Y.; Feng, S.; Zhang, X.; Jiao, L. SAR image segmentation based on quantum-inspired multiobjective evolutionary clustering algorithm. *Inf. Process. Lett.* **2014**, *114*, 287–293. [CrossRef]

277. Li, G.; Wang, W.; Zhang, W.; You, W.; Wu, F.; Tu, H. Handling multimodal multi-objective problems through self-organizing quantum-inspired particle swarm optimization. *Inf. Sci.* **2021**, *577*, 510–540. [CrossRef]

278. Dey, S.; Bhattacharyya, S.; Maulik, U. Chapter 6—Quantum-inspired multi-objective simulated annealing for bilevel image thresholding. In *Quantum Inspired Computational Intelligence*; Bhattacharyya, S., Maulik, U., Dutta, P., Eds.; Morgan Kaufmann: Boston, MA, USA, 2017; pp. 207–232.

279. Yan, L.; Chen, H.; Ji, W.; Lu, Y.; Li, J. Optimal VSM Model and Multi-Object Quantum-Inspired Genetic Algorithm for Web Information Retrieval. In Proceedings of the International Symposium on Computer Network and Multimedia Technology, Wuhan, China, 18–20 December 2009; pp. 1–4.

280. Kumar, D.; Chahar, V.; Kumari, R. Automatic Clustering using Quantum based Multi-objective Emperor Penguin Optimizer and its Applications to Image Segmentation. *Mod. Phys. Lett. A* **2019**, *34*, 1950193. [CrossRef]

281. Liu, R.; Wang, X.; Yangyang, L.; Zhang, X. Multi-objective Invasive Weed Optimization algorithm for clustering. In Proceedings of the Congress on Evolutionary Computation, Brisbane, QLD, Australia, 10–15 June 2012; pp. 1–8.

282. Dey, A.; Bhattacharyya, S.; Dey, S.; Platos, J.; Snasel, V. Quantum-Inspired Multi-Objective NSGA-II Algorithm for Automatic Clustering of Gray Scale Images (QIMONSGA-II). In *Quantum Machine Intelligence*; CRC Press: Boca Raton, FL, USA, 2022; pp. 207–230.

283. Srinivas, N.; Deb, K. Muiltiobjective Optimization Using Non dominated Sorting in Genetic Algorithms. *Evol. Comput.* **1994**, *2*, 221–248. [CrossRef]

284. Mukhopadhyay, A.; Bandyopadhyay, S.; Maulik, U. Clustering using Multi-objective Genetic Algorithm and its Application to Image Segmentation. In Proceedings of the 2006 IEEE International Conference on Systems, Man and Cybernetics, Taipei, Taiwan, 8–11 October 2006; IEEE: Piscataway, NJ, USA, 2006; Volume 3, pp. 2678–2683.