

Fast Fractal Coding of MRI Images using Deep Reinforcement Learning

Bejoy Varghese¹, S. Krishnakumar²

Angmaly, Federal Institute of Science and Technology, Ernakulam, Kerala, India¹
STAS, M G University Research Centre, Ernakulam, Kerala, India²

Abstract—This paper presents an algorithm based on Fractal theory by using Iterated Function Systems (IFS). An efficient and fast coding mechanism is proposed by exploiting the self similarity nature in the Brain MRI images. The proposed algorithm utilizes Deep Reinforcement Learning (DRL) technique to learn the transformations required to recreate the original image. We avail of the Adaptive Iterated Function System (AIFS) as the encoding scheme. The proposed algorithm is trained and customised to compress the Medical images, especially Magnetic Resonance Imaging (MRI). The algorithm is tested and evaluated by using the original MR head scan test images. It learns from an existing biomedical dataset viz The Internet Brain Segmentation Repository (IBSR) to predict the new local affine transformations. The empirical analysis shows that the proposed algorithm is at least 4 times faster than the competitive methods and the decoding quality is far distinct with a reduction in the bit rate.

Keywords—Fractal compression; deep reinforcement learning; MRI image compression; deep learning; adaptive fractal coding

I. INTRODUCTION

Medical imaging has become one of the most rapidly growing fields in image processing and medical research. It includes multimodality imaging techniques like Computed Tomography (CT), Magnetic Resonance Imaging (MRI), Ultrasound (US), Elastography, and Digital Subtraction Angiography (DSA). Medical images help doctors in diagnosis, clinical staging and to prescribe therapeutics to heal the disease. But a large number of such images demand enormous storage space and the transmission bandwidth of the PACS. These requirements demand the need for high-quality medical image compression algorithms.

Efforts in reducing the encoding time with better SSIM cause the loss of information on the lesion, leading to misdiagnosis and does not achieve the required effect. Therefore, a Machine Learning-based FIC algorithm for medical images is proposed in this work. The model uses reinforcement learning techniques to gain a better compression ratio with much less encoding time. Biomedical images exhibit a very high structural similarity within the image itself. Because of this self-similarity [1], the learned algorithm can compress the image with a better compression ratio at high PSNR.

A lossless image compression seems to be more suitable for medical images, as almost all the information in it contributes a lot in the diagnosis process. The compression ratio offered by a lossless compression is very much less than the lossy compression schemes, causing it less suitable to reduce the storage and bandwidth requirements. Lossy compression schemes offer a very high level of compression ratio by omitting certain information in the source image. This causes

certain degradations in the reconstructed image, which may lead to inevitable loss of information. So many researches in this area attempt to improve the reconstruction quality of the encoded images at a fixed code rate. At the same time it's evident that an efficient lossless compression technique can play a crucial role in managing the storage space and transmission bandwidth. There are different ways to look into this problem.

The primary lossy compression technique such as JPEG [2] uses the method of identifying the information in terms of frequency components, which is more sensitive to the human eye. Another lossy compression method such as Fractal doesn't consider any of the frequency information, instead it looks for the similarities present in an image. Fractal coding is a lossy coding compression scheme, which utilizes the self similarities in an image. Fractal-based Image Compression (FIC) is rarely used in the area of biomedical images. It has been discarded due to the increased encoding time complexity of the existing algorithms.

If the lossless image compression is considered, PAQ algorithm exhibits a better performance as compared to the similar algorithms as the predictor in the PAQ coder takes a decision only based on the weighted probabilities from a large number of predictors. However, PAQ is a well stitched algorithm to compress the text rather than an image. Another image file format, TIFF is suitable to archive images, as it can hold images both in lossy and lossless schemes [3].

Hence it is observed that there are two ways to develop an efficient compression scheme for medical images. The first is focussed on the self similarity aspect of medical images, which can help in calculating the affine transformations required to reach a single fixed point. This may help us to achieve a better compression ratio and less decoding time, as promised by the fractal theory. The second way focuses on the computational theories to reduce the algorithm complexity and thus the encoding time. It is observed that a Deep Reinforcement Learning algorithm (DRL) [4] is capable of predicting the fractal similarities in an image with appreciably less time in comparison with the classical and Adaptive IFS compression schemes [5].

Iterated Function System (IFS) is a well accepted method to generate the fractals. It is a finite collection of contractive maps w_i and contractivity factors s_i in the complete metric space (X, d) , where $w_i: X \rightarrow X$ and $i = 0, 1, \dots, n$. The representation of IFS is $\{X; w_1, w_2, \dots, w_n\}$ with its contractivity factor $s = \max\{s_1, s_2, \dots, s_n\}$. The value of s lies in between $0 & 1$; $1 > s \geq 0$. By applying the contractive maps in a recursive manner on any arbitrary values leads to the generation of a fixed single point,

which is called the attractor of a particular IFS. The thrust process in fractal generation is to generate the contractive maps or affine transformations. The most celebrated Collage theorem [6] helps to identify the affine transformations that minimizes the distance between the given subsets. So the required process is to compute the coefficients of contractive maps and its probability factors. This can be calculated by using the Markov operator, which itself is a contractive map in the complete metric space of probabilities. It has been found that the collage theorem is the most suitable method to calculate the IFS with probabilities, hence to identify the fixed point of the block under consideration. The modified version of classical fractal compression method based on probability and multiscaling division proves to be more efficient in terms of encoding time and computational complexity [7].

In the case of biomedical images, the number of similar patterns are immense and repetitive over the time scale. Majority of the biomedical imaging techniques produce multi-modality images and the resemblance in the image of the same organ for different patients is high [8]. Hence, the initial idea is to develop an adaptive and efficient Fractal based method, capable of using the self-similarity essence in the human body to compress the respective biomedical images.

The paper is structured as follows. Section 2 explains the fundamentals of fractal compression and latest findings in the field. The idea of reinforcement learning and its applications to fractal compression is explained in section 3. Section 4 explains the proposed method. Results from the work are discussed in section 5. Section 6 concludes the paper.

II. RELATED WORKS

The idea of applying IFS in fractal image compression is suggested by M. Barnsley [9]. Later on his student A Jacquin [10], [11] could automate the basic IFS by using Markov contractive operators. These developments upsurge the use of IFS based fractal image compression in 1990. But still the speed of the encoding process did not meet the practicality. Y Fisher [12] developed a quad tree based partitioning system that did much to ameliorate the encoding process.

Despite all these advancements, the biggest problem still pulls back the fractal based image compression is the calculation of affine transformations, by comparing domain and range blocks. B. Hurtgen [13] suggested to consider the average pixel intensities and block variances to limit the number of possible domain range comparisons. Similarly, the idea of reducing range domain comparisons by applying the Nearest Neighbour Search algorithm was suggested by D. Saupe [14]. FASON algorithm developed by Tan [15], insists to do the comparison directly without storing the domain block pool. This was followed by the use of entropy which is capable of representing the statistical characteristics of pixel data. This method is developed by Yusong Tan and is observed as the most successful method in the pile of various classical IFS algorithms. All these works were mainly oriented towards the Search-less technique in calculating the contractive maps. Wang et. al [16] developed a No-search algorithm to improve the quality of decoded images generated by the quad tree based no-search algorithm implemented by Furoo and Hasegawa's [17]. However, both the search-less and no-search algorithms could not guarantee the quality of decoded images.

All these FIC based algorithms are categorized into three: Classification based, Feature vector based methods and meta heuristic approach. The first method uses a common characteristic metric to classify the domain and range blocks into a predetermined pool. This helps to restrict the search within a limited or same class of blocks [18], [13]. But the second method needs to calculate a particular feature of the partitioned image to classify into different block pools or to discard it from a particular range- domain comparisons [14], [19]. First generation of the third method utilized Genetic Algorithms(GA) [20], Particle swarm optimization [21], ant colony which seems inefficient in terms of encoding time complexity. Because the GAs use brute force search between pairs after considering the mutation to obtain best pairs. The search may never converge, if the system can't find the best suitable domain-range pairs. This process is very similar to the exhaustive search algorithm used in baseline FIC. GA uses boundary conditions to limit the search space and hence to finish the exhaustive searching. This limitation can be overcome by utilizing the characteristics of parallel computing. On the other hand, GA is capable of offering better compression ratio and compression accuracy, which in turn leads to a better PSNR and makes it more suitable for low bit rate image compression applications [22].

Second generation meta heuristic approach adopts the statistical learning theories and proves that the computational complexity is barely minimum as it utilizes algorithms like stochastic gradient descent, the Least square optimization etc. A typical Neural Network (NN) algorithm encodes the input images to vectors in latent space and hence to make it more compact. The two main categories of NN are one-time feed forward frameworks and multi-stage recurrent frameworks [23]. Both these frameworks have their own characteristics, pros and cons. It is observed that feed forward neural networks take less time to encode and decode, as the network needs to execute fewer times. The Training phase is also easier in feed forward because the back propagation path is shorter and shallower in contrast with the recurrent networks.

Todeciri et. al [24] developed a recurrent convolution LSTM [25] based network for learned image compression and proves as an efficient method to handle the variable bit rate algorithms. There have been many variants of the basic Divisive Normalization method proposed by Balle et. al [23] and all such algorithms exhibit an excellent performance in compressing the images. Nakanishi [26] developed a 3D convolutional neural network for learning the conditional probability model to compress the image. Rippel et. al [27] proposed a model based on adversarial loss function and its decoding process is improved by Tschannen et. al [28]. He suggested using the Generative adversarial network instead of the loss function, proves to be the more suitable method to improve the decoding image quality with very low bit rate. The basic concept of policy based reinforcement learning approaches seems to be very efficient in predicting the required transformations [29], [28].

Chen et.al [30] proposed a Non-Local Attention optimization and Improved Context modeling-based image compression (NLAIC) algorithm that relies on trained deep neural networks to achieve improved rate distortion. Ma et. al [31] customized an architecture based on neural network and wavelet trans-

form capable to support both lossy and lossless compression schemes. Cheng et.al developed a flexible entropy model based on discretized Gaussian mixture likelihoods by taking the advantage of recent attention modules and is proved its efficiency in reducing the latency [32].

III. BACKGROUND

Reinforcement learning is the method of learning the best action to optimize the solution based on the reward or punishment. The system that works to learn the action is called an agent and the system that provides the reward or punishment is called environment. So to decide on the action that affects the environment, there are many algorithms that exist. One among the widely known methods is called Q learning. Q learning will decide the action based on a table called Q-Table. The algorithm continuously updates the table based on the observations and rewards from the environment. It uses the Bellman equation to calculate the action values. Bellman equation refers to a set of equations that calculate the value from the reward and discounted future values.

$$Q(s, a) = r(s, a) + \gamma \times \max_a Q(s', a) \quad (1)$$

In the Equation (1), $r(s, a)$ is the immediate reward by taking action a and in state s . $Q(s', a)$ represents the Q -value possible from the next state s' . The γ the discount factor to diminish the effects of future Q -value. This is a recursive equation, that starts with random values of Q in the initial state. On each iteration of the algorithm, the values will be updated based on the reward. In a practical implementation, The Equation (1) can be updates to include the learning rate α .

$$Q(S_t, A_t) = Q(S_t, A_t) + \alpha [R_{t+1} + \gamma \times \max_a Q(S_{t+1}, a) - Q(S_t, A_t)] \quad (2)$$

But the major issue with Q Learning is that the size of the table becomes very large depending on the dimensionality of the input. So applying the Q Learning algorithm to relatively large data input such as images to take the decision can be a cause of slower execution. Following section presents a solution to the issue by applying a Deep Neural network to make a policy decision.

A. Deep Reinforcement Learning (DRL)

In many cases the decision process is a high dimensional problem based on the input. So to make use of the reinforcement learning technique in high dimensional space the decision process is modified by introducing a multi layer neural network for learning the policy. Depending on the size of the problem the neural network may use the deep learning technique to achieve a good accuracy. In the case of a fractal compression method, the action space or the set of transformations required to compress the image is too large to be completely known to the system. The neural network can approximate the policy function that can be used to map the states to action values. In classical RL algorithms, mapping is based on the lookup table which stores all possible combinations of state value pairs.

The input observation and reward will be used to compose a state matrix and feed to a policy network. Policy network is a deep neural network that was trained to predict the action

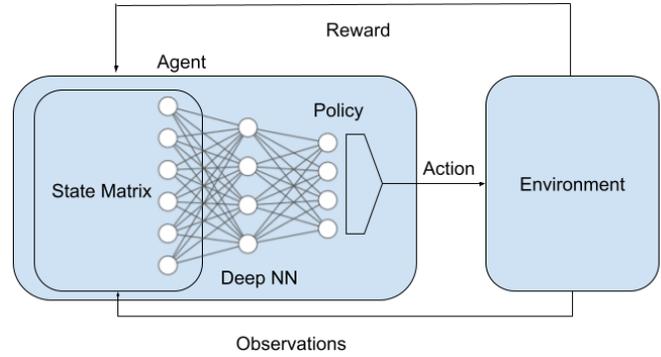


Fig. 1. Deep Reinforcement Learning System.

based on a set of states. Figure 1 shows the architecture of the deep reinforcement learning method. A multi layer neural network will predict the Q values and actions based on the state matrix. Back propagation algorithm is used to train the policy network. A multi layer neural network will predict the Q values and actions based on the state matrix. Back propagation algorithm is used to train the policy network. Deep learning method is used in conjunction with Iterated system function to achieve the fractal compression on raw image data.

IV. PROPOSED METHOD

To reduce the time complexity of the classical fractal compression methods, the brute force search is replaced by the Deep Reinforcement Learning system. Consider T represents the set of Transformation generated by the classical fractal compression for a raw image X . Then the T' represents the set transformation predicted by the DRL system. The relation between T and T' is calculated using Pearson Correlation Coefficient (PCC). PCC is -1 if the transformations are entirely different, 1 if the Transformations are equal and 0 if there is no linear correlation. Then the reward is calculated using the Equation (3).

$$r = \text{conv}(T, T') / (\sigma T \sigma T') \quad (3)$$

In the Equation (3) conv is covariance, σT is the standard deviation of T and $\sigma T'$ is the standard deviation of T' .

A. DRL Training Process

To adapt the fractal compression logic proposed system uses a function approximation such as a neural network with parameter θ to estimate the Q -Values.

$$L_i(\theta_i) = \mathbb{E}_{s,a,r,s' \sim \rho(\cdot)} [(y_i - Q(s, a; \theta_i))^2] \quad (4)$$

In Equation (4), $y_i = r + \gamma \max_{a'} Q(s', a'; \theta_{i-1})$ Where y_i is called the temporal difference and $y_i - Q$ is called temporal difference error. The ρ is called the behavioral distribution by considering the states s, a, r, s' .

To train the network to predict the transformations, the output from a classical fractal compression is considered as the true transformation value for the raw image X . The training process of DRL is described in the Code Snippet 1.

```
1 Data: Training Image set
2 Result: Trained network
3 image index=1;
4 while image index i~= size of training
  set do
5   read image;
6   range block set=partition(image);
7   domain block set=partition(image);
8   range index=1;
9   while range index = size of range
    blocks do
10    read range block;
11    domain book= transformations (
      domain blocks);
12    if Does any member of domain
      book matches with range
      block then
13      record the
        transformation;
14    end
15    else
16      record the best possible
        transformation T;
17    end
18    increase range index;
19  end
20  T' = RL agent prediction (image);
21  Calculate reward r from T and T';
22  forward the reward r to the agent.
23  end
```

Code Snippet 1: DRL Training algorithm

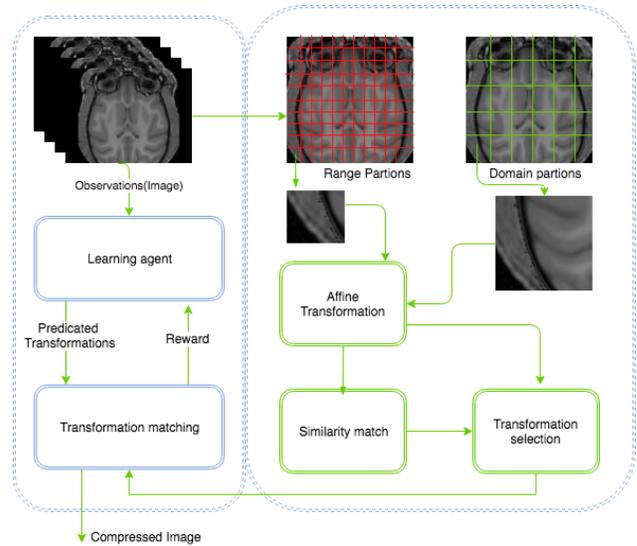


Fig. 2. System Architecture of DRL based Fractal Compression.

Figure 2 shows the system architecture of DRL based fractal compression. Raw image set is compressed using the classical fractal compression algorithm to generate the data for training the DRL. Classical compression algorithm uses the domain-range comparison to obtain the transformation set T. To avoid the time taken by the classical fractal compression, the entire raw data images are compressed in advance before training the DRL system.

The proposed system detaches the classical compression scheme from generating the transformations, once the RL algorithm learns enough to make more accurate predictions for the particular raw image inputs. To compress a new image, the raw data is passed to the DRL system to predict the transformations. The trained DRL predicts the transformation in the context of training data, in this case the MRI images. To compress any other type of images, the DRL system. The results obtained from the proposed solution is discussed in the following section.

B. Preprocessing and Model Architecture of Policy Network

The input raw image consists of three channels that follow the RGB standard. Each value in the image is 8bit size. The resolution of the image is varying from 400x400 pixels to 5120x5120 pixels. Images are re-scaled to the resolution of 800x800 pixels. The size to the input image array to the policy network is set to the maximum size of the image input. If the resolution is less than the input size of the network, the

image is aligned to the center of the 800x800 pixels and the rest of the input taken as 0. Each layer in the input image is treated separately and the transformations are combined after processing the three channels. The first layer of the neural network consists of 64000 neurons. The second, third and fourth layers include 1200,1200 and 800 neurons respectively. So the output layer of the network consists of 800 neurons, corresponding to the number of transformations. All together the network comprises 79 million parameters.

V. RESULT AND DISCUSSION

In this section we discuss some of the conducted experiments, the experimental setup and the analysis. The experiments are using the images from The Internet Brain Segmentation Repository (IBSR).

A. Experiments

We have performed 10 experiments using 3 different types of MRI images — Sagittal, Coronal and Cross-sectional.

In each experiment the training sample sizes are 500, 1500, 2500, 3500, 4500, 5500, 6500, 7500, 8500, 9500 respectively. Each experiment includes a validation set to verify the performance of the system. The size of the validation set is 20% of the training set. All the experiments use the same policy network architecture, learning algorithm and hyperparameters settings. The training of the network uses RMSProp algorithm with a mini batch size of 16 samples. The performance of the system is measured on the validation test after each experiment, by measuring the PSNR of the uncompressed image.

B. Evaluation Metrics

- 1) **Execution time:** It indicates the total time required to complete the execution of the algorithm. It does not include the time required to read or write images.
- 2) **Peak Signal to Noise Ratio (PSNR) :** Indicates the quality of the reconstructed image. It is the ratio of

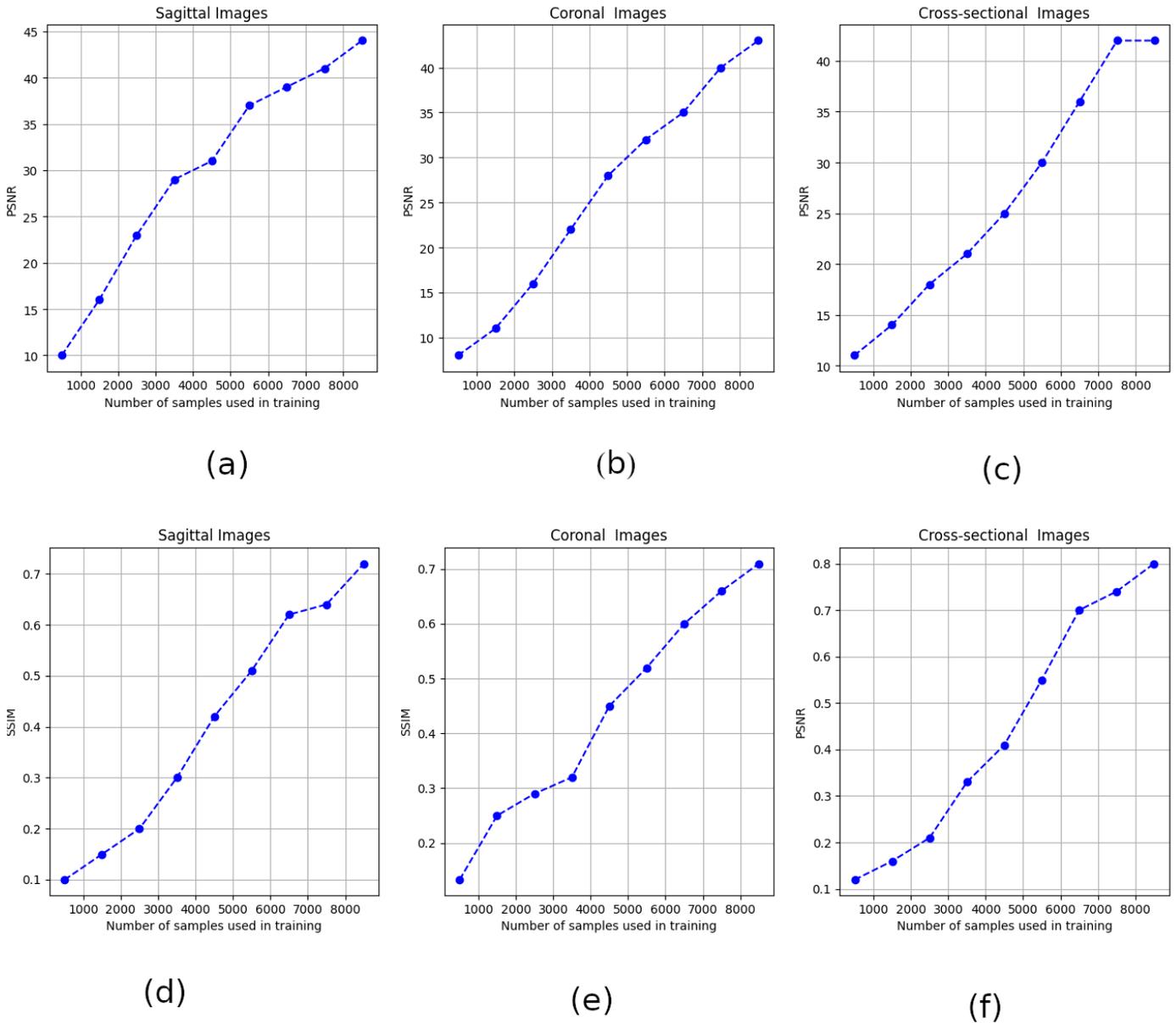


Fig. 3. (a)-(c) PSNR Variations with Respect to the Training Samples Size. (d)-(e) SSIM Variations with respect to Sample Size.

maximum possible power of the signal and the power of corrupting noise in it.

- 3) **Structural Similarity Index Measure (SSIM):** SSIM refers to the structural similarity between the two images. In the case of compression, SSIM works as a metric that indicates the change in the compressed image.
- 4) **Space saving (SS):** Space saving shows the amount of space that can be saved using the compressed image instead of uncompressed image. IT can be derived from the equation, $SS = 1 - \frac{\text{Compressed size}}{\text{Uncompressed size}}$
- 5) **Compression Ratio:** It is defined as the ratio between the size uncompressed image to size of compressed image.

C. Evaluation of Learning Process

The evaluation of the learning process is achieved by feeding the system with a set of training samples in batch and measuring the PSNR and SSIM. The batch size is increased step by step for observing PSNR and SSIM. Another experiment conducted by applying the natural images as the test set for the network that is trained using MRI dataset. Figure 4 shows the PSNR variations of a Cross-sectional brain MRI image trained using different sample size batches.

From Figure 4, we observe that PSNR value is proportional to the training batch size. In Figure 4, part a, b and c shows the decoded image for the Cross-sectional brain MRI image.

Figure 3 shows the SSIM and PSNR variations with respect to the training batch size of Sagittal, Coronal and Cross-

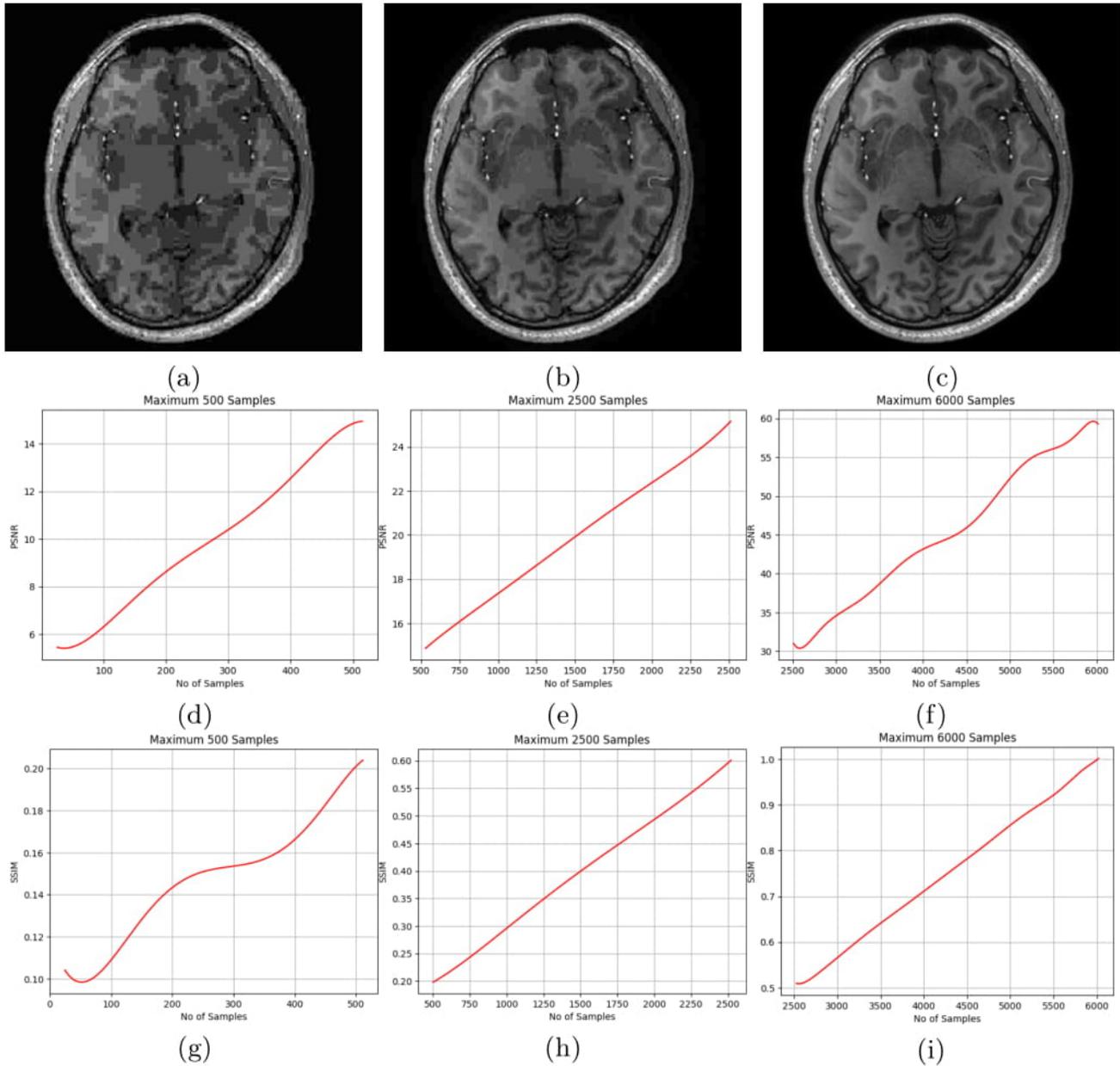


Fig. 4. a)-(c) - Decompressed Image After Training with Sample Size 500, 2500 and 6000 respectively. (d)-(f) - Variations in PSNR with respect to the Sample Set Size 500,2500 and 6000 respectively.(g)-(i) - Variations in SSIM with respect to the Sample Set Size 500,2500 and 6000 respectively.

sectional brain MRI images. It shows that transformation prediction significantly improves by increasing the training batch size. In Figure 5, a cross evaluation of the system is conducted by applying different sets of test images. It shows that system suffers from the over fit towards a trained data set. And the prediction performance of the system completely depends on the category of images used in the training.

So the evaluation process summarizes that the system is able to achieve better results if the training process and testing process uses the same category of images.

System performance is evaluated against four fractal based

compression methods, three classical compression methods and 3 machines learning based compression methods. Evaluation matrices include the PSNR, SSIM, Space Saving, Compression ratio and execution time.

Figure 6 shows the results obtained after training the proposed system with 9000 samples. Performance of the proposed algorithm is compared with three Non-fractal and widely adopted image compression methods-JPEG, PNG and TIFF.

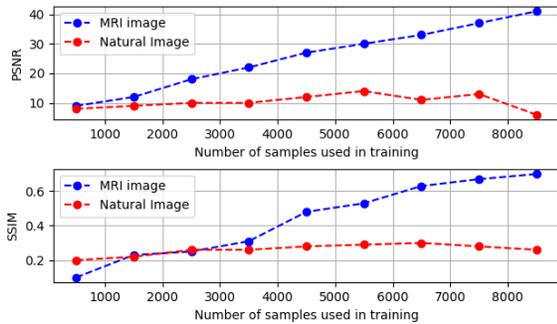


Fig. 5. Cross Evaluation of Proposed System.

D. Evaluation of System Performance

The matrices considered in the comparisons are PSNR and SSIM for quality of the image decompression, and space-saving for storage efficiency of the methods. Part a, b, c in the Figure 6 shows the PSNR, SSIM, and space-saving for different size images. The Proposed method outperforms the other non-fractal methods in all three scenarios. This shows that the suggested method can be a good replacement for the traditional algorithms in an application that requires a large amount of storage space such as Picture Archiving and Communication System(PACS).

Part d, e,f of the Figure 6 shows the comparison between the Machine Learning based Compression algorithms - GMM & Attention [32], iWave++ [31], Non-Local 3D-Context [30]. The proposed method is compared to the existing machine learning method on the basis of PSNR, SSIM and Space Saving. The comparison clearly shows that the proposed method is far superior to the existing techniques.

In Figure 7 the proposed method is compared to the existing fractal methods — Quadtrees [18], No-search [17], Genetic [22]. Part 1 shows that the proposed method outperforms other methods in execution time. So the proposed method solves one of the major disadvantages of the fractal compression, its compression time. Part b shows the proposed method is as good as the existing method in the case of preserving the structural similarity in the image.

VI. CONCLUSION

A time complexity reduction method in fractal image compression has been implemented using the deep reinforcement learning algorithm. While retaining the idea of Iterated function system of domain and range transformations, the search is confined based on a q-learning policy. Consequently, the encoder can compress the image with a better compression ratio and less execution time. This method differs from other recently proposed methods in the prediction of transformations, which doesn't include an exhaustive search. Since the cost to compute the transformations is high, the proposed method uses a neural network-based policy agent to predict the transformations. The empirical analysis shows that the proposed system can be a promising method in the area of medical image compression.

The proposed method identifies DRL as the key technology to reduce the encoding time in FIC. Even if the statistical learning strategy such as DRL is not a popular field in the area of image compression, the proposed work validates the significance of such strategies. This opens up a wide variety of possibilities to modify the image compression techniques with the policy grading algorithms. The user can change the policy grading algorithm depending on the type of images and can take advantage of the DRL based technique to have a better compression ratio with a much higher encoding speed. The Medical image archiving system can be revamped by modifying the existing image archiving system with the proposed DRL based FIC scheme to save the storage space, and hence the cost.

Our work opens up a new area of integrated Machine learning technique for fractal compression. The proposed method can be extended to video and audio compression to achieve a better compression ratio. In the case of video, each frame can be compressed on the basis of initial frame and frame to frame changes can be taken in count to achieve an efficient compression. The method can be used in the case of audio compression, because the number of similar patterns are very high and can be utilized to obtain the better transformation pairs.

REFERENCES

- [1] B. B. Mandelbrot and B. B. Mandelbrot, *The fractal geometry of nature*. WH freeman New York, 1982, vol. 1.
- [2] G. K. Wallace, "The JPEG still picture compression standard," *IEEE transactions on consumer electronics*, vol. 38, no. 1, pp. xviii–xxxiv, 1992.
- [3] P. Aguilera, "Comparison of different image compression formats," *Wisconsin College of Engineering, ECE*, vol. 533, 2006.
- [4] K. Arulkumar, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, 2017.
- [5] B. Varghese and S. Krishnakumar, "A novel fast fractal image compression based on reinforcement learning," *International Journal of Computational Vision and Robotics*, vol. 9, no. 6, pp. 559–568, 2019.
- [6] Y. Fisher, E. W. Jacobs, and R. D. Boss, "Fractal image compression using iterated transforms," pp. 35–61, 1992.
- [7] S. K. Mitra, C. A. Murthy, M. K. Kundu, and B. B. Bhattacharya, "Fractal image compression using iterated function system with probabilities," in *Proceedings International Conference on Information Technology: Coding and Computing*, 2001, pp. 191–195.
- [8] S. Liu, W. Bai, N. Zeng, and S. Wang, "A fast fractal based compression for MRI images," *IEEE Access*, vol. 7, pp. 62 412–62 420, 2019.
- [9] M. F. Barnsley, *Fractals everywhere*. Academic press, 2014.
- [10] A. E. Jacquin, "A fractal theory of iterated Markov operators with applications to digital image coding." 1990.
- [11] A. Jacquin, "Image coding based on a fractal theory of iterated contractive Markov operators, Part I: Theoretical Foundation," 1989.
- [12] Y. Fisher, *Fractal image compression: theory and application*. Springer Science & Business Media, 2012.
- [13] B. Hürtgen and C. Stiller, "Fast hierarchical codebook search for fractal coding of still images," in *Video Communications and PACS for Medical Applications*, vol. 1977, 1993, pp. 397–408.
- [14] D. Saupe, "Accelerating fractal image compression by multi-dimensional nearest neighbor search," in *Proceedings DCC'95 Data Compression Conference*, 1995, pp. 222–231.
- [15] Y. Tan and X. Zhou, "A novel speed-up algorithm of fractal image compression," in *International Workshop on Advanced Parallel Processing Technologies*, 2003, pp. 582–589.

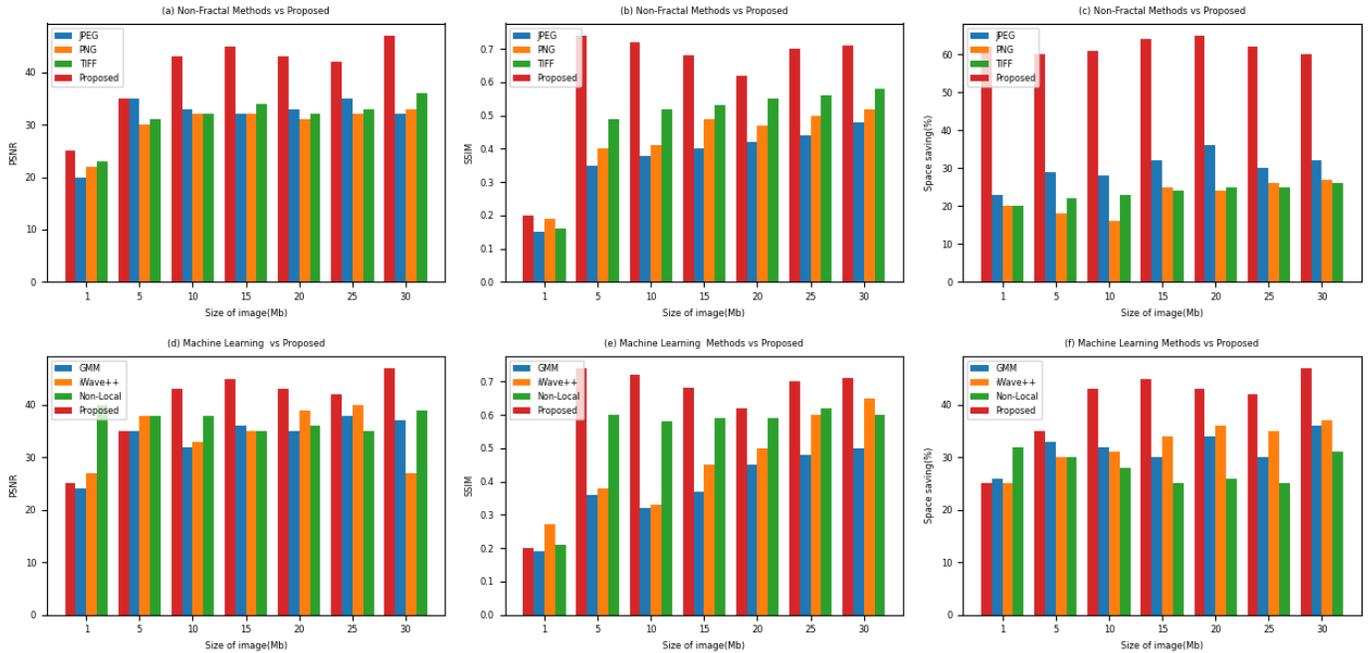


Fig. 6. (a)-(c) - Comparison of Proposed Method and Non-Fractal /Classical Compression Methods. (d)-(f) Comparison of Proposed Method and Machine Learning based Compression Methods.

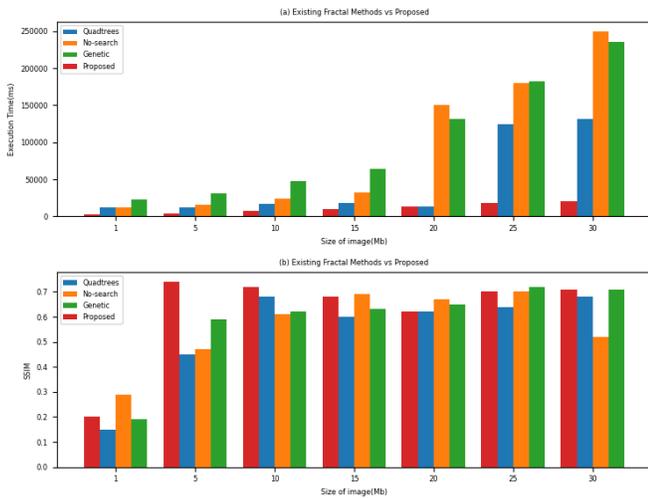


Fig. 7. Comparison of Proposed Method to the Existing Fractal Methods.

[16] X.-Y. Wang, Y.-X. Wang, and J.-J. Yun, "An improved no-search fractal image coding method based on a fitting plane," *Image and Vision Computing*, vol. 28, no. 8, pp. 1303–1308, 2010.

[17] S. Furoo and O. Hasegawa, "A fast no search fractal image coding method," *Signal Processing: Image Communication*, vol. 19, no. 5, pp. 393–404, 2004.

[18] Y. Fisher, "Fractal image compression with quadrees," pp. 55–77, 1995.

[19] V. Chaurasia and S. Sharma, "Similarity based kickoff method for fractal image compression," 2016.

[20] D. E. Golberg, "Genetic algorithms in search, optimization, and machine learning," *Addion wesley*, vol. 1989, no. 102, p. 36, 1989.

[21] J. Kennedy, "Encyclopedia of machine learning," *Particle Swarm Optimization (ed)*, pp. 760–766, 2011.

[22] M.-S. Wu and Y.-L. Lin, "Genetic algorithm with a hybrid select mechanism for fractal image compression," *Digital Signal Processing*, vol. 20, no. 4, pp. 1150–1161, 2010.

[23] J. Ballé, V. Laparra, and E. P. Simoncelli, "End-to-end optimization of nonlinear transform codes for perceptual quality," in *2016 Picture Coding Symposium (PCS)*, 2016, pp. 1–5.

[24] G. Toderici, S. M. O'Malley, S. J. Hwang, D. Vincent, D. Minnen, S. Baluja, M. Covell, and R. Sukthankar, "Variable rate image compression with recurrent neural networks," *arXiv preprint arXiv:1511.06085*, 2015.

[25] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[26] K. M. A. Okanohara.

[27] O. Rippel and L. Bourdev, "Real-time adaptive image compression," in *International Conference on Machine Learning*, 2017, pp. 2922–2930.

[28] M. Tschannen, E. Agustsson, and M. Lucic, "Deep generative models for distribution-preserving lossy compression," *arXiv preprint arXiv:1805.11057*, 2018.

[29] M. Sewak, "Policy-Based Reinforcement Learning Approaches," pp. 127–140, 2019.

[30] T. Chen, H. Liu, Z. Ma, Q. Shen, X. Cao, and Y. Wang, "Neural image compression via non-local attention optimization and improved context modeling," *arXiv preprint arXiv:1910.06244*, 2019.

[31] H. Ma, D. Liu, N. Yan, H. Li, and F. Wu, "End-to-End Optimized Versatile Image Compression With Wavelet-Like Transform," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[32] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized gaussian mixture likelihoods and attention modules," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7939–7948.