Quantum Machine Learning for Video Compression: An Optimal Video Frames Compression Model using Qutrits Quantum Genetic Algorithm for Video multicast over the Internet

Oday Ali Hassen[1,*] , Huda Lafta Majeed[2] , Mohammed Abdulhasan Hussein[1], Saad M. Darwish[3], Omar Al-Boridi[4]

[1]Ministry of Education, Wasit Education Directorate, Kut 52001, Iraq
[2]Computer Science and Information Technology, University of Wasit, Al Kut 52001, Iraq
[3]Department of Information Technology, Institute of Graduate Studies and Research, Alexandria University, Egypt
[4]School of Engineering - RMIT University, Melbourne, Australia

Emails: odayali@uowasit.edu.iq; hulafta@uowasit.edu.iq; mohammedabdalhassan7@gmail.com; saad.darwish@alexu.edu.eg; omar.alboridi@rmit.edu.au

Abstract

The transmission of video is greatly aided by video compression. Redundancy (spatial, temporal, statistical, and psycho-visual) within and between video frames is something that video compression approaches aim to get rid of. The degree to which similarity-based redundancy exists between consecutive frames, however, is a function of how often the frames are sampled and how the objects in the scene are moving. Existing neural network-based video compression approaches rely on a static codebook to perform compression, which prevents them from adapting to new video's data. In order to create an optimal codebook for vector quantization, which is then employed as an activation function inside a neural network's hidden layer, this research offers a modified video compression method based on a Qutrits based Quantum Genetic Algorithm (QQGA). Using quantum parallelization and entanglement of the quantum state, QQGA is capable of solving the same set of problems as a traditional genetic algorithm while considerably accelerating the evolutionary process. The technique is built on the concept of utilizing Qutrits (three-level quantum system) to represent population individuals. The evolution operator, which is responsible for the updates to the quantum system state, has been constructed using a straightforward approach that does not need a lookup table. Compared to qubit, qudit provides a larger state space to store and process information, and thus can enhance the algorithm's efficiency. To create the context-based initial codebook, the background subtraction algorithm is used to extract moving objects from frames. Moreover, important wavelet coefficients are compressed losslessly using Differential Pulse Code Modulation (DPCM), whereas low energy coefficients are compressed lossy using Learning Vector Quantization neural networks (LVQ). To obtain a high compression ratio, Run-Length Encoding is then used to encode the quantized coefficients. In comparison to the conventional evolutionary algorithm-based video compression method, experiments have shown that the quantum-inspired system may achieve a greater compression ratio with acceptable efficiency as evaluated by PSNR.

Keywords: Video compression; Adaptive coding; Optimization; Quantum machine learning; Context-based compression, Quantum genetic algorithm

1. Introduction

Over the last several decades, digital information has been more in demand due to the proliferation of multimedia technologies. Due to this terrible need, existing technology is unable to effectively manage the enormous volume of data. This issue is resolved by video compression, which reduces the redundancies that are present in it [1]. The primary objective of video compression is to keep video quality constant at a lower bit rate for transmission and

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

storage. Lossless and lossy compression are the two kinds of compression methods [2]. When compressing or encoding video, lossless compression ensures that the original can be properly restored from the compressed or encoded version. These are also known as noiseless compression methods since they do not introduce any additional noise to the original frame, in contrast to lossy compression methods, where the recovered video does not have the same quality as the original. Due to its high compression rate, this kind has seen extensive application [3–5]. Figure 1 illustrates the video compression' main steps [1].

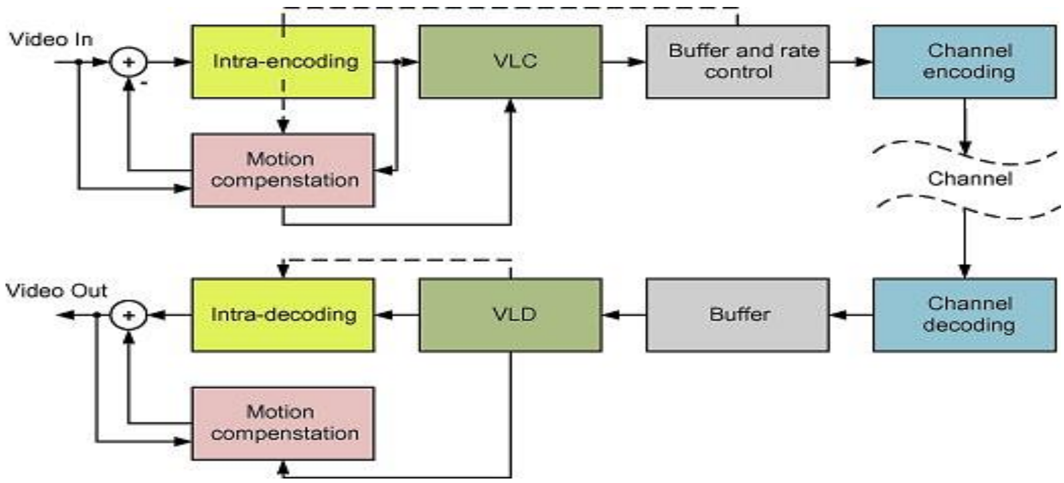

Figure 1. Video compression procedure using variable length coding (VLC) and decoding (VLD).

Unfortunately, there are still several issues or obstacles that prevent video compression from gaining widespread use. The issue at hand is how to find a trade-off between compression level and video quality as measured by peak signal-to-noise ratio (PSNR) is how to find a trade-off between compression level and video quality as measured by peak signal-to-noise ratio (PSNR). The video quality after reconstruction decreases if the compression ratio is large. The algorithmic complexity of video compression should be addressed as well. Another difficulty lies in the fact that the foundations for creating perception-aware coding techniques—knowledge of videos frame colour spaces and their perceptual mechanisms—are still being developed. Furthermore, researchers have often been impeded in their search for broadly applicable perceptual compression strategies by users' varying expectations of quality across applications and contexts. However, perceptual video compression has significant promise as a solution to ease the management of multimedia material because of its efficacy in reducing data rates [6-9] [10].

In an effort to address these issues, numerous new methods have been proposed recently. Spatial, temporal, statistical, and psycho-visual redundancy make up the taxonomy of these methods [10]. Elements, such as pixels in a video frame, may be replicated inside a structure due to spatial redundancies (intra-coding). Compression is carried out by making use of spatial redundancy; any unresolved detail is averaged out. When two video frames share the same values at the same locations, this is called inter-coding, or temporal redundancy. The compressed coefficients are encoded using statistical redundancies. In the last step of video compression, variable-length coding is utilized to make use of binary codes to take advantage of these statistical redundancies and increase compression rate. The term "psycho-visual redundancy" describes a redundancy that corresponds to the varied sensitivities of all frame signals seen by human eyes. As a result, it could be acceptable if we ignored certain minor details when processing visual data. By taking advantage of spatial redundancy, higher frequencies may be effectively removed or reduced without compromising the perceived quality [1][3][7].

Due to its ease of use and efficiency, vector quantization (VQ) has become a popular method for compressing video. Encoding, codebook generation, and decoding are all steps in the VQ process [11]. In VQ, we encode a number of samples at a time (a vector). The easiest thing to do is to compare the vector against every potential encoding in a codebook and then transmit the index of the code word. Designing vector quantization codebooks can be done randomly, which works surprisingly well, or by K-means or other machine learning techniques. The Linde-Buzo-Gary (LBG) algorithm is the most popular choice for VQ in the academic literature. However, LBG suffers from a local optimum issue, and each code word in the codebook has very little utility. The codebook ensures minimal distortion at the local level but not at the global level, which is the source of the local optimality issue. Since each data point is represented by its index to the nearest centroid, the inaccuracy is less for more often occurring data and larger for less frequently happening data. Therefore, lossy data compression is a good fit for VQ [11-14].

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

The vast majority of published video encoding techniques make use of neural networks, which include both a spatial component that encodes intra-frame visual patterns and a reconstructive component that aggregates information to forecast de-details. Several of these techniques deepen the spatial component to take advantage of spatial redundancies during reconstruction of high-frequency structures [15-17]. The performance of the neural video compression approach using the predictive VQ algorithm relies on accurate recognition of key frames. When it comes to explicitly identifying relationships, neural networks are somewhat of a "black box" with limited capabilities. Recently, evolutionary optimization approaches have been used to improve NN learning and construct intelligent vector quantization. These techniques include genetic algorithms (GAs) and swarm intelligence [18] [19]. Many complex optimization problems have been solved successfully with GA thanks to its useful solutions for NN learning and optimal codebook design [20] [21]. The process of creating a codebook may be thought of as a search problem, with the desired outcome being the identification of the best possible codebook to use for compressing videos. The main disadvantage of GA is the convergence speed which is in turn is enhanced by the quantum adaptation of the algorithm.

Quantum machine learning is an advancement on and sometimes acceleration of conventional machine learning that makes use of quantum methods to tackle machine learning problems [22]. In order to use such algorithms on a quantum computer, one must first encode the provided classical data set into the quantum computer's language. Measurements of the quantum system are then used to deduce the outcome of the quantum computation after the use of quantum information processing algorithms [22] [23]. Four different approaches to combine the disciplines of quantum computing and machine learning (see Figure 2). Quantum-Inspired optimization algorithms exploit some of the advantages of quantum computing on classical hardware, providing a speedup over traditional approaches.

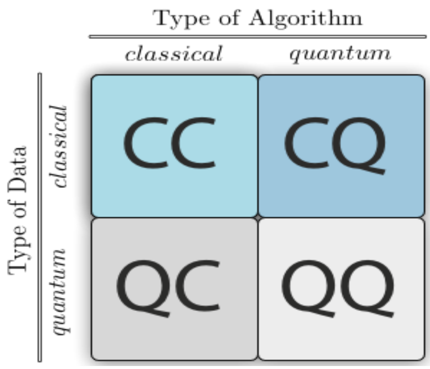

Figure 2. Quantum machine learning types.

The quantum genetic algorithm (QGA) is a new evolutionary algorithm based on a combination of quantum computation and conventional genetic algorithm technology [24]. This algorithm can be applied to the same set of problems that the conventional genetic algorithm is used for, but it significantly accelerates the evolutionary process through quantum parallelization and entanglement of the quantum state. The probabilistic mechanism of quantum computations combined with the evolutionary algorithm provides a global search for a solution with rapid convergence and a small population. These algorithms have demonstrated their effectiveness for solving combinatorial and functional optimization problems, video processing, and many others [24] [25]. Recently, ternary quantum logic has been employed, which helps in reaching better optimum values faster than the qubit-based existing metaheuristics. A Qutrit (or quantum trit) is a unit of quantum information that is realized by a three-level quantum system that may be in a superposition of three mutually orthogonal quantum states [26][27]. Qudits may minimize circuit complexity, and enhance algorithm efficiency by storing and processing information in a greater state space than qubits.

The suggested video compression scheme's novelty is that it is based on removing several forms of redundancies in a single package. The system deals with spatial redundancy in the frame by removing the duplicate in the high-frequency coefficients of the discrete wavelet transform (DWT) by adapting vector quantization based neural network, and by using differential pulse code modulation (DPCM) to remove the redundancy in the low-frequency (high energy) coefficients. By using the background subtraction technique to extract motion objects inside frames, the system is able to regulate the enter-frame temporal redundancy and provide a reduced initial codebook. The method uses run-length encoding to reduce statistical redundancy and improve compression. By using Qutrits Quantum Genetic Algorithm (QQGA) with a fitness function based on the Euclidean distance between the original codebook and each frame in the video, we can generate the ideal codebook for vector quantization, which is crucial to the system's overall performance. This is the first time that the QQGA has been utilized for building optimal codebook in the video compression domain.

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

The remaining sections of the paper are structured as follows: Recent studies in this field are discussed in Section 2. Section 3 provides a comprehensive overview of the proposed system. Section 4 presents the results and provides discussion. In Section 5, the research results are summarized.

2. State of the Art

Research in the field of video compression has gained a lot of attention in recent years. Mainly because it is difficult to successfully achieve a high compression ratio and quality after decoding without reconstructed video degradation [1][3][10][28][29]. In [30], the authors presented an algorithm that handles the low compression ratio of the neural video codec (NVC) because of its constrained contexts. The authors argued that expanding NVC's temporal and spatial context diversity is the best way to do so. To begin, the model is instructed to learn hierarchical quality patterns across frames, which improve the quality of long-term yet high-quality temporal contexts. Additionally, the authors presented group-based offset diversity, where cross-group interaction is advocated for improved context mining, to fully realize the promise of an optical flow-based coding system. Encoding the latent representation concurrently increases the diversity of spatial contexts, and this research proposes a quad tree-based partition to achieve this. In comparison to prior NVC methods, experimental results indicated that their codec reduced bitrate by 23.5%.

In order to more accurately depict spatiotemporal information, the authors in [31] used a novel convolutional architecture for video representation and a training technique that could optimize both frame rate and distortion simultaneously. By concurrently learning all network and quantization parameters from end to end, they eliminate the need for time-consuming post-training methods common in prior efforts. New state-of-the-art results for video compression using a network video recorder (NVR) are achieved. Learned transforms and entropy coding transforms were suggested in [32] as potential (non-)linear drop-in replacements or improvements for linear transforms in already-existing codecs. Multi-rate capabilities of these transforms make it possible for a single model to function throughout the full rate-distortion space. To prove our framework's worth, they compressed intra-frame AV1 block prediction residuals using the DCT plus learned quantization matrices and adaptive entropy coding. In comparison to more sophisticated nonlinear transforms, they found significant gains in BD-rate and perceptual quality at a fraction of the computing cost.

In [33], the authors introduced a video compression technique called sandwiched video compression, which uses neural networks to compress videos by encasing a regular video codec in a compressed format. A common video codec sits in the middle of a neural pre-processor and a post-processor in the sandwich design. With the aim of vastly outperforming the baseline codec across a wide range of compression use cases, the networks are trained simultaneously to optimize a rate-distortion loss function. A distinguishable proxy for the baseline video codec that includes temporal processing with motion correction, inter/intra mode choices, and in-loop filtering is necessary for end-to-end training in this context. Furthermore, differentiable approximations to critical video codec components were provided, and it was shown that the neural codes of the sandwich yield much higher rate-distortion performance compared to compressing the original frames of the input video in two crucial cases. By using low-resolution HEVC to transmit high-resolution video, the sandwich system improves compression efficiency by 6.5 dB compared to baseline HEVC.

The key to video compression, as opposed to image compression, is to effectively exploit the temporal context in order to eliminate inter frame redundancy. The coding performance can't be improved any more than it already is since existing learned video compression approaches depend on either image-oriented codecs or short-term temporal correlations. In order to boost the efficiency of learned video compression, the research described in [34] proposes a new temporal context-based video compression network (TCVC-Net). In order to aggregate long-term temporal context and acquire an accurate temporal reference for motion-compensated prediction, a global temporal reference aggregation (GTRA) module is provided. Additionally, a temporal conditional codec (TCC) is developed to maintain structural and detailed information by utilizing the multi-frequency components in temporal context, allowing for effective compression of the motion vector and residue. The suggested TCVC-Net achieved better PSNR and MS-SSIM outcomes in experiments than existing public state-of-the-art approaches.

In order to improve the efficiency of motion compensation, the authors in [35] created a multiple-hypothesis-based method for the learned video compression. This method fuses together different hypotheses in a timely manner. In particular, we suggest the use of the multiple hypotheses module, which generates multiple movements and warped features for mining enough temporal information. Using a channel-wise squeeze-and-excitation layer and a multi-scale network, the hypotheses attention module is implemented to make better use of these speculations. To further strengthen the contexts with potent temporal priors, the context combination is used to fuse the weighted hypotheses. By combining weighted and warped features based on the valid contexts, compression efficiency is improved. Extensive experiments validated the effectiveness of the suggested strategy in enhancing the rate-distortion performance of learned video compression. Over 13% bit rate reductions can be accomplished on

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

average in terms of PSNR and MS-SSIM compared to the state-of-the-art technology for end-to-end video compression.

In [36], the authors suggested a semantic-aware video compression (SAC) framework for transmitting video frames from automotive cameras to the processing unit(s), where the data would be used for perception tasks like object detection, semantic segmentation, etc. The experimental results demonstrate that, for a given overall compression ratio, their SAC technique retains equivalent or superior image quality when evaluated using both conventional and semantic-aware measures. Despite years of research, learning-based video compression still has problems adjusting to different motion patterns and entropy models. In [37], the authors suggested a multi-mode video compression (MMVC) framework based on block-wise ensemble deep video compression. MMVC chooses the best mode for feature domain prediction and can adapt to various motion patterns. In order to make temporal predictions using spatial block-based representations, the authors partitioned the feature space into blocks. In order to maximize compression efficiency, entropy coding takes into account both dense and sparse blocks of post-quantization residual data. As such, their approach employs a dual-mode entropy coding technique directed by a binary density map, which provides a substantial rate decrease that more than compensates for the additional cost of sending the binary selection map. Their technique achieves superior or competitive results in terms of PSNR and MS-SSIM when compared to state-of-the-art video compression algorithms and popular codecs.

In order to increase the video signal's data rate and efficiency, the method proposed in [38] employs the removing temporal redundancy (RTR) approach to get rid of the highly redundant information included in each video signal block. The PSNR values of this method are compared to those of the JPEG video compression standard. The method was discovered to be appropriate for wireless networks and embedded devices due to its low memory and processing demands. Results from a comprehensive battery of tests show that the RTR compression method is capable of a compression ratio of 22.71 and a decrease in bit rate of 95% with no impact on signal quality. The self-similarity notion is used in fractal video compression, which means that the fractal image has self-similarity inside itself that is explained and expressed by change. Since fractal video encoding requires more processing power, researchers have come up with a variety of techniques to lower this overhead. Using the diamond-search-pattern-block-matching motion estimation method and the hash-based fractal video compression technology to speed up the decoding process is the primary goal of the work suggested in [39]. In [40], each pair of odd and even frames was compressed and decompressed using an adaptive FIS (Fuzzy Inference System). At first, all feature pairs of each odd-even frame pair were used to train adaptive FIS. The adaptive FIS is a codec used for video compression and decompression. Mean, standard deviation, mean absolute deviation, and mean standard deviation are some of the measures used. The average DCT (Discrete Cosine Transform) components of all video frames are used as a quality metric in this research.

By reusing pixels from previously decoded frames in accordance with motion and residual correction, video compression may increase coding efficiency. Hierarchical redundancy in video frames is specified on two levels by the authors in [41]: In the first order, motion and residual compensation successfully capture redundancy in pixel space, i.e., similarities in pixel values across neighbouring frames; in the second order, smooth motion in natural videos causes redundancy in motion and residual maps. Second-order redundancy is captured in neural video codecs using predictors, a challenge not addressed in the previous research on neural video coding. They implemented residual predictors that learn to extrapolate from previously decoded data as well as general motion predictors. These predictors are lightweight and can be used to enhance the performance of most neural video codecs in terms of rate distortion. Their results on the UVG dataset indicated that combining our predictors with a popular neural video codec reduced bitrate by 38% in the RGB colour space and 34% in the YUV420 colour space.

In terms of subjective quality and rate distortion, neural video compression algorithms are almost on par with hand-crafted codecs. On the other hand, many neural codecs are rigid black boxes that don't let users tweak the bitrate or quality of the reconstructed signal. Combining concepts from variable-bitrate codecs and region-of-interest-based coding, the authors in [42] introduced a versatile neural video codec. They were able to acquire granular control over the per-frame bitrate and the reconstruction quality in the ROI by conditioning the model on a global rate-distortion trade-off parameter and a region-of-interest (ROI) mask. The resultant codec offers realistic use cases, including encoding under bitrate limits with constant ROI quality at the expense of just a small amount of performance loss due to rate distortion. Their codec significantly outperforms non-ROI codecs in the area of interest, with BD-rate reductions over 60% in certain circumstances, when the sequence comprises complicated motion, such as sequences with camera panning or sports videos.

In most current neural video codecs, only motion coding supplies motion vectors for frame coding; this one-way flow of information is called "unidirectional information flow. In [43], the authors propose that the synergy between motion coding and frame coding may be attained through information exchanges. Together, motion coding and frame coding benefit from motion information propagation's successful introduction of bidirectional information exchanges. The high-dimensional motion feature from the motion decoder acts as motion guidance to

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

reduce alignment errors while producing the temporal contexts for frame coding. Meanwhile, the feature from context generation will be conveyed as a motion condition for encoding the next motion latent, in addition to aiding frame coding at the present time step. This feedback loop is crucial for feature propagation in motion coding, which improves the system's ability to take advantage of temporal correlation across large distances. Experiments validated that their solution is superior to the prior-generation SOTA neural video codec in terms of bit rate savings (12.9%).

In [44], the authors presented a number of designs that, when applied to high-definition video, provide state-of-the-art compression performance, and they examined the benefits and drawbacks of each. They suggested (i) enhanced temporal autoregressive transformations, (ii) enhanced entropy models with structural and temporal dependencies, and (iii) variable bitrate versions of our techniques. Their work supports the argument that the generative modelling perspective may contribute to the development of neural video coding, given that its enhancements are broadly applicable to pre-existing models.

Based on a compressive sensing (CS) framework, the work in [45] is devoted to video coding. Assumptions are made in computer science that if a video sequence is sparse in any transform domain, it may be reconstructed using fewer samples (also known as measurements) than are required by the Nyquist-Shannon theorem. Here, the quality of the codec is determined by the methods used to acquire (or sense), compress, and rebuild the video from the decoded measurements. In this case, a quicker encoding time may be possible using this codec instead of the more common block-based intra-frame encoding methods as Motion JPEG (MJPEG), H.264/AVC, or H.265/HEVC. However, current CS-based video codecs perform poorly in terms of rate distortion compared to legacy codecs, rendering them worthless in real-world applications. The authors introduced a CS-JPEG video codec. Their results demonstrate that the proposed CS-JPEG encoding is 2.2 times faster than MJPEG, 1.9 times faster than H.264/AVC, and 30.5 times faster than H.265/HEVC when compared to their optimized software implementations, and it improves the peak SNR by 2.33 dB, 0.79 dB, and 1.45 dB, respectively. For video applications with severe constraints on processing resources or the battery life of an upstream streaming device, it may be more appealing.

Encoding videos as neural networks is a recently proposed approach that allows for new forms of video processing. For the purpose of video compression, however, conventional methods continue to perform better than neural video representation (NVR) approaches. Existing NVR methods use architectures that either don't get a good representation of temporal and spatial information in a small space or try to minimize rate and distortion by overfitting the video network and then compressing the model to make up for the performance gap. There is still space to make video compression more efficient and useful in real-world applications, despite the fact that it has been researched for over ten years. Studies in the past, as summarized in the aforementioned review, focused mostly on (1) using the video data to employ a variety of redundancies for compression (both intra- and inter-frame). (2) The problems with codebook building for vector quantization compression techniques (often constructed at random) are ignored. (3)Most neuro-coding methods need extensive training since they depend on weight matrix adjustment to accomplish compression. However, as far as we're aware, no serious effort has been put into developing a new optimal codebook and increasing its efficiency for vector quantization –based compression techniques. This paper utilizes a quantum genetic algorithm based on a three-level quantum (Qudit) system in order to accelerate the evolutionary process to deal with the third issue. Qudit is a structure with several, or more than two, states that can also be used to encode a chromosome. A qudit is basically a unit of quantum information, which may be in any of the *n* states or in any superposition of those.

3. Methodology

This research suggests a novel approach that merges intra-frame and inter-frame video coding into a single framework with the objective of reducing spatial, temporal, and statistical redundancy. Intra-frame coding is accomplished by integrating information from wavelet transformation and quantization. The wavelet transform de-correlates the pixels of the input frame. This process converts the pixels into a set of coefficients that can be encoded more efficiently compared to the original pixel values. The quantization information is derived from differential pulse code modulation (DPCM), which serves as the fundamental component of nearly all lossless compression algorithms. The success of the model as a whole is determined by how well the optimal codebook for vector quantization is built. The suggested model makes use of the Qutrits-based Quantum Genetic Algorithm (QQGA) for this purpose, with the fitness function being the Euclidean distance between the initial codebook and each frame of the video. Utilizing QQGA seems to increase the population's size with the aim of improving the optimal feature selection process that is required for building codebook.

The statistical success of promoting advantageous building codebooks (blocks) in conventional genetic algorithms may be attributed to their capacity to generate children who possess high fitness, enabling their survival and further propagation of the beneficial building block. Nevertheless, when a novel constituent emerges within the

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

population, it is afforded just one opportunity to prove its worth. The integration of superimposed individuals within the QQGA framework serves to mitigate a significant portion of the inherent stochasticity associated with traditional genetic algorithms (GAs). Therefore, the QQGA is expected to have a much higher statistical advantage when using effective building blocks. Consequently, this phenomenon is expected to result in an accelerated proliferation of high-quality building blocks. In our case, Qudit, the unit of quantum information described by a superposition of *d* states where the number of states is an integer greater than two, provides a larger state space to store and process information, which can enhance the algorithm's efficiency. Figure 3 depicts the primary components of the compression and decompression phases, respectively, as well as their interconnections.

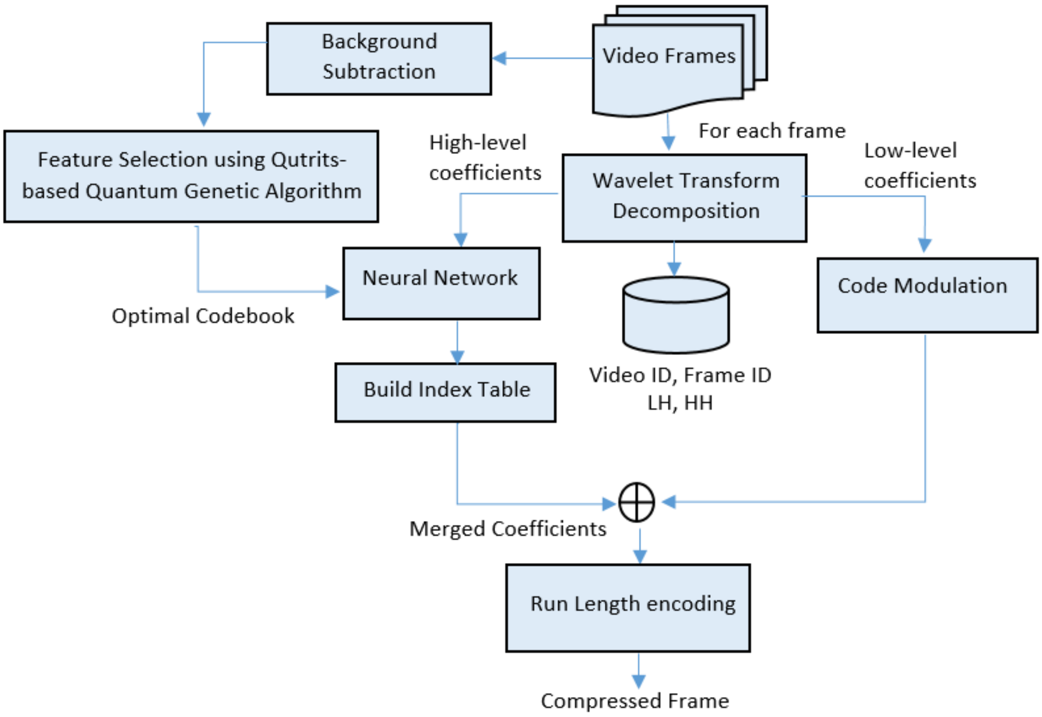

Figure 3. Flow diagram of proposed quantum machine learning video compression system: compression phase.

Step 1: ***Initial Codebook Generation***

At this stage, an off-line codebook is constructed for each video by isolating the frames' moving parts (foreground) and their associated static backdrop parts (background); they are then represented by code words. Background subtraction is used to separate out the moving items. Detecting moving objects in videos captured by stationary cameras is a common task, and one for which the background subtraction technique is often used [46].

$$\left|\boldsymbol{f}_{\boldsymbol{c}} - \boldsymbol{f}_{\boldsymbol{p}}\right| < \boldsymbol{\delta} \qquad \textbf{(1)}$$

f_c is the current frame, f_p is the previous frame, and δ is the threshold; the precision of this method depends on how quickly movement occurs in the scene. A greater threshold could be necessary for faster movement. To begin, a dataset $P = \{p_i | i = 1,2, \dots, N_p\}$ is created by erasing the background, and then the LBG method is employed to construct an initial codebook. N_p is the total number of moving objects plus background. For more information, please refer to [47][48].

Step 2: ***Codebook Optimization***

After an initial codebook has been established, the code words inside it may be fine-tuned in accordance with a predefined objective function. To do this, we make use of the Qutrits-based Quantum Genetic Algorithm (QQGA), which is well-suited to the search for optimal solutions to optimization problems with discrete or decidable sets of potential solutions. When used to generate a codebook, QQGA seems to increase the population size in an effort to enhance the optimum feature selection process [26][27]. Qudit, with its greater state space for storing and processing information, may improve the efficiency of algorithms even further than qubit. It is possible to express the qutrit state as a superposition using a linear combination:

$$|\boldsymbol{q}\rangle = \propto |\boldsymbol{0}\rangle + \boldsymbol{\beta}|\boldsymbol{1}\rangle + \boldsymbol{\gamma}|\boldsymbol{2}\rangle, \ \textbf{with constraint} \ \propto^{2} + \boldsymbol{\beta}^{2} + \boldsymbol{\gamma}^{2} = \boldsymbol{1} \qquad \textbf{(2)}$$

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, |1\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, |2\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, |q\rangle = \begin{pmatrix} \propto \\ \beta \\ \gamma \end{pmatrix} \qquad (3)$$

In this case, there are 3^N base states in a system with N qutrits, as opposed to 2^N in binary logic.In many-valued logic, the number of basis states grows exponentially, resulting in improved algorithm speed while maintaining the same level of search accuracy as in binary logic. The number of basis states in a quantum system, the required search accuracy ε, and the search area $[x_{max} - x_{min}]$ all play a role in determining the optimal length of a quantum chromosome; n is a number of quantum system basis states.

$$N = log_n\left(\frac{x_{max}-x_{min}}{\varepsilon} + 1\right) \qquad (4)$$

Accordingly, the length of the quantum chromosome must be at least 21 qubits with n = 2, search accuracy $\varepsilon = 10^{-6}$, and search area [-1, 1]. With n = 3, only 14 qutrits are needed. In this analogy, each codeword is a gene and each codebook is a chromosome. For Qutrit Encoding, a quantum chromosome is a structure made up of N qutrits, which may be represented by a matrix as follows.

$$\begin{bmatrix} \propto_1 & \propto_2 & \propto_1 & \cdots\cdots\cdots\cdots\cdots & \propto_N \\ \beta_1 & \beta_2 & \beta_2 & \cdots\cdots\cdots\cdots\cdots & \beta_N \\ \gamma_1 & \gamma_2 & \gamma_3 & \cdots\cdots\cdots\cdots\cdots & \gamma_N \end{bmatrix} \qquad (5)$$

Here, $\{\propto_i, \beta_i, \gamma_i\}$ represent the state of the ith qutrit, and N qutrits represent a single individual in the population. Given that the qutrits' initial state contains no information about the system's state, initializing the base population by making all probability state amplitudes equal is the simplest approach. This implies that by the time the initialization process is complete, each qutrit is in the state.

$$|q\rangle = \frac{1}{\sqrt{3}}|0\rangle + \frac{1}{\sqrt{3}}|1\rangle + \frac{1}{\sqrt{3}}|2\rangle \qquad (6)$$

Regarding the observation of genes, the final state vector $|\grave{\psi}\rangle$ contains classical information about the solution to the issue and may be recovered by observation at the quantum scale. The classical representation of a quantum chromosome is the Qutrit in one of the basic states, which is derived by quantum observation. Algorithm 1 may be put forward for quantum N-qutrit state chromosome observation [26]. Following execution of the procedure, the quantum chromosome is converted to its classical representation in the ternary numbering system given that the ground state p (0, 1 or 2) with a probability $\propto_i^2, \beta_i^2, and\ \gamma_i^2$ respectively.

$$\begin{bmatrix} \propto_1 & \propto_2 & \propto_1 & \cdots\cdots\cdots\cdots\cdots & \propto_N \\ \beta_1 & \beta_2 & \beta_2 & \cdots\cdots\cdots\cdots\cdots & \beta_N \\ \gamma_1 & \gamma_2 & \gamma_3 & \cdots\cdots\cdots\cdots\cdots & \gamma_N \\ & & \textit{Observation} & & \\ 0 & 2 & 0\ \ 1\ \ 1\ \ 2\ \ 1\ \ 0 & & 0 \end{bmatrix} \qquad (7)$$

Algorithm 1: Qutrit state observation

(1) For i in 1, . . . , N do

(2) r ←random number in range [0, 1]

(3) If r < $[a_i]^2$ then

(4) p ← 0

(5) else if r < $[a_i]^2 + [\beta_i]^2 + [\gamma_i]^2$ then

(6) p ← 1

(7) else

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

(8) $p \leftarrow 2$

(9) **end if**

(10) end for

The algorithm of the quantum gate is the central challenge in the development of any quantum genetic algorithm, since it contains all relevant information about the problem and the technique for solving it. Algorithm 2 presents the pseudocode for the genetic operator updating the qutrit state (the evolutionary quantum gate) due to changes in the probability amplitudes $\propto_i, \beta_i, and\ \gamma_i$. Importantly, the lookup table used by the classic QGA is not required by the utilized quantum gate technique. For example, if the best individual in the classical representation of the population has the value 2 at location i, then all amplitudes are increased by a factor μ of except for γ_i. Its value, which is determined experimentally, is somewhere between [0, 1]. Therefore, only the probability amplitude, which represents the best individual in the population at the previous stage of evolution, would increase. Simultaneously, the amplitudes of the probabilities and will drop. All solutions have the same probability in the initialized population. This indicates a random search is performed as the first step in the QGA procedure. The distribution changes during the course of evolution. In general, the process of local convergence works best when there are more individuals in the area of the best individual, and it is desirable to use an adaptive quantum gate operator μ

Algorithm 2: Update of quantum states

(1) for i **in 1, . . . ,** N **do**

(2) $bestamp \leftarrow i$**th gene of the population's best individual**

(3) $sum \leftarrow 0$

(4) for amp in {0, 1, 2} do

(5) **if** $amp \neq bestamp$ **then**

(6) $q'[amp] = \mu \cdot q$

(7) $sum \leftarrow sum + q'[amp]2$

(8) **end if**

(9) **end for**

(10) $q'[bestamp] \leftarrow \sqrt{1 - sum}$

(11) end for

If a population evolves on the quantum stage, it might eventually converge on a locally optimum solution. When the fitness function has a complicated structure around the global optimum, or when the search area is huge, this topic becomes paramount. In general, using a quantum disaster operation is a good way to escape the local optimum solution. The simplest implementation of the quantum disaster operation is illustrated in Algorithm 3, which involves the initialization of some individuals in the population other than the best with an initial state. This operation "virtually" enlarges the size of the population and increases the area for optimal value search.

Algorithm 3: Quantum disaster

(1) for i **in 1, . . . ,** s **do**

(2) **if the chromosome is not the best then**

(3) **if disaster condition then**

(4) **for** i **in 1, . . . ,** N **do**

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

(5) $\propto_i = \boldsymbol{\beta}_i = \boldsymbol{\gamma}_i = \frac{\mathbf{1}}{\sqrt{\mathbf{3}}}$

(6) end for

(7) end if

(8) end if

(9) end for

By comparing the source vector to all the code vectors in the codebook, the newer generations find the one with the least dissimilarity (distortion). The codebook's layout is crucial to the success of vector quantization (VQ). Contradictions between distortion and complexity, such as the size of the codebook and the dimension of the code vector, affect VQ's performance. To get the optimal centroid of each partition, GA is utilized [12-14]. Here, we turn each frame into a vector and use the Euclidean distance between the original codebook and each frame to determine the fitness function [21]. By producing an ideal codebook that supports expressing the majority of frame data in a correct index, this method improves the efficiency of video compression and reduces the computing cost of VQ coding by taking full use of the robustness of Qutrits based quantum–inspired evolutionary algorithms. In order to utilize this codebook table in the decompression process, it will be saved to a database.

Step 3: Compression

When designing data compression systems, designers must strike a balance between a number of competing considerations, such as the compression ratio, the amount of distortion imposed by lossy compression, and the amount of processing power needed for compression and decompression [30]. Lossless compression using DPCM and lossy compression using an improved LVQ neural network make up the two primary steps of the compression process. On the Haar wavelet domain of each frame, both stages operate. One of the key advantages of Haar wavelets is their ability to efficiently represent signals with sharp transitions. This makes them particularly useful in image compression algorithms, where preserving important details while reducing file size is crucial. By decomposing an image into Haar wavelet coefficients, it becomes possible to selectively discard less significant coefficients, resulting in a compressed representation of the image [49]. Figure 4 visualizes the schematic diagram of Haar wavelet transform (HWT) for 2D video's frame [50].

Stage 1: Lossless Compression

To prevent irretrievable loss of essential details (salient features), lossless compression is applied to each frame's high-energy low-wavelet-frequency coefficients. DPCM is used here as a signal encoder; it builds on the foundation of pulse-code modulation (PCM), but with additional features based on the prediction of the signal samples [51]. PCM is inefficient because it creates many more bits than it needs and uses more bandwidth. Therefore, the DPCM was developed as a solution to the aforementioned issue. DPCM is also made up of sampling, quantization, and coding processes like PCM. The difference between the actual sample and the predicted value is quantized in DPCM, which differs from PCM as illustrated in Fig. 5. As an outcome, the difference between adjacent samples is consist of an average power which is smaller than the average power of the original signal. Differential PCM is so named because of this feature. By using this method, the signal's short-term redundancy (positive correlation between nearby values) is removed, and compression ratios of 2 to 4 can be attained by entropy coding the differences between the two signals. This is because the entropy of the difference signal is much smaller than that of the original discrete signal, which is treated as independent samples.

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

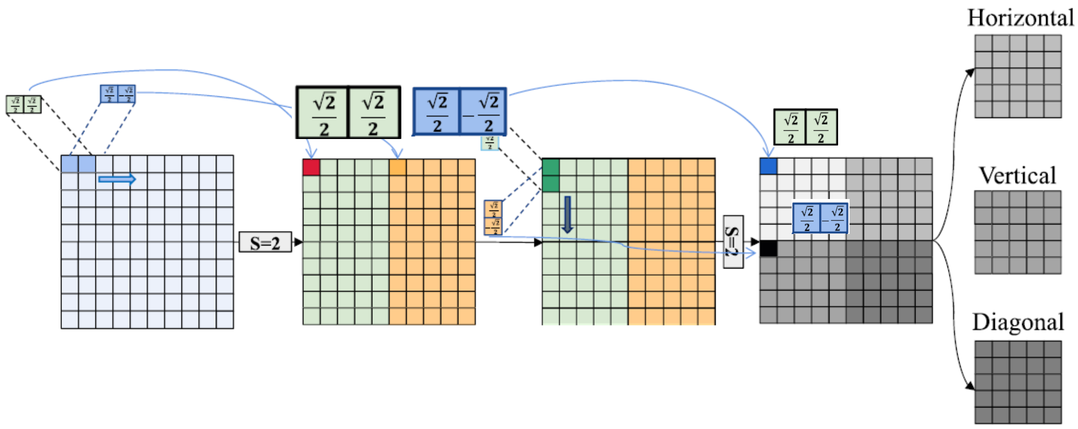

Figure 4. Schematic depiction of the Haar wavelet transform, in which a wavelet transformation of the rows and columns is performed in dimension one.

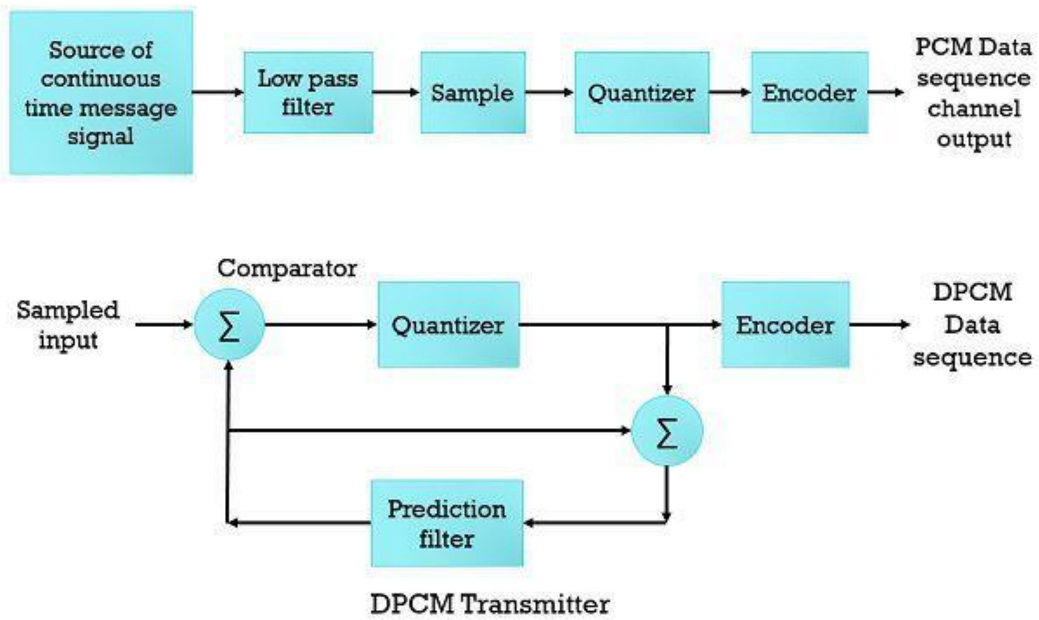

Figure 5. Difference between PCM and DPCM

Stage 2: Lossy Compression

Each frame's high wavelet frequency coefficients that contain little energy (i.e., are not salient features) are lossy compressed in order to get a high compression ratio. An optimized codebook for each video is used by the LVQ neural network [52] as a dynamic vector quantization that is integrated in the hidden layer as an activation function, allowing the network to compress these coefficients. The proposed approach adopts the optimized VQ obtained in step 2 as an activation function incorporated in each neuron of the hidden layer, which is an improvement over the present techniques that use the neural network as a black box for lossy compression. As a compression approach, we use a vector quantization neural network, which combines vector quantization with supervised learning. When it comes to supervised video compression, the LVQ neural network, see Fig.6, outperforms the back-propagation one in terms of minimizing errors while preserving rapid convergence [53]. Configure the settings of both the competitive and linear layers as the initial phase in the building of an LVQ neural network. Each frame's HL wavelet band (chosen based on trial results) is used as a vector input to the input layer. The optimized codebook (a vector for each node in the hidden layer) is stored in the hidden layer, and the index used for encoding is the output. To update the codebook, LVQ may be utilized since it is a supervised learning method. Utilizing a feed-forward neural network, it combines clustering and classification procedures.

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

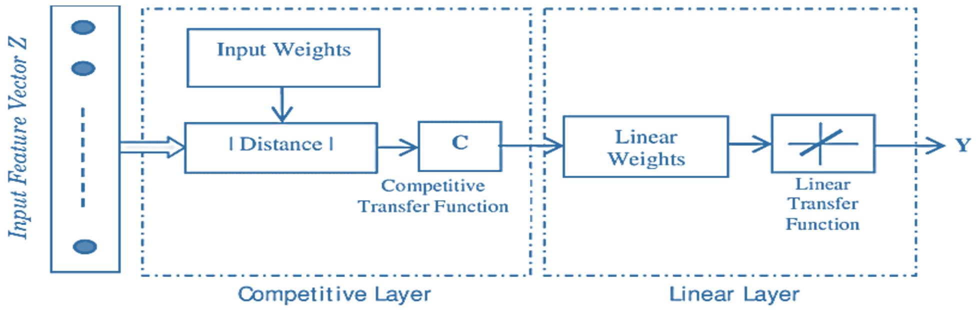

Figure 6. The LVQ competitive neural network architecture.

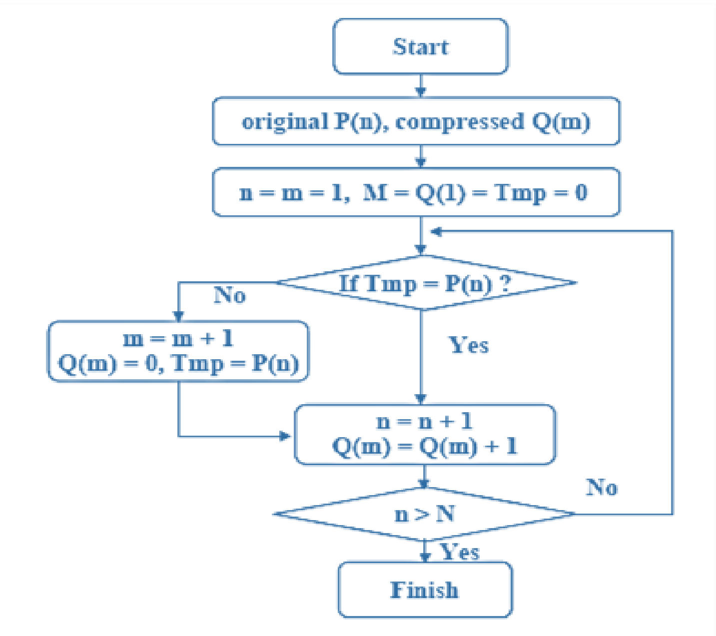

Figure 7. The flowchart of the run length encoding procedure.

For the original data array P(n), which is to be encoded, the temporary variable Tmp is compared with P(n) for increasing the index n. Because the temporary variable Tmp is set to P(n-1) in the n-1 time step, the encoded data array Q(m) should increase until Tmp is not equal to P(n) in the n time step. When the Tmp is equal to P(n), Q(m) should stop increasing and be set to 0. When the index n is greater than N, the total number of original data arrays is P(n), and the iteration process is finished.

Stage 3: Run Length Encoding

After the DPCM lossless compression stage, the quantized coefficient vector and the VQ index vector from the LVQ neural network lossy compression stage are combined into a single vector with carefully defined boundaries so that they may be decoded together. In this situation, each frame has a single unified vector. Run-length encoding (RLE) is used to deal with statistical redundancy among unified vector elements, which increases the compression ratio. If the source has numerous strings of identical symbols, a run-length algorithm will perform extremely well. Each video is ultimately represented as a matrix, with each row containing the RLE of a single frame and the total number of rows being equal to the total number of frames in the movie. Fig 7 illustrates the main steps of run length encoding method [54] [[55].

Step 4: Decompression

As shown in Fig. 8, the stages involved in decompression are the inverse of those involved in compression. To get the combined coefficient vector, we must first apply run-length decoding, see Fig. 9, to each row of the video compression matrix. Each frame's quantized coefficients and VQ index are included in this vector. The uncompressed coefficients (low frequencies) may be obtained by applying inverse DPCM as illustrated in Fig. 10 to the quantized coefficients. In order to get the high frequency coefficients (there is one such vector in each frame), the index value for the VQ vector is translated to the corresponding vector using the codebook table that has been

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

previously saved. The decompressed frame is obtained by combining the bands obtained in the previous phase (LL and HL) with the two unmodified bands (LH and HH) provided by the database and inverting the HWT. To restore the original video, follow the same steps for each row of the compressed matrix.

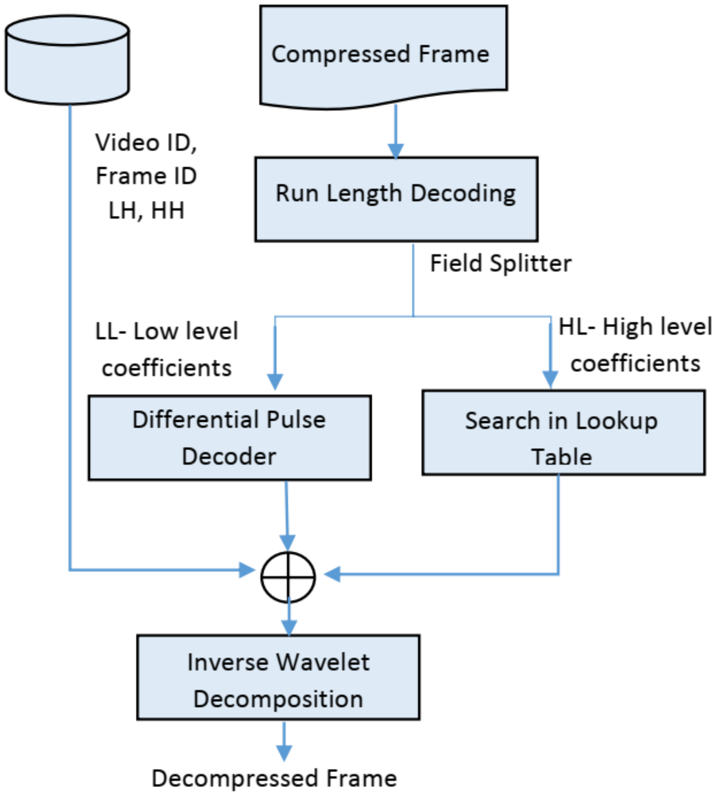

Figure 8. Flow diagram of proposed quantum machine learning video compression system: decompression phase.

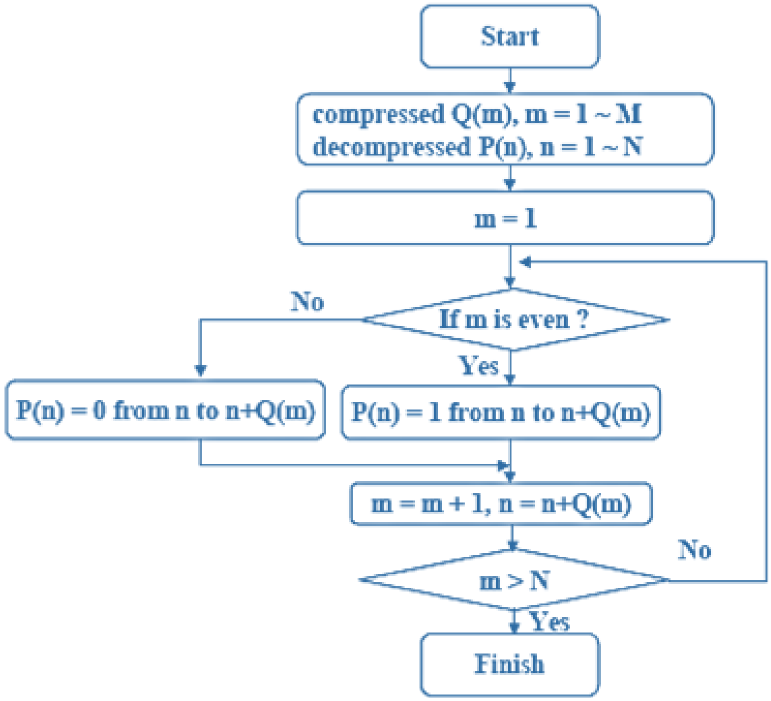

Figure 9. The flowchart of the run length decoding procedure.

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

In the case of run length decoding, it should be determined whether the index m is odd or not when there is an data array Q(m) which is to be decoded. If m is odd, the 1's are occurring in data file repeatedly corresponding to the value of Q(m). And if m is even, the 0's are filled up in the sequences of an interval of next Q(m) points. This process is repeated until the index m is greater than the number of data array M.

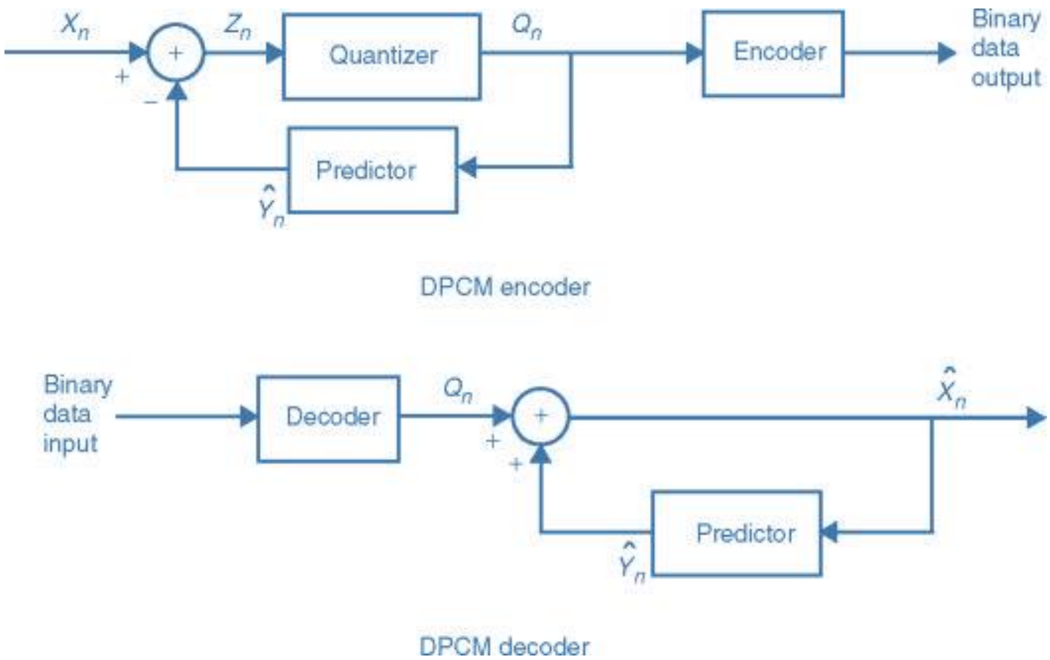

Figure 10. DPCM System Block Diagram

4. Results and Discussions

A benchmark video dataset https://media.xiph.org/video/derf/ was used for the experiments to evaluate the Quantum evolutionary based intelligent vector quantization approach proposed in this study. The testbed is a collection of videos saved in a variety of formats, including avi, and mpeg. Samples of the testbed videos are shown in Table 1 associated with its description of this dataset. All of these videos have a static background, while the action takes place either in a somewhat static foreground, as in Miss America, or in a dynamic foreground, such in Aquarium. The impact of changing Qutrits based Quantum Genetic Algorithm parameters on the overall system assessment has been explored by testing many iterations of the proposed system. Additionally, eight video sequences with varying spatial and temporal information were downloaded in the uncompressed format *.YUV to validate the proposed model's dependability while dealing with Full HD (1920×1080) and Ultra HD (3840×2160) video sequences. SJTU YUV (SJTU 4 K Video Sequences; available online at http://medialab.sjtu.edu.cn/web4k/index.html).

The suggested model has been implemented in MATLAB (Release 2022a). The model has been implemented using the laptop computer with the following specifications: Processor: Intel (R), Core (TM) i7 CPU, L640 @ 2.31 GHz 2.31 GHz with RAM: 4 GB. System type: 64-bit operating system. Microsoft Windows 8.1 Enterprise as running operating system, and Hard Disk: 500 GB. The compression ratio (CR) and peak signal-to-noise ratio (PSNR) are commonly used metrics that were put to use in order to evaluate the effectiveness of the suggested model in comparison to other current strategies. Intuitively, a better signal-to-noise ratio corresponds to a higher PSNR value. In this case, the original frame serves as the signal, while the mistake in reconstruction serves as the noise [56-58]. Using the right values for crossover and mutation probability prevents GA from being stuck at local optima, which is an issue plaguing all GA-based video compression methods. Wide exploration and deep exploitation are balanced in a good GA [59]. To hasten the algorithm's convergence, the best solution produced at each iteration might be subjected to the local search approach based on the utilized Qutrits quantum coding.

The first set of experiments was conducted to demonstrate the effect of the wavelet detail bands on the PSNR quality of the proposed system. Table 2 demonstrates that compression ratio is independent of band type. The proposed approach encodes a small number of lossy detail coefficients in contrast to approximation coefficients, which include a high number of coefficients, which may be one rationale for these results. A single index is used to decode these small coefficients from a lookup table. The index representation of the coefficients in the three

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

bands is similar because of the relationship between the bands. As a result, the compression ratios throughout these spectrums are identical. The results in the table also show that the HL band provides the greatest PSNR since it comprises the most salient features, especially in the frame with the texturing and smooth areas.

To demonstrate how the suggested system's effectiveness varies with codebook size, a second series of experiments was carried out. Table 3 demonstrates that with a bigger codebook, the CR increases but the PSNR decreases. It has been observed that when the codebook size is increased to 256 or more, CR stabilizes or even significantly improves. These results may be explained by the fact that a bigger codebook vector contains more information. Due to the fact that this information is represented by a single index, the compression ratio improves as the codebook grows in size. When decompressing, the lookup table is used to accomplish the searching; if there's an error in the index representation, the mistake will be represented for a lot of data, which will lower the PSNR and, by extension, the video quality.

In the third set of experiments, we compared the compression ratios and quality performance of the proposed system, which employs the Qutrits based Quantum Genetic Algorithm (QQGA) to construct an optimal codebook for vector quantization that is used as an activation function inside the neural network's hidden layer, to that of a purely randomized LBG-based video compression technique [48] and that relies on tradition Qubit based Quantum Genetic Algorithm with the same main steps. Table 4 demonstrates that compared to the LBG video coding method, Qutrits QGA may enhance compression ratio by around 6 % and PSNR by about 10 %. In addition, Qutrits QGA outperforms conventional Qubit QGA by around 3% and 4% for *CR* and *PSNR* respectively.

Table 1: Samples of Benchmark Dataset

Video		Frame resolution	Number of frames	Time	Extension
Aquarium		720 × 480	155	5 sec.	MPEG
Man running		160 × 120	154	5 sec.	AVI
Traffic road		512 × 512	48	2 sec.	MPEG
Akiyo		720 × 480	89	3 sec.	AVI
Boxing		160 ×120	154	5 sec.	AVI
Miss America		640 × 480	89	3 sec.	AVI
Tennis		320 × 240	30	1 sec.	MPEG
Suzie		192 × 144	30	1 sec.	MPEG

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

Table 2: System Performance Evaluation Regarding the Wavelet Details Bands (Code book size=256)

Wavelet Bands	HH		HL		LH	
	CR	**PSNR**	**CR**	**PSNR**	**CR**	**PSNR**
Man running	33.61	41.34	33.61	**45.25**	33.61	39.72
Traffic road	33.52	34.96	33.52	**36.26**	33.52	32.08
Aquarium	33.51	33.82	33.51	**35.86**	33.51	32.17
Akiyo	33.61	44.24	33.61	**45.95**	33.61	42.73
Miss America	33.61	44.56	33.61	**45.97**	33.61	42.91
Boxing	33.61	42.14	33.61	**43.63**	33.61	41.42
Runners (UHD)	33.50	42.90	33.50	**44.08**	33.50	42.34
Fountains (FHD)	33.18	42.94	33.18	**44.15**	33.18	42.52

Table 3: Effect of Codebook Size of Vector Quantization on Video Coding in terms of CR and PSNR based on HL bands

Codebook size	Video	CR	PSNR
16	Man Running	28.93	45.23
	Traffic Road	26.90	32.55
	Aquarium	25.26	34.22
	Akiyo	26.51	35.99
	Miss America	29.92	45.33
	Boxing	26.19	43.65
	Runners (UHD)	25.86	44.10
	Fountains (FHD)	25.94	44.20
32	Man Running	30.66	42.59
	Traffic Road	27.72	33.42
	Aquarium	27.56	33.40
	Akiyo	29.51	34.60
	Miss America	30.37	45.31
	Boxing	28.65	42.36
	Runners (UHD)	29.87	44.03
	Fountains (FHD)	29.65	44.40
128	Man Running	31.50	41.32
	Traffic Road	32.54	33.39
	Aquarium	32.55	34.28
	Akiyo	32.65	42.50
	Miss America	32.87	43.89
	Boxing	32.34	42.36
	Runners (UHD)	33.68	44.08
	Fountains (FHD)	33.43	44.12
256	Man Running	33.63	45.24
	Traffic Road	33.50	36.23
	Aquarium	33.50	35.84
	Akiyo	33.64	45.93
	Miss America	33.68	45.91
	Boxing	33.62	43.60
	Runners (UHD)	33.53	44.15
	Fountains (FHD)	33.20	44.18

Table 4: Performance evaluation with random, GA, qubit QGA, and Qutrits QGA-based cookbook generation with code book size=256

Video	Random codebook building method using LBG		Codebook building using GA		Codebook building using Qubit QGA		Codebook building using Qutrits QGA	
	CR	*PSNR*	*CR*	*PSNR*	*CR*	*PSNR*	*CR*	*PSNR*
Man Running	30.37	37.56	31.61	40.29	32.04	41.77	33.63	45.24
Traffic Road	29.34	27.25	30.67	30.51	31.13	32.18	33.50	36.23

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

Aquarium	28.97	28.25	30.60	31.80	31.48	32.85	33.50	35.84
Akiyo	28.79	27.97	30.32	30.15	31.32	32.34	33.64	45.93
Miss America	28.41	40.70	30.34	25.46	31.35	32.23	33.68	45.91
Boxing	29.67	37.86	30.33	40.86	31.31	42.47	33.62	43.60
Runners (UHD)	28.31	37.63	30.11	40.76	30.59	42.59	33.53	44.15
Fountains (FHD)	28.37	37.71	30.13	40.52	30.84	42.87	33.20	44.18

One of the benefits of a qubit QGA is that it may locate an optimum solution in all dimensions simultaneously. This skill is also helpful when creating a VQ codebook that is optimized for video compression. As a result, we'll have a better chance of obtaining a really representative codebook. The suggested model employs the lossless compression to compress several important coefficients, which accounts for the low compression ratio. When the quantization vector had few items, the two techniques qubit QGA and Qutrits QGA were comparable in all cases. When the number of items to be considered increases, however, Qutrits QGA outperforms qubit QGA in every case. Our method relies on the use of qutrits (a three-level quantum system) to stand in for individuals within a population. The evolution operator, which is in charge of modifying the state of the quantum system, has been built in a basic manner that does not include a lookup table. Qudit, in comparison to qubit, offers a greater state space in which to store and process information, which may improve the effectiveness of the algorithm.

In the fourth round of experiments, we look at how adjusting the threshold for the background subtraction δ in Eq.1 affects the performance of the proposed system. Table 4 shows the PSNR obtained by the system in different videos as the δ value shifts from 0.06 to 0.10. In general, thresholds for video compression algorithms need to be higher when there is more motion in the video, and they need to be lower when there is less motion. Table 6 demonstrates that a better PSNR may be achieved by adjusting the threshold upwards depending on the kind of video being analyzed. After a certain point δ = 0.08, there is no further rise in PSNR. The explanation for these findings is that each codeword in the codebook comprises a tiny quantity of data since the big threshold leads to a reduction in the number of dissimilar pixels across the frames. Due to the enormous size of the codebook, the PSNR of the decompressed data will be low if there was any distortion.

In the next set of experiments, we'll see how our suggested system stacks up against several similar video compression techniques [60][61]. With the use of an ant colony-based motion estimation approach and a modified fast Haar wavelet transform, the temporal redundancy is eliminated in the first comparative procedure [61]. In order to get rid of spatial redundancy, the second approach [61] uses a quick curvelet transform combined with run-length encoding and Huffman coding. Instead, the suggested approach eliminates temporal and spatial redundancy via the use of optimum vector quantization, spatial redundancy through the use of DPCM, and statistical redundancy through the use of run-length encoding.

Table 5 demonstrates that the suggested system outperforms the first and second algorithms in terms of PSNR of the reconstructed video. The new method improves PSNR by around 8% and 11%, respectively. Furthermore, in comparison to the alternative methods, the compression ratio is enhanced by 2 % and 7 % respectively in the suggested system. These results are explained by the fact that Qutrits QGA aids in the construction of an accurate codebook with minimal distortion for the vector quantization method. In addition, as compared to the second technique, the CR is increased when RLE is used for statistical redundancy, DPCM is not used, and vector quantization is used.

The process of creating a codebook may be thought of as a searching problem, with the desired outcome being the identification of the most representative codebook that can be used effectively for video compression [62-64]. More PSNR is expected if the QQGA-based vectors are mapped to their closest codebook equivalent with regard to a distortion function. The evolution operator, which is responsible for the updates to the quantum system state, are used in QQGA to identify the most representative codebook for video compression.

Table 5: The Effect of Background Subtraction Threshold for Reconstructed Videos in Terms of PSNR with code book size=256

Threshold Value δ	Video	PSNR
0.06	Man Running	40.43
	Traffic Road	32.68
	Aquarium	33.02
	Akiyo	41.56
	Miss America	44.91
	Boxing	41.43

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

	Suzie	41.73
	Runners (UHD)	43.11
	Fountains (FHD)	43.15
0.07	Man Running	41.05
	Traffic Road	33.18
	Aquarium	33.53
	Akiyo	42.20
	Miss America	45.68
	Boxing	42.06
	Suzie	42.01
	Runners (UHD)	43.77
	Fountains (FHD)	43.81
0.08	Man Running	41.32
	Traffic Road	33.39
	Aquarium	34.28
	Akiyo	42.50
	Miss America	43.89
	Boxing	42.36
	Suzie	42.31
	Runners (UHD)	44.08
	Fountains (FHD)	44.12
0.1	Man Running	45.24
	Traffic Road	36.23
	Aquarium	35.84
	Akiyo	45.93
	Miss America	45.91
	Boxing	43.60
	Suzie	42.31
	Runners (UHD)	44.15
	Fountains (FHD)	44.18

Table 6: Comparative result with related video coding techniques

Video	**Choudhury, H. et al. [60]**		**Nithin, S. et al. [61]**		**Proposed System**	
	CR	*PSNR*	*CR*	*PSNR*	*CR*	*PSNR*
Man Running	32.35	36.76	25.98	33.49	33.63	45.24
Traffic Road	31.45	30.63	26.24	32.23	33.50	36.23
Aquarium	30.49	30.34	25.72	31.70	33.50	35.84
Akiyo	31.83	32.09	26.41	32.28	33.64	45.93
Miss America	31.41	39.80	27.13	50.31	33.68	45.91
Boxing	32.06	38.65	28.34	34.46	33.62	43.60
Runners (UHD)	33.36	39.16	29.84	35.09	33.53	44.15
Fountains (FHD)	33.19	39.97	29.45	35.16	33.20	44.18

Time complexity analysis, which is a subfield of computational complexity theory and is used to characterize how an algorithm makes use of computer resources, was used to the final set of experiments to assess the proposed system's complexity. Compressing and decompressing takes more time as the number of frames grows, as seen in Table 11. We also know that the majority of the time required to run the program is spent on the compression phase, which includes several different subprograms including the wavelet transform, neural network, DPCM, and RLE. However, this process may be done offline once before the video is sent, and then used whenever necessary. In the case of Full HD and Ultra HD video sequences, compression and decompression times naturally rise. This time is very machine-dependent and may be decreased by using a high-configuration machine or a parallel processing paradigm.

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

Table 7: Average Time Consumed for the Whole System Performance in Seconds (average).

Video	Compression Time	Decompression Time
Suzie	60	2
Miss America	100	4
Aquarium	270	5

4.1 Limitations

In general, there are several limitations to utilizing the suggested system, such as (1) only using static backdrop videos (not a moving camera) to effectively apply the background subtraction technique, and (2) only using grayscale video. Since the identical procedures must be applied to each color channel when working with color video. Finally (3) more memory space is required to store certain information (wavelet HH and LH bands) that are employed in the decoding process,

5. Conclusions

There are many different criteria and parameters that must be met in order to create an effective video compression system, just as there are with any other issue. Vector quantization requires the use of an optimal codebook, which is not always easy to find. Therefore, there is a necessity for optimization-based approaches that can be employed to get an optimal solution that satisfies the criteria. In a perfect world, the optimization strategy would result in the objective function's global optimum. This study presents an improved strategy for video compression using a QQGA based on Qutrits. QQGA can solve the same class of problems as a classic genetic algorithm while vastly speeding up the evolutionary process thanks to quantum parallelization and entanglement of the quantum state. The method is based on the idea of representing individuals within a population as Qutrits (a three-level quantum system). A simple method that does not need a lookup table has been used to generate the evolution operator, which is responsible for the changes to the state of the quantum system. Qudit, with its bigger state space for storing and processing information, may improve the efficacy of the algorithm. The background subtraction technique is used to separate moving objects from frames in order to generate the context-based initial codebook. In addition, Differential Pulse Code Modulation (DPCM) is used to losslessly compress important wavelet coefficients, while Learning Vector Quantization (LVQ) neural networks are used to lossy compress low energy coefficients. After the coefficients have been quantized, Run-Length Encoding is employed to compress them further. The system can be improved such that it can process movies shot with a moving camera instead of a fixed one. In addition, other transformations, such as the ripple and chirplet transforms, may be studied for accurate frame' features representation. Finally, an alternative optimization approach may be used in lieu of the GA to choose the best codebook for vector quantization.

Funding: "This research received no external funding"

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

References

[1] Zhang, Y., Zhu, L., Jiang, G., Kwong, S., Kuo, C. A survey on perceptually optimized video coding. ACM Computing Surveys, vol. 55, no. 12, pp. 1-37, 2023.

[2] Ji, R., Karam, L. Learning-based Visual Compression. Foundations and Trends in Computer Graphics and Vision, vol.15, no. 1, pp. 1-12, 2023.

[3] Dong, Y., Pan, W. A Survey on Compression Domain Image and Video Data Processing and Analysis Techniques. Information, vol.14, no. 3,184, pp.1-21, 2023.

[4] Aliouat, A., Kouadria, N., Maimour, M., Harize, S., Doghmane, N. Region-of-interest based video coding strategy for rate/energy-constrained smart surveillance systems using WMSNs. Ad Hoc Networks, vol. 140, 103076, pp.1-13, 2023.

[5] Jeny, A., Islam, M. Optimized video compression with residual split attention and swin-block artifact contraction. Journal of Visual Communication and Image Representation, vol. 90, 103737, pp.1-14, 2023.

[6] Gandam, A., Sidhu, J., Singh, M., Kaur, H. Graph Theory-Based HEVC Video Compression of Satellite Videos. In Proceedings of the International Conference on Small Satellites, pp. 23-29, Singapore: Springer Nature Singapore, 2023.

[7] Yang, R., Timofte, R., Li, X., Zhang, Q., Zhang, L., Liu, F., He, D., Li, F., Zheng, H., Yuan, W., Ostyakov, P. Aim 2022 challenge on super-resolution of compressed image and video: Dataset, methods and results.

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

In Proceedings of the Computer Vision–Workshops, Part III 2023, pp. 174-202, Cham: Springer Nature Switzerland, 2023.
[7] Khadir, M., Hashmi, M. High Efficient Quality of Video Compression using Variational Autoencoders in Deep Learning. In Proceedings of the International Conference on Power, Control & Embedded Systems, pp. 1-6, 2023.
[8] Mozhaeva, A., Mazin, V., Cree, M., Streeter, L. Video quality assessment considering the features of the human visual system. In Proceedings of the International conference on Image and Vision Computing, pp. 288-300, Cham: Springer Nature Switzerland, 2023.
[9] Winkler, S. Perceptual video quality metrics—A review. Digital video image quality and perceptual coding, vol. 19, pp. 155-180, 2017.
[10] Rahebi, J. Vector quantization using whale optimization algorithm for digital image compression. Multimedia Tools and Applications. vol. 81, issue 14, pp. 20077-20103, 2022.
[11] Barman, D., Hasnat, A., Barman, B. A codebook modification method of vector quantization to enhance compression ratio. In Proceedings of the International Conference in High Performance Computing and Networking, pp. 227-234, Singapore: Springer Singapore, 2022.
[12] Barman, D., Hasnat, A., Barman, B. A quantization based codebook formation method of vector quantization algorithm to improve the compression ratio while preserving the visual quality of the decompressed image. Multidimensional Systems and Signal Processing, vol. 34, pp. 127–145, 2023.
[13] Guo, J., Wu, C., Huang, Z., Wang, F., Huang, M. Vector Quantization Image Compression Algorithm Based on Bat Algorithm of Adaptive Separation Search. In Proceedings of the International Conference on Advanced Intelligent Systems and Informatics, pp. 174-184, Springer International Publishing, 2022.
[14] Bidwe, R., Mishra, S., Patil, S., Shaw, K., Vora, D., Kotecha, K., Zope, B. Deep learning approaches for video compression: a bibliometric analysis. Big Data and Cognitive Computing, vol. 6, issue 2, 44, pp. 1-40, 2022.
[15] Hsu, C., Hung, T., Hsu, C. Optimizing immersive video coding configurations using deep learning: A case study on TMIV. ACM Transactions on Multimedia Computing, Communications, and Applications, vol. 18, issue 1, pp. 1-25, 2022.
[16] Putra, A., Gaffar, A., Sumadi, M., Setiawati, L. Intra-frame Based Video Compression Using Deep Convolutional Neural Network (DCNN). International Journal on Informatics Visualization, vol.6, no. 3, pp. 650-659, 2022.
[17] Moussafir, M., Chaibi, H., Saadane, R., Chehri, A., Rharras, A., Jeon, G. Design of efficient techniques for tomato leaf disease detection using genetic algorithm-based and deep neural networks. Plant and Soil, vol. 479, no. 1-2, pp. 251-266, 2022.
[18] Bas, E., Egrioglu, E., Kolemen, E. Training simple recurrent deep artificial neural network for forecasting using particle swarm optimization. Granular Computing, vol. 7, no. 2, pp. 411-420, 2022.
[19] Dubey, R., Agrawal, J. An Improved Genetic Algorithm for Automated Convolutional Neural Network Design. Intelligent Automation & Soft Computing, vol. 32, no. 2, pp. 747-763, 2022.
[20] Chavan, P., Sheela, B., Murugan, M., Chavan, P., Kulkarni, M. An Analysis of Codebook Optimization for Image Compression: Modified Genetic Algorithm and Particle Swarm Optimization Algorithm. In Proceedings of Fourth International Conference on Communication, Computing and Electronics Systems, pp. 849-866, Singapore: Springer Nature Singapore, 2023.
[21] Lamata, L. Quantum machine learning and quantum Biomimetics: A perspective. Machine Learning: Science and Technology, vol. 1, no. 3, 033002, pp. 1-12, 2020.
[22] Zeguendry, A., Jarir, Z., Quafafou, M. Quantum Machine Learning: A Review and Case Studies. Entropy, vol. 25, no. 2, 287, pp.1-41, 2023.
[23] Choe, I., Kim, G., Kim, N., Ko, M., Ryom, J., Han, R., Han, T., Han I. Can quantum genetic algorithm really improve quantum back propagation neural network. Quantum Information Processing, vol. 22, no. 3, 154, pp. 1-12, 2023.
[24] Belkebir, D. DIP-QGA: a secure and robust watermarking technique based on direct image projection and quantum genetic algorithm. International Journal of Information and Computer Security, vol. 20, no. 3-4, pp. 221-247, 2023.
[25] Tkachuk, V. Quantum genetic algorithm based on Qutrits and its application. Mathematical Problems in Engineering, vol.2018, 8614073, pp. 1-8, 2018.
[26] Dutta, T., Dey, S., Bhattacharyya, S., Mukhopadhyay, S., Chakrabarti, P. Hyperspectral multi-level image thresholding using Qutrit genetic algorithm. Expert Systems with Applications, vol. 181, 115107, pp. 1-19, 2021.
[27] Joy, H., Kounte, M., Chandrasekhar, A., Paul, M. Deep Learning Based Video Compression Techniques with Future Research Issues. Wireless Personal Communications, vol.131, pp. 1-27. 2023.

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

[28] Petreski, D., Kartalov, T. Next Generation Video Compression Standards–Performance Overview. In Proceedings of the 30th International Conference on Systems, Signals and Image Processing, Ohrid, North Macedonia, pp. 1-5, 2023.
[29] Li, J., Li, B., Lu, Y. Neural Video Compression with Diverse Contexts. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 22616-22626, 2023.
[30] Gomes, C., Azevedo, R., Schroers, C. Video Compression with Entropy-Constrained Neural Representations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18497-18506, 2023.
[31] Duong, L., Li, B., Chen, C., Han, J. Multi-rate adaptive transform coding for video compression. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 1-5, 2023.
[32] Isik, B., Guleryuz, O., Tang, D., Taylor, J., Chou, P. Sandwiched video compression: Efficiently extending the reach of standard codecs with neural wrappers. ArXiv preprint arXiv: 2303.11473, pp.1-5, 2023.
[33] Jin, D., Lei, J., Peng, B., Pan, Z., Li, L., Ling, N. Learned Video Compression with Efficient Temporal Context Learning. IEEE Transactions on Image Processing, vol. 32, pp. 3188 – 3198, 2023.
[34] Lin, R., Wang, M., Zhang, P., Wang, S., Kwong, S. Multiple Hypotheses Based Motion Compensation for Learned Video Compression. Neurocomputing, vol. 548,126396, pp.1-15, 2023.
[35] Wang, Y., Chan, P., Donzella, V. Semantic-Aware Video Compression for Automotive Cameras. IEEE Transactions on Intelligent Vehicles, vol.8, issue 6, pp.1-12, 2023.
[36] Liu, B., Chen, Y., Machineni, R., Liu, S., Kim, H. MMVC: Learned Multi-Mode Video Compression with Block-based Prediction Mode Selection and Density-Adaptive Entropy Coding. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18487-18496, 2023.
[37] Habib, S., Albattah, W., Alsharekh, M., Islam, M., Shees, M., Sherazi, H. Computer Network Redundancy Reduction Using Video Compression. Symmetry. vol.15, issue 6, 1280, pp.1-14, 2023.
[38] Saini, D., Kamble, S., Shankar, R., Kumar, M., Kapila, D., Tripathi, D. Fractal video compression for IOT-based smart cities applications using motion vector estimation. Measurement: Sensors. vol. 26, 100698, pp.1-15, 2023.
[39] Malani, R., Suprapty, B., Putra, A., Gaffar, A. Inter-frame video compression based on adaptive fuzzy inference system compression of multiple frame characteristics. Knowledge Engineering and Data Science, vol. 6, no. 1, pp. 1-14, 2023.
[40] Pourreza, R., Le, H., Said, A., Sautiere, G., Wiggers, A. Boosting neural video codecs by exploiting hierarchical redundancy. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 5355-5364, 2023.
[41] Fathima, N., Petersen, J., Sautière, G., Wiggers, A., Pourreza, R. A Neural Video Codec with Spatial Rate-Distortion Control. In Proceedings of the Winter Conference on Applications of Computer Vision, pp. 5365-5374, 2023.
[42] Qi, L., Li, J., Li, B., Li, H., Lu, Y. Motion Information Propagation for Neural Video Compression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6111-6120, 2023.
[43] Yang, R., Yang, Y., Marino, J., Mandt, S. Insights from generative modeling for neural video compression. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 45, issue 8, pp. 9908 – 9921, 2023.
[44] Belyaev, E. An Efficient Compressive Sensed Video Codec with Inter-Frame Decoding and Low-Complexity Intra-Frame Encoding. Sensors, vol. 23, no. 3, 1368, pp. 1-23, 2023.
[45] Kalsotra, R., Arora, S. Background subtraction for moving object detection: explorations of recent developments and challenges. The Visual Computer, vol. 38, no. 12, pp. 4151-4178, 2022.
[46] Mirzaei, B., Nezamabadi-pour, H., Abbasi-moghadam, D. An effective codebook initialization technique for LBG algorithm using subtractive clustering. In Proceedings of the Iranian Conference on Intelligent Systems, Feb 4, pp. 1-5, 2014.
[47] Ferreira, F., Leitao, H., Lopes, W., Madeiro, F. Hybrid firefly-Linde-Buzo-Gray algorithm for channel-optimized vector quantization codebook design. Integrated Computer-Aided Engineering, vol. 24, no. 3, pp. 297-314, 2017.
[48] Bao, W., Zhu, C. A secure and robust image encryption algorithm based on compressive sensing and DNA coding. Multimedia Tools and Applications, vol. 81, Issue 11, pp. 15977-15996, 2022.
[49] Zou, Y., Luo, C., Zhang, J. DIFLD: domain invariant feature learning to detect low-quality compressed face forgery images. Complex & Intelligent Systems. vol.10, Issue 1, pp.1-12, 2023.

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024

[50] Song, Y., Yu, Z., Liu, E., Huang, H., Sun, K., Yin, F., Xu, K. Digital radio-over-fiber system based on differential pulse code modulation and space division multiplexing. Optics Letters, vol. 48, no. 7, pp. 1806-1809, 2023.

[51]Liu, R., Wang, C., Tang, A., Zhang, Y., Yu, Q. A twin delayed deep deterministic policy gradient-based energy management strategy for a battery-ultracapacitor electric vehicle considering driving condition recognition with learning vector quantization neural network. Journal of Energy Storage, vol. 71, 108147, pp.1-11, 2023.

[52] Zhang, K., Chen, Z., Yang, L., Liang, Y. Principal Component analysis (PCA) based sparrow search algorithm (SSA) for optimal learning vector quantized (LVQ) neural network for mechanical fault diagnosis of high voltage circuit breakers. Energy Reports, vol. 9, pp. 954-962, 2023.

[53] Radhika, K., Geetha, D. Retraction Note: Augmented Recurrence Hopping Based Run-Length Coding for Test Data Compression Applications. Wireless Personal Communications, vol. 128, 747, pp.1-20, 2023.

[54]Khairi, N., Jambek, A. Run-length encoding (RLE) data compression algorithm performance analysis on climate datasets for Internet of Things (IoT) application. International Journal of Nanoelectronics and Materials, vol. 14, pp. 191-197, 2021.

[55] Joy, H., Kounte, M., Chandrasekhar, A, Paul, M. Deep Learning Based Video Compression Techniques with Future Research Issues. Wireless Personal Communications. Vol. 131, pp. 2599–2625, 2023.

[56] Menon, V. Multi-resolution encoding and optimization for next generation video compression. arXiv preprint arXiv: 2301.12191 , pp.1-8, 2023.

[57][Ghassab, V., Gonsalves, R., Mathur, S., Bouguila, N. Optimizing Video Compression with CNN-Based Autoencoders with Chroma Subsampling. SMPTE Motion Imaging Journal, vol. 132, no. 3, pp. 18-26, 2023.

[58] Alsmirat, M., Sharrab, Y., Tarawneh, M., Al-shboul, S., Sarhan, N. Video coding deep learning-based modeling for long life video streaming over next network generation. Cluster Computing, vol. 26, no. 2, pp. 1159-1167, 2023.

[59] Belodedov, M., Fonkants, R., Safin, R. Development of an Algorithm for Optimal Encoding of WAV Files Using Genetic Algorithms. In Proceedings of the International Youth Conference on Radio Electronics, Electrical and Power Engineering, Vol. 5, pp. 1-6, 2023.

[60] Choudhury, H., Sinha, N., Saikia, M. Nature inspired algorithms (NIA) for efficient video compression–A brief study. Engineering Science and Technology, an International Journal, vol. 23, no. 3, pp. 507-526, 2020.

[61] Nithin, S., Suresh, L., Krishnaveni, S., Muthukumar, P. Developing novel video coding model using modified dual-tree wavelet-based multi-resolution technique. Multimedia Systems, vol. 28, no. 2, pp. 643-657, 2022.

[62] Pandit, S., Shukla, P., Tiwari, A., Shukla, P., Maheshwari, M., Dubey, R. Review of video compression techniques based on fractal transform function and swarm intelligence. International Journal of Modern Physics, vol. 34, no. 08, 2050061, pp.1-15, 2020.

[63] Ravuri, R. Diamond Search Optimization-Based Technique for Motion Estimation in Video Compression. International Journal of e-Collaboration, vol. 19, no. 3, pp. 1-14, 2023.

[64]Veerasamy, B., Bharathi, B., Ahilan, A. Video compression using improved diamond search hybrid teaching and learning-based optimization model. The Imaging Science Journal, vol.71, Issue 6, pp.1-15, 2023.

DOI: https://doi.org/10.54216/JCIM.150205
Received: April 27, 2024 Revised: June 24, 2024 Accepted: October 11, 2024