Intelligent Parameter Tuning Using Segmented Recursive Reinforcement Learning Algorithm

Modalavalasa Hari Krishna[#1], Dr.Makkena Madhavi Latha[*2]

[#]Research Scholar, Dept.of ECE, JNTUH-CEH, JNT University- Hyderabad

*Professor, Dept.of ECE, JNTUH-CEH,JNT University- Hyderabad
Hyderabad, Telangana, India.

Abstract *—Now-a-days, Machine Learning plays a vital role in enhancing the capabilities of traditional algorithms and processing techniques to handle profusion of data due to advances in digital technologies. Many processing problems can be solved using optimization-based solutions. In general, these solutions are normalized for different applications. Most of these solutions have control parameters to maintain relative importance to a specific application and also to optimize the performance of the solution. Parameter tuning is a straightforward task by manually determining the direction for adjustment of the parameter. But manual adjustment for these control parameters is tedious and consumes too much time and effort. Manual control becomes impractical if the solution has more parameters and requires precise tuning. With control being complex, machine learning has emerged as a key component in setting up correct and precise parameters. Reinforcement Learning (RL) can solve this problem but existing RL algorithms requires huge amount of learning time and resources. This paper aims at solving this problem and proposes novel Segmented and Recursive Reinforcement Learning (SRRL) algorithm to train the system that can automatically adjust the parameters accurately and precisely with minimal learning time. Performance of the proposing algorithm is validated in wavelet-based noise reduction technique by employing SRRL algorithm to adjust 3 control parameters of Noise based Hybrid Threshold method. After integrating with the proposing SRRL algorithm, the performance of the considered noise reduction technique is improved and provide better PSNR values with minimum learning time than with existing RL algorithms.*

Keywords *—Intelligent Parameter Tuning, Noise based Hybrid Threshold, Machine Learning, Reinforcement Learning, Segmented and Recursive Reinforcement Learning.*

I. INTRODUCTION

Machine Learning (ML) is a sub-field of Artificial Intelligence (AI) that provides systems the ability to automatically learn from data and improve from experience without being explicitly programmed by human beings. Machine learning allows software applications to become more accurate in predicting outcomes with continuous leaning. The basic premise of machine learning algorithm is to build model that can receive input data and use statistical analysis to predict an accurate output value within acceptable range.

In a class of Machine Learning algorithms, Supervised Machine Learning (SML) algorithms are very powerful and have many advantages but useful only for the applications with proper labelled data [4][8][14][15]. Data labelling is very complex and time-consuming process and also requires human efforts. Maintenance of labelled data is not possible for all applications. Unsupervised Machine Learning (UML) algorithms also require system response data prior to interact with the actual environment for learning phase [5][8][15]. These Unsupervised algorithms are useful for classification and segmentation applications. If the system response is not available prior to deployment of algorithms into real time application and the only way to collect information about the environment or application is interacting with real time environment, then one should use Reinforcement Machine Learning (RML or RL) algorithm. RL algorithms are the only option for the applications for which model of the environment is known but analytical solution is not available [6][9][11].

RL agent interacts with the real time environment during runtime in discrete instances. At each time instance **'t'**, the RL agent receives an observation $\boldsymbol{o_t}$ in initial state $\boldsymbol{s_t}$, which typically includes the reward$\boldsymbol{r_t}$,corresponding to that time instance t. It then chooses an action $\boldsymbol{a_t}$,from the set of available actions in solution space, which is subsequently sent to the environment. The environment moves to a new state $\boldsymbol{s_{t+1}}$ and the new reward $\boldsymbol{r_{t+1}}$,associated with the transition which is represented as a set $\boldsymbol{(s_t, a_t, s_{t+1})}$.The main goal of this reinforcement learning agent is to collect as much reward as possible by continuously interacting with the environment. The RL algorithms require more resources for the applications with larger solution space. Particularly, if the application requires the fine and precise solution then RL algorithms requires more memory and computational resources as well as huge amount of learning time. Many algorithms like Criterion of Optimality, State-Value Function, Brute Force, Value Function, Monte Carlo, Temporal Difference, Direct Policy search are developed for Reinforcement Learning [6][8][9]. But in all the above algorithms, the resource requirement rapidly increases with the increase in the solution space. Another important challenge for RL algorithms is finding the global maximum in the larger solution space with a greater number of local maxima. If the algorithm is modelled with care, it is possible to converge to the global optimum, this is the ideal behaviour that maximizes the reward but requires very high learning time. Moreover, real time problems are generally very modular, similar behaviours reappear often and modularity can be introduced to avoid learning everything all over again and again. Hierarchical approaches can solve this problem but doing this automatically is proving a challenge [1][2][6].

II. OBJECTIVES AND GOALS

A number of processing problems can be solved using optimization-based solutions [1]. In general, these solutions are normalized for wide range of applications to increase the adaptability of the solution. Many of these normalized solutions have adjustable control parameters to maintain relative importance to specific application. These control parameters also useful to performance tuning and optimization of the solution depending on the application. Parameter tuning is a straightforward and easy task for a human in case of limited number of control parameters by determining the direction for adjustment of the parameter [3][7]. Manual tuning is tedious and one has to navigate through entire solution space and identify the best optimal value. Setting up of the parameters can be done by human by verifying the quality of the output but the precision is not good enough for advanced algorithms and processing techniques. The Time to Control is another important constraint for design of advanced systems with processors of speed in range of few Giga Hertz. Manual control becomes impractical for the tuning of systems with more control parameters and Giga Hertz clocks [12][13].

Machine Learning has its applications in many areas like Analytics, Image Processing, Medical, Agriculture, Internet, Finance and marketing [8]. In this paper Image Processing is considered as application to validate the performance of proposing algorithm. Images play an important role in human life and vision is most important sense of human beings. Now-a-days, images are everywhere and generation of huge amount of image data also very easy for everyone due to advances in digital technologies [10]. With such a profusion of images, existing traditional image processing techniques are not capable to handle huge data, have to cope with more complex problems and have to face their adaptability according to human vision. Machine learning algorithms can be employed into Image Processing techniques to enhance the capabilities of the Image processing techniques. Both Image processing and Machine learning have their own excellences in their own fields. One can design more intelligent and sophisticated system by integrating the capabilities of ML and image processing in an intelligent way.

III. METHODOLOGY

Noise reduction is the one of the important phases in the Image Processing. Even though many noise reduction algorithms are developed, this are requiring further research to increase the performance [3]. Noise based Hybrid Threshold is the one of the novel techniques to handle the noise in the input image as well as the noise increased during the image enhancement phase [3].

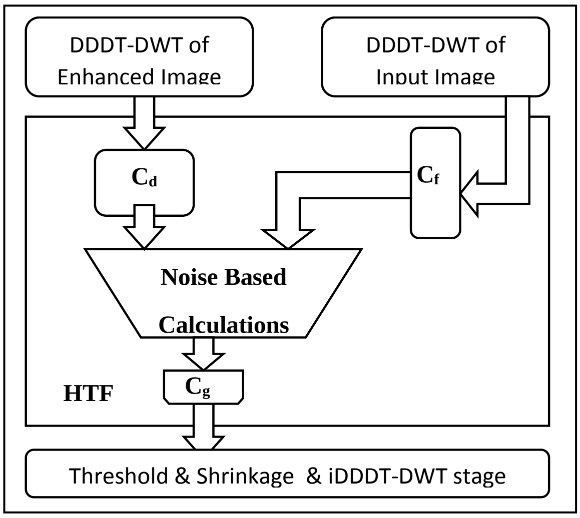

Fig1. Block diagram of Noise based Hybrid Threshold factor Calculation

As show in Fig1, this technique has three control parameters such as Direct Noise Controller (C_d), Feed through Noise Controller (C_f) and Noise Gain Controller (C_g). These parameters need to be adjusted preciously in order to optimize the performance of the technique. Previously these parameters were setup manually for particular application. To increase the adoptability of the technique, one should dynamically adjust the all the three control parameters during real time depending on the application, nature of image or noise levels. Manual tuning is not possible in this case as the parameters needs to be adjusted during run time with the speed of real time systems. Machine Learning algorithms can be incorporated into proposed hybrid threshold algorithm to determine the Hybrid Threshold Factor (HTF) dynamically depending on noise levels in the particular application/Image. Reinforcement learning algorithms can solve this problem and adjust the control parameters by interacting with the real time environment during the run time. For normal range of three control parameters, solution space is huge and requires a greater number of iterations to get the best reward. With the existing RL algorithms, it requires 1071 iterations to search the entire solution space and tune the parameters up to one decimal point precision. These algorithms are not suitable for real time application with time constraints as the learning time is large.

A. Proposed Method

In this paper, a novel reinforcement algorithm is proposed to handle the complexity of the problem and tune the control parameters more precisely with lesser learning time. The proposing Segmented Recursive Reinforcement Learning (SRRL) algorithm is segmented, maxima concentric and iterative approach to provide the fewer and nearly optimal solutions. It searches for global maximum region in initial stages and tunes the coefficients to optimal values in later stages. This SRRL algorithm searches in the entire possible solution space / natural range of coefficients for global maximum region. It can precisely tune the all three coefficient values up to 4 to 5 decimal points with the solution space within 124 solutions. This algorithm provides the solution which is closest to best/optimal solution and provides good noise reduction performance in terms of Peak Signal to Noise Ratio (PSNR). As the proposing SRRL algorithm provides optimal solution with in 124 iterations, the learning time is very less and it can meet the requirements of the real time applications.

IV. SIMULATION AND RESULTS

The existing RL algorithm and proposing SRRL algorithm are developed and simulated using MATLAB-R2019A software. The computational platform is a core-i7 workstation with 32GB RAM and 8GB GPU card. In this work, Lena image has been considered as source image. Parameter tuning performance of both existing RL and proposing SRRL algorithms are validated for Hybrid threshold-based noise reduction against input noisy image with three different noise types such as Gaussian, Speckle and Salt & Pepper noises and at different noise levels.

The existing RL algorithm learns and tunes all three control parameters in 1071 iterations. The effect of each control parameter on noise reduction performance can be observed in the parallel mapping diagram as shown in Fig2.

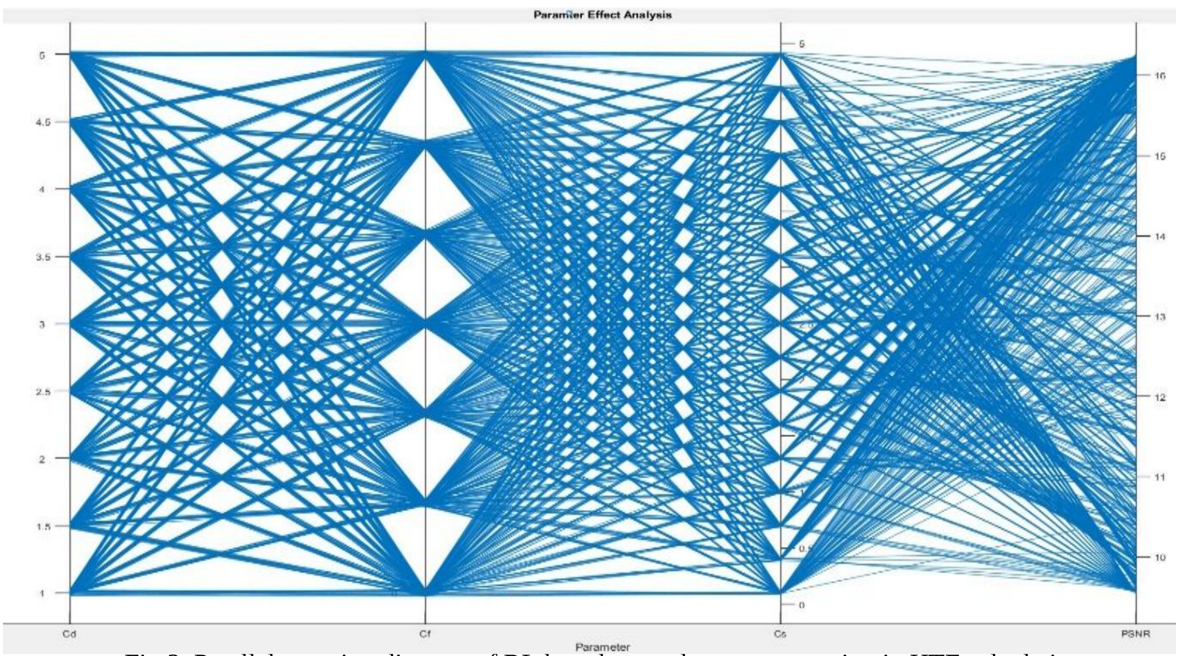

Fig 2. Parallel mapping diagram of RL based control parameter tuning in HTF calculation

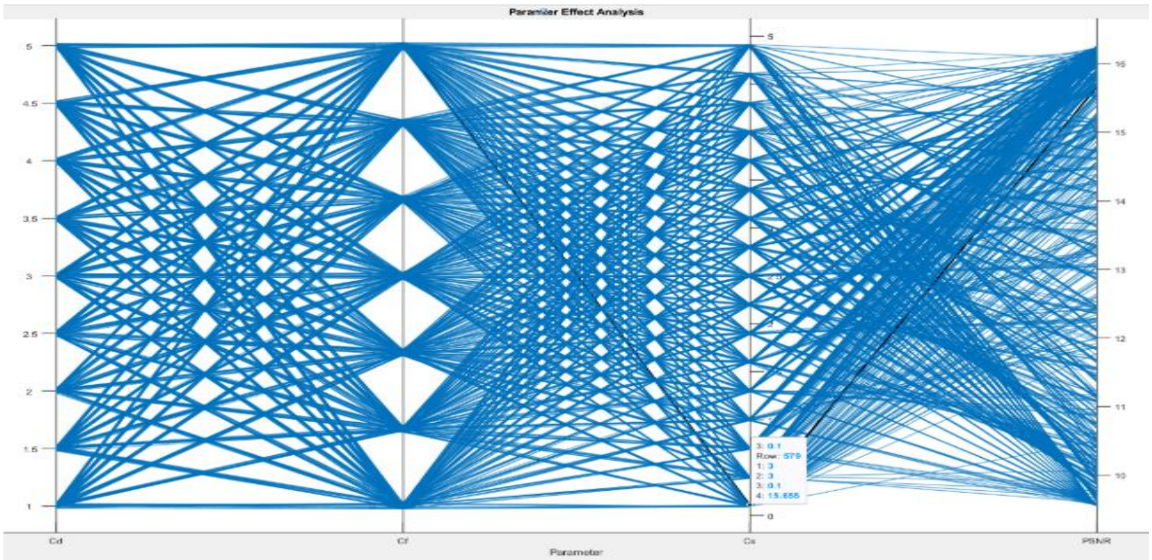

Fig 3.a Effect of C_f on PSNR in Parallel mapping diagram [observation-1]

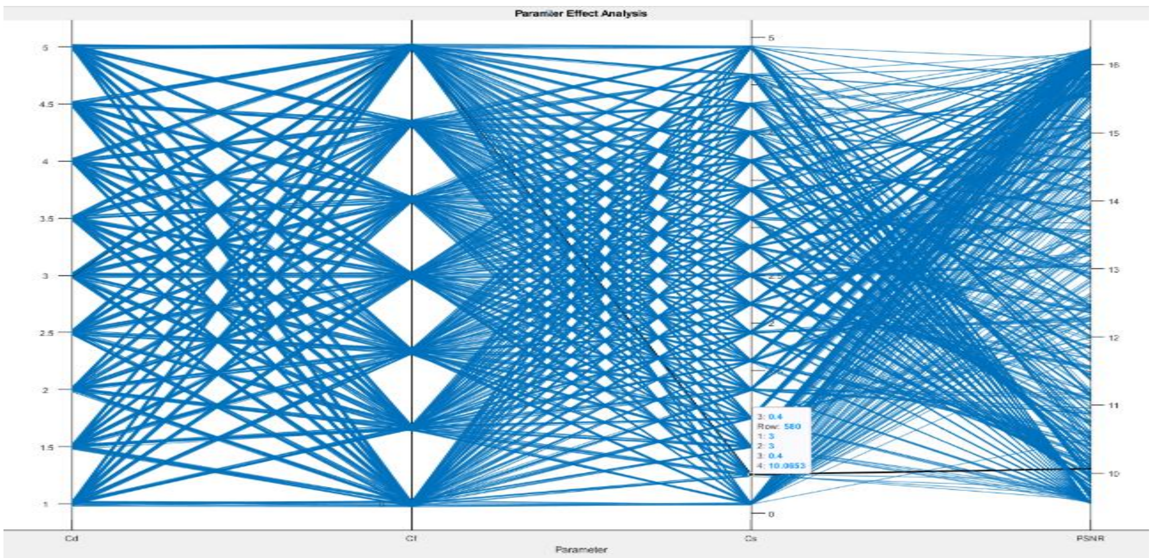

Fig 3.b Effect of $C_f$on PSNR in Parallel mapping diagram [observation-2]

The rapid effect of parameter C_f can be observed in the Fig 3.a &Fig3.b. By observing the C_f to PSNR mapping area, one can understand that the effect of Control parameter is not always straight forward. Existing RL algorithm searches entire solution space in uniform spacing manner and provides the maximum rewarding parameter values as optimum results. This type of learning is time consuming and almost predictable. The Fig4.a shows the predictability values of the individual parameter tuning and PSNR values using MATLAB inbuilt tracking algorithm. The solution space is divided into equal parts and parameters are tuned in regular intervals and tuning frequency can be observed from the Fig4.b.

The Proposing SRRL algorithm provides the optimal tuning within 124 iterations without compromising the noise reduction performance. The SRRL algorithms concentrate towards best reward and this can be observed from the parallel mapping diagram as shown in Fig 5. Most of the parameter lines mapped towards best PSNR value.

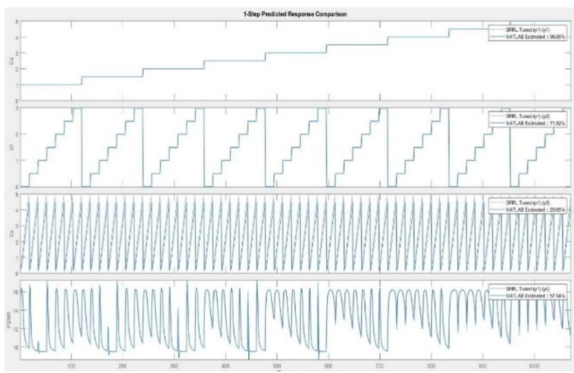

Fig4.a. Predictability of Parameter Tuning in RL algorithm.

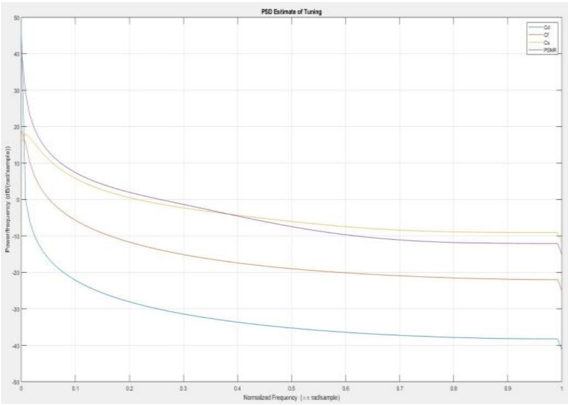

Fig4.b. Parameter tuning Frequency (Normalized) in RL algorithm.

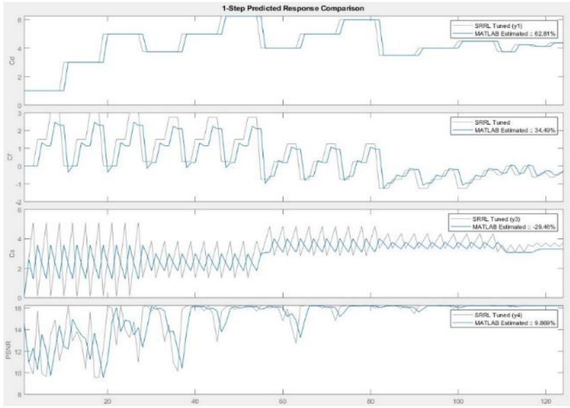

Fig6.a. Predictability of Parameter Tuning in SRRL algorithm.

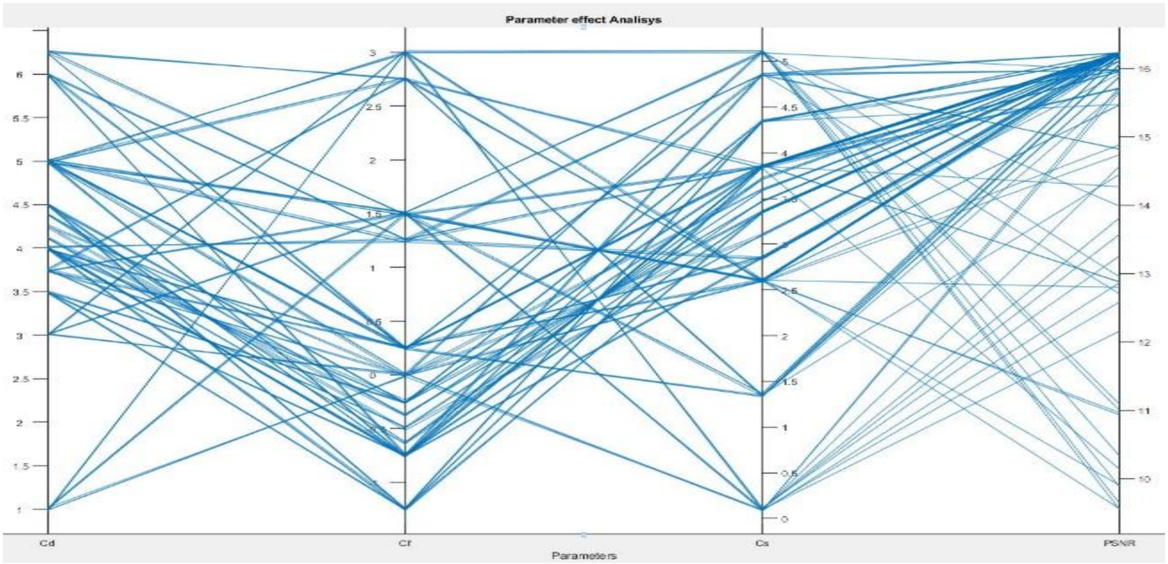

Fig 5. Parallel mapping diagram of SRRL based parameter tuning in HTF calculation

The tuning of SRRL algorithm is intelligent, understands the effect of parameters and tunes the parameters towards optimal value. So, the prediction of SRRL tuning is not easy and the predictability values are very less and shown in Fig6.a. The tuning is neither linear nor regular because this algorithm tunes parameters towards optimality instead of linear tuning and the normalized tuning frequency of the SRRL algorithm can be observed from Fig6.b. The tuning performance of the both the algorithms starts equally with rapid variations and in RL the variations are same until the end as the tuning is uniform but in SRRL the variations reduce segment by segment because the tuning is non-uniform and optimality concentric.

The tuning performances of both RL and SRRL algorithms are compared and validated in parameter tuning for Hybrid threshold factor calculation with input image with Gaussian noise of zero mean and 0.03 variance. After enhancement the PSNR of noisy

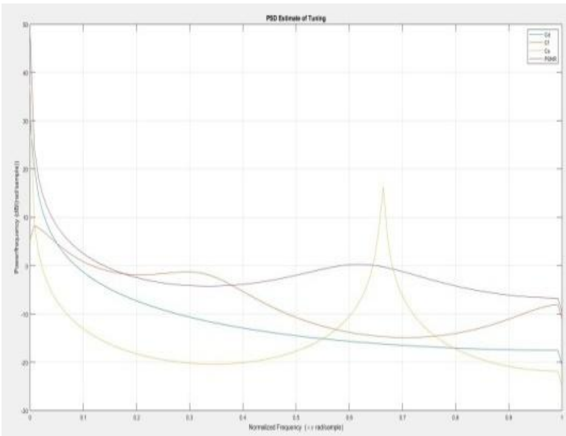

Fig6.b. Parameter tuning Frequency (Normalized) in SRRL algorithm.

input image reduces from 15.68 dB to 11.41 dB. After Double Density Dual Tree Discrete Wavelet Transform (DDDT-DWT) based noise reduction with Hybrid threshold technique using RL algorithm, The PSNR increased to 16.68dB in 1071 iterations.

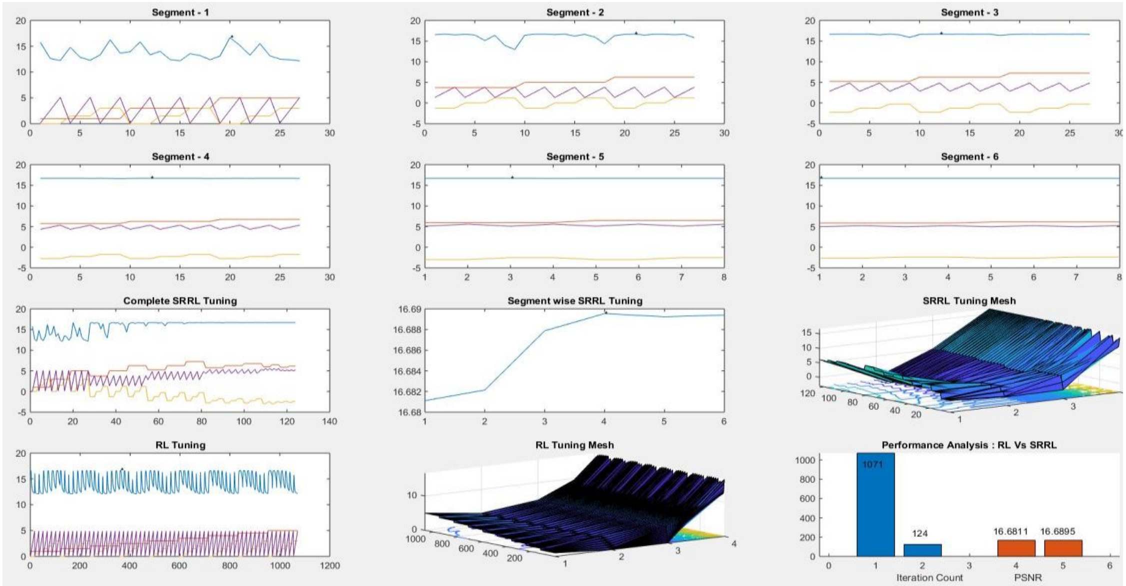

Fig 7. Comparison of parameter tuning using RL & SRRL with Gaussian noisy image input

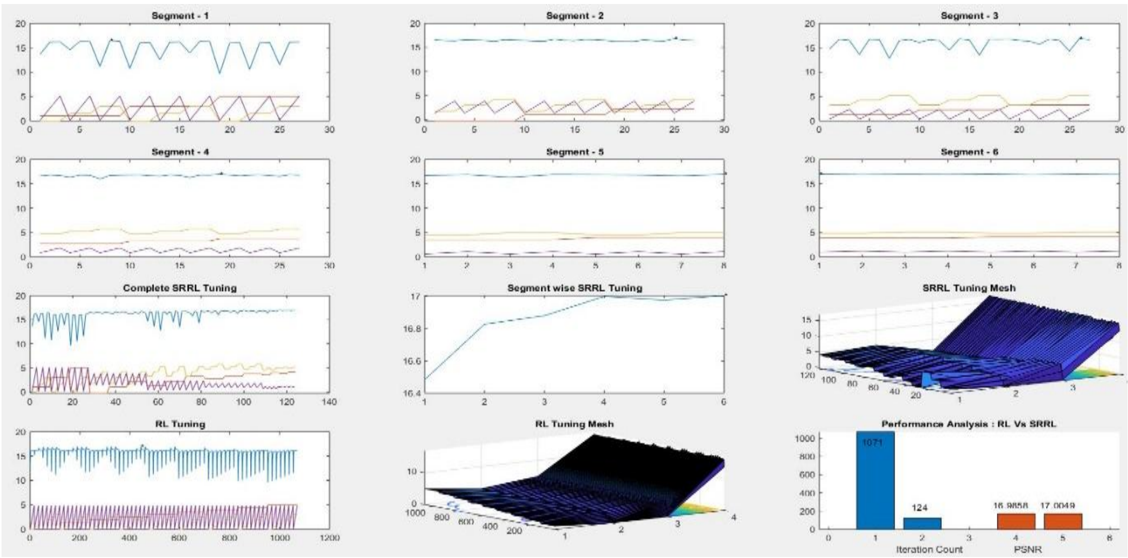

Fig 8. Comparison of parameter tuning using RL and SRRL with Salt& Pepper noisy image input.

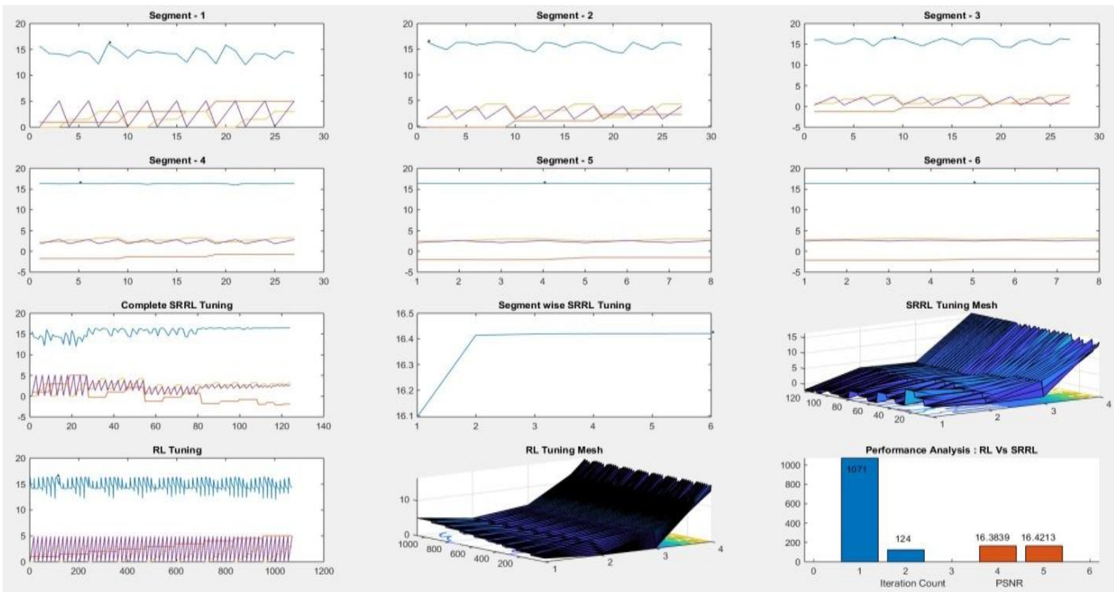

Fig 9. Comparison of parameter tuning using RL and SRRL with Speckle noisy image input.

Table-I. Noise reduction Performance comparative analysis table

Stage \ Noise Type	Gaussian Noise		Salt & Pepper Noise		Speckle Noise	
Noise Level (Variance)	0.03	0.3	0.03	0.3	0.03	0.3
Input Image (PSNR)	15.68	8.71	20.54	10.53	20.94	17.08
Enhanced Image (PSNR)	11.41	8.61	15.94	10.21	13.66	11.93
RL Output (PSNR)	16.68	16.26	16.98	16.49	16.38	15.64
SRRL Output (PSNR)	16.69	16.24	17.01	16.51	16.42	15.64

In the same noise reduction technique with SRRL algorithm the PSNR increased from 11.41dB to 16.69dB with in 124 iterations. The segment wise tuning and overall comparative analysis are shown in Fig 7. The tuning performances of both RL and SRRL algorithms are also compared and validated with input image with Salt and Pepper noise of zero mean and 0.03 variance. After enhancement the PSNR of noisy input image reduces from 20.54 dB to 15.54 dB. After DDDT-DWT based noise reduction with Hybrid threshold technique using RL algorithm, The PSNR increased to 16.98dB in 1071 iterations. In the same noise reduction technique with SRRL algorithm the PSNR increased from 15.54dB to 17.01dB with in 124 iterations. The segment wise SRRL tuning and overall comparative analysis are shown in Fig8.

Parameter tuning performances of both algorithms are also validated with input image with Speckle noise of 0 mean and 0.03 variance. After enhancement the PSNR of noisy input image reduces from 20.94 dB to 13.66 dB. After DDDT-DWT based noise reduction with Hybrid threshold technique using RL algorithm, the PSNR increased to 16.38dB in 1071 iterations. In the same noise reduction technique with SRRL algorithm the PSNR increased from 13.66dB to 16.42dB with in 124 iterations. The segment wise SRRL tuning and overall comparative analysis between RL and SRRL are shown in Fig9.

Table 1 shows the comparative analysis between the noise reduction performance of Hybrid Threshold -DDDT-DWT based technique with RL control parameter tuning and that of Hybrid Threshold - DDDT-DWT based technique with SRRL control parameter tuning. Gaussian, Salt & Pepper and Speckle noise models are considered in input image with zero mean and different variances such as 0.03 and 0.3. In all the cases the SRRL algorithm tunes the all 3 control parameters to Global optimum values and provides the best performance in terms of PSNR within 124 iterations.

V. CONCLUSION

The proposed Segmented Recursive Reinforcement Learning algorithm is developed, simulated and its parameter tuning performance is validated against the existing RL algorithm. DDDT-DWT based noise reduction technique with Hybrid Threshold is considered for performance comparison of both the algorithms. Existing RL algorithm tuned the all 3 control parameters of Hybrid threshold to their optimal values with precision of one decimal point after 1071 iterations. Proposed SRRL algorithm tuned all the three parameters to their optimal values with a precision of 3 to 4 decimal points within 124 iterations and provides global maximum PSNR as optimal value. The parameter tuning performance of proposed SRRL algorithm is validated against different noise models and different noise levels. In all the cases the SRRL algorithm tunes control parameters to their précised optimal values with less learning time.

REFERENCES

[1] Chenyang Shen, Yesenia Gonzalez, Liyuan Chen, Steve B. Jiang, Xun Jia, *"Intelligent Parameter Tuning in Optimization-based Iterative CT Reconstruction via Deep Reinforcement Learning"*, IEEE Transactions on Medical Imaging, doi: 10.1109/TMI.2018.2823679, 2018.

[2] Juan Cruz Barsce, Jorge A. Palombarini, Ernesto C. Mart´ınez , "*Towards Autonomous Reinforcement Learning: Automatic Setting of Hyper-parameters using Bayesian Optimization"*, doi : 978-1-5386-3057-0/17,IEEE,2017.

[3] M. Hari Krishna, G. Sateesh Kumar, "*A Noise Based Hybrid Thresholding For Enhanced Noise Reduction Using Double Density Dual Tree Complex Wavelet Transform*", 13th International Conference on Electromagnetic Interference and Compatibility (INCEMIC), doi : 978-1-5090-5350-6/15, IEEE,2015.

[4] R. Saravanan, Pothula Sujatha, "*A State of Art Techniques on Machine Learning Algorithms: A Perspective of Supervised Learning Approaches in Data Classification*", Second International Conference on Intelligent Computing and Control Systems (ICICCS-2018), IEEE, doi: 10.1109/ICCONS.2018.8663155,2018.

[5] Ahmad Maroof Karimi, Justin S. Fada, JiQi Liu, Jennifer L. Braid, Mehmet Koyutürk, Roger H. French, *"Feature Extraction, Supervised and Unsupervised Machine Learning Classification of PV Cell Electroluminescence Images*",

IEEE 7th World Conference on Photovoltaic Energy Conversion (WCPEC)-2018, IEEE Conferences, doi: 10.1109/PVSC. 2018.8547739, PP: 0418 – 04ADAPTIVE,2018.

[6] Snehasis Mukhopadhyay, Omkar Tilak, Subir Chakrabarti, "*Reinforcement Learning Algorithms for Uncertain, Dynamic, Zero-Sum Games*", 17th IEEE International Conference on Machine Learning and Applications (ICMLA)-2018, IEEE Conferences, doi: 10.1109/ICMLA.2018.00015, PP: 48 – 54,2018.

[7] L. I. Rudin, S. Osher, and E. Fatemi, "*Nonlinear total variation based noise removal algorithms*", Phys. D, Nonlinear Phenomena, vol. 60, nos. 1–4, pp. 259–268, Nov. 1992.

[8] Michael G. Pecht, Myeongsu Kang, "*Prognostics and Health Management of Electronics: Fundamentals, Machine Learning, and the Internet of Things"*, IEEE Books, Wiley-IEEE Press, doi: 10.1002/9781119515326, e-ISBN: 9781119515326.

[9] Parag Kulkarni, *"Reinforcement and Systemic Machine Learning for Decision Making*", IEEE Books, Wiley-IEEE Press, doi: 10.1002/9781118266502, e-ISBN: 9781118266502, 2012.

[10] Oge Marques, *"Practical Image and Video Processing Using MATLAB"*, IEEE Books, Wiley-IEEE Press, doi: 10.1002/9781118093467, e- ISBN: 9781118093467, 2011.

[11] M.Waltz, K.Fu, "*A heuristic approach to reinforcement learning control systems"* IEEE Transactions on Automatic Control, DOI: 10.1109/TAC.1965.1098193, volume-10, Issue-4 , 1965.

[12] [12] V. Solo, "*Selection of tuning parameters for support vector machines,"* Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005., Philadelphia, PA, 2005, pp. v/237-v/240 Vol. 5.

[13] E. B. Kosmatopoulos and A. Kouvelas, *"Large Scale Nonlinear Control System Fine-Tuning Through Learning*," in IEEE Transactions on Neural Networks, vol. 20, no. 6, pp. 1009-1023, June 2009.

[14] Jun Yu; Dacheng Tao, "*Modern Machine Learning Techniques*," in Modern Machine Learning Techniques and Their Applications in Cartoon Animation Research , , IEEE, 2013, pp.63-104.

[15] John D. Kelleher; Brendan Tierney, *"4 MACHINE LEARNING 101*," in Data Science, MITP, 2018, pp.97-150