An Optimized Framework of Video Compression Using Deep Convolutional Neural Networks (DCNN)

Dr.M.Sreelatha[1], Dr.R.Lakshmi Tulasi[2] , Mr.K.Siva Kumar[3]

Professor & Head, CSE department, R.V.R & J.C College of Engineering, Guntur,
Andhra Pradesh, India. lathamoturicse@gmail.com

Professor, CSE department, R.V.R & J.C College of Engineering, Guntur,
Andhra Pradesh, India. rltulasi.2002@gmail.com

Assistant Professor,CSE department, R.V.R & J.C College of Engineering, Guntur,
Andhra Pradesh, India. kommerlasivakumar@gmail.com

Abstract: Video streaming demand has risen significantly in the modern world and now accounts for a significant portion of internet traffic, making it a difficult job for service providers to stream videos at high speeds while using fewer storage spaces. The existing video compression prototypes necessitate non learning based designs in order to follow inefficient analytical coding design. As a result, we propose a DCNN technique for obtaining optimal set of frames by relating each frame pixel with preceding and subsequent frames, then identifying related blocks and reducing unnecessary pixels by incorporates OFE-Net, MVE-Net, MVD-Net, MC-Net, RE-Net, and RD-Net. The proposed DCNN technique generates high video quality at low bit rates with respect to MSSIM and PSNR.

Keywords:
Deep neural networks, ***E***ncoding, Decoding, Video Compression.

1. Introduction

People who watch videos on the internet are about 90%, this is expected to rise in the near future. As a result, an effective video compression model is required to deliver higher-quality frames while using less bandwidth.

Video codecs rely on hand-drawn models to compress videos. The existing models are not optimized, despite their excellent design. By optimizing the overall codec model, the video compression process can be improved even more.

Video compression using Deep Neural Networks has outperformed traditional image codecs such as the Joint Photographic Experts Group. End-to-end training is required for deep neural network-based models, which rely on extremely nonlinear transformation.

Manuscript received May 5, 2022
Manuscript revised May 20, 2022
https://doi.org/10.22937/IJCSNS.2022.22.5.71

Building a model employing various video compression algorithms is not a simple undertaking. The most significant aspect is motion estimation, which generates and compresses motion data. The process of video compression relies heavily on motion information to eliminate temporal redundancy. To express motion vectors, the only approach is to utilize an optical flow net. Although learning-based optical flow estimation focuses on generating precise flow information, correct optical flow isn't always the best option for specific video jobs. Furthermore, the ability of optical flow data is higher than existing models directly compressing optical flow value using existing methods will result in high bit rate information.

The goal of reducing rate distortion is to provide more quality reconstructed frames at a given bit rate. It is required for video compression to work properly.

In order to achieve benefits of end-to-end training for deep learning-based video compression models, rate-distortion must be reduced. The following are the primary advantages of this model: Deep neural networks are used to implement all of the phases in the DCNN model. All of the steps in the DCNN model are reliant on rate distortion and are integrated using a single loss function, resulting in a high compression ratio. Research persons working on computer vision, video compression, and deep model construction will benefit from this study.

2. Related Work

In [1], the video compression task can be categorized in to three types. They are - the Classical Era,the era of Generic Heuristics and the era of modern techniques with Deep Learning. By the detailed

study of the literature through the past decades it is learned that various schemes have been proposed towards the video compression. These schemes have contributed a lot of efficient mechanisms in different ways. However, further improvements are also needed towards the same pertaining to the limitations observed as specified.

In [2], illustrate and explain various issues for video compression processin the field of DNNs. Still the additional investigation is look for to achive the upcoming generation and neural networks-basedcodec's.

In [3], has presented a deep netwrok with fast and light weight model for optical flow process. Previous pyramid feature is replaced with U-shaped network and this model obtains better results. And this model can help computer vision applications.

In [4], described a optical flow approach which provides the features of deep learning based optical flow algorithms. This approach gives the better accuracyresults compared with an existing method and surpassing it in several benchmarks.

In [5], The MEMC (Estimation and Compensation of Motion) neural network is proposed for learning and improving video frame interpolation. This model takes advantage of the MEMC framework's capabilities for managing massive amounts of motion data, as well as learning-based methods for extracting features quickly. Many video enhancement activitiescan be done with this MEMC framework. The qualitative and quantitative evaluation of these methods against state-of-the-art video interpolation and improvement algorithms on various standard data sets demonstrates that they outperform them.

In [6], describes video compression framework based on deep learning which provides MV and RP network. Here the experiment results shown that MV and RP network be able to improve performance of compression by modeling spatial correlations among the frames accurately.

In [7], present anefficient video compression framework based on deep learning. Here comparison of x264, withthis Deep Coder has shownby similar type of codingefficiency (lossy) with the familiar testing series used by the video coding society and video compressionon deep learning is an alternative framework of the process of video coding in feature.

In [8], propose PMCNN and modeled spatio temporal to achieve predictive based coding and learning-based framework of effective process for video compression is explored. Even though lack of entropy-based coding and still this achieves a better result for video compression, exhibiting new attainable handling of video compression.

In [9] ,presented a nonlinear transform coding based image compression method and a framework to optimize it end-to-end for rate–distortion performance. Nevertheless, additional visual improvements might be possible in terms of perceptual metric like MSE,if the method were optimized.

In[10], provides a variational auto encoder-based image compression trainable model. When evaluating rate–distortion performance using a traditional metric based on squared error, this model leads to picture compression when using the MS-SSIM index, and it outperforms ANN-based techniques when using a traditional metric based on squared error (PSNR).

3. Proposed Methodology

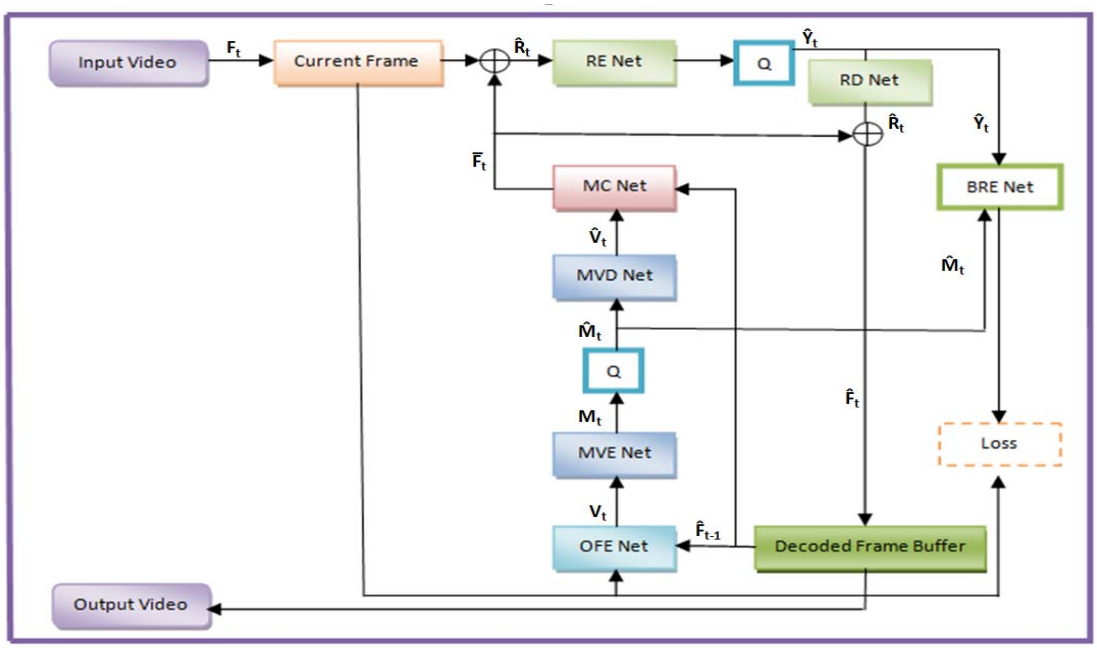

Figure 1: DCNN Framework.

Introducing the symbols: Assume $\mathbf{V} = \{\mathbf{F_1}, \mathbf{F_2}, ...\mathbf{F_{t-1}}, \mathbf{F_t}, ...\}$ represents the sequences of current video, and at time step **t**, $\mathbf{F_t}$ isframe. The symbols $\bar{\mathbf{F}}_\mathbf{t}$ and $\hat{\mathbf{F}}_\mathbf{t}$ representspredicted frameandreconstructed or decoded frame. The residual information or error information between original frame $\mathbf{F_t}$ and predicted frame $\bar{\mathbf{F}}_\mathbf{t}$ is $\mathbf{R_t}$. The reconstructed (decoded residual)information is denoted by $\hat{\mathbf{R}}_\mathbf{t}$. In orderly, motion informationis essential to reduce the temporal redundancy. Among them, the optical flowor motion vector value represents with $\mathbf{V_t}$ and its corresponding reconstructed form is $\hat{\mathbf{V}}_\mathbf{t}$. To improve the compression efficiency, either a linear transform or nonlinear transform techniques can be used. Consequently,residual information $\mathbf{R_t}$ is converted to $\mathbf{Y_t}$, motion information $\mathbf{V_t}$converted to $\mathbf{M_t}$, corresponding quantized versions $\hat{\mathbf{R}}_\mathbf{t}$ and $\hat{\mathbf{M}}_\mathbf{t}$ respectively.

Step 1 - Motion estimation andCompression:

The MVE-MVD net is proposed for compressing and decoding optical flow values. A sequence of convolution and nonlinear-transform procedures are used to extract or provide the optical flow $\mathbf{V_t}$. In this example, there are a total of 128 output channels for convolution (deconvolution), which equals 2. Given an optical flow $\mathbf{V_t}$ of size MxNx2, the MVE net will generates the motion information representation $\mathbf{M_t}$ of size M/16xN/16x128. After then, $\hat{\mathbf{M}}_\mathbf{t}$ is quantized to $\mathbf{M_t}$. The MVD net obtains the quantized representation, which is subsequently used to reconstruct the motion information $\hat{\mathbf{V}}\mathbf{t}$. Entropy coding will also be done using the quantized representation $\hat{\mathbf{M}}_\mathbf{t}$.

Step 2 - MC Net:

The motion compensation network obtains predicted frame $\bar{\mathbf{F}}_\mathbf{t}$, which is near to current frame $\mathbf{F_t}$ as possible, using both previously reconstructed frame $\hat{\mathbf{F}}_{\mathbf{t-1}}$and motion vector $\hat{\mathbf{V}}_\mathbf{t}$. At beginning, previous frame $\hat{\mathbf{F}}_{\mathbf{t-1}}$warped to present frame using motion information $\hat{\mathbf{V}}_\mathbf{t}$. However, there are still artefacts in warped frame. To remove these artefacts, we send warped frame $\mathbf{W}$ $(\hat{\mathbf{F}}_{\mathbf{t-1}}, \mathbf{V_t})$, reference frame $\hat{\mathbf{F}}_{\mathbf{t-1}}$, and the motion vector $\hat{\mathbf{V}}_\mathbf{t}$ into another CNN, which produces the refined predicted frame $\bar{\mathbf{F}}_\mathbf{t}$. The proposed method follows pixel-based motion compensation strategy that give more precise temporal information.

Step 3 – RE net and RD net:

Using both the prior reconstructed frame and the motion vector $(\hat{F}_{t-1}, \hat{V}_t)$, motion compensation network gets predicted frame $\bar{F}_t$, which is as close to the present frame F_t as possible. First, the motion information $\hat{V}_t$ is used to warp previous frame $\hat{F}_{t-1}$ to the current frame. The distorted frame, however, still contains artefacts. We send the warped frame $W(\hat{F}_{t-1}, V_t)$, a reference frame $\hat{F}_{t-1}$, and motion vector $\hat{V}_t$ into another CNN to obtain refined predicted frame $\bar{F}_t$. The solution is pixel-by-pixel motion compensation strategy that can provide more precise temporal information.

Step 4 - Entropy coding:

The quantized motion information $\hat{M}_t$ from Step-1 and residual information $\hat{Y}_t$ from Step-3 are coded into bits and transmitted to the decoder during the testing stage. From the training stage, by employing CNNs for number of bits cost are estimated (BRE Net in Figure) and subsequently to acquireprobability distribution of each symbol in $\hat{M}_t$ and $\hat{Y}_t$.

Step 5 - Frame reconstruction:

By adding $\bar{F}_t$ in Step 2 and$\hat{R}_t$ in Step3, obtains the reconstructed frame $\hat{F}_t$,i.e. $\hat{F}_t = \bar{F}_t + \hat{R}_t$. At Step 1, for motion estimation, reconstructed frame will be used by $(t+1)^{th}$ frame. For the decoder, from step-4, bits providingthroughencoder, from step-2, motion compensation,from step-3, the quantized frameand then obtain reconstructed frame $\hat{F}_t$ at step-5.

4. Training Strategy

4.1 Loss Function: Our framework's major goal is to use fewest number of bits possible while encoding video, while also distortion is reducing between F_t (original input frame) and $\hat{F}_t$ (final output frame) (reconstructed frame). An optimal rate-distortion problem is proposed for this:

$$\lambda D + R = \lambda d(F_t, \hat{F}_t) + (H(\hat{M}_t) + H(\hat{Y}_t)), \quad (1)$$

In our approach, we utilize mean MSE and $d(F_t, \hat{F}_t)$ to represent distortion between F_t and $\hat{F}_t$. The number of bits used for encoding is represented by H (·). Both residual data $\hat{Y}_t$ and motion data $\hat{M}_t$ should be encoded into bit streams in this case. The Lagrange multiplier (λ), which affects the number of bits versus distortion trade-off. The loss function's inputs are reconstructed frame $\hat{F}_t$,,original frame F_t, and estimated bits, as indicated in the figure.

4.2 Quantization: The residual information Y_t and motion information M_t must and should be quantized before the process of entropy coding. Use the optimal approach here, and then replace the quantization operation in training stage with adding noise uniformly using the optimized method. Take Y_t as an example: in training step, quantized information $\hat{Y}_t$ is adding uniform noise to Y_t. ,

i.e., $\hat{Y}_t = Y_t + \eta$, (2)

Where η represents uniform noise.

In next level, rounding function directly apply,

i.e., $\hat{Y}_t = \text{round}(Y_t)$. (3)

4.3 Bit Rate Estimation:

Here, the complete network is optimize for both bit rate and distortion, needs toobtain bit rate $(H(\hat{Y}_t), H(\hat{M}_t))$ of generated latent representations $(\hat{Y}_t, \hat{M}_t)$. The entropy of the relevant representation symbolsis the accurate measure for bit rate. Therefore, the probability distributions of $\hat{Y}_t$ and $\hat{M}_t$ are estimate and then obtaining the corresponding entropy. Here, for estimation of thedistributions we use the CNNs.

4.4 Frame Buffer: In both ME and MC networks, the previously rebuilt frame $\hat{F}_{t-1}$ is necessary while compressing the current frame, as shown in Figure. As a result, the prior reconstructed frame$\hat{F}_{t-1}$ is network output ofprevious constructed frame, which is primarily depends on reconstructed frame $\hat{F}_{t-2}$, same as remaining procedure. We'll do on-the-fly updating here, with each iteration being saved in a buffer. When encoding F_{t+1}, $\hat{F}_t$ in the buffer will be helpfull for motion estimate and compensation via iterations. As a result, for each trained sample in the buffer, the epoch is updated. Furthermore, one frame for a video clip can be optimized and stored for each repetition, which is more efficient.

5. Results

The BDBR calculated by MS-SSIM in Table 1 and The BDBR calculated by PSNR in Table 2 and getting better results compared with the various learning methods.

Dataset	Video	DVC 11	Cheng 13	Habibian 14	HLVC 12	Proposed DCNN
UVG	Beauty	−14.85	-	−44.63	−41.39	27.87
	Bosphorus	10.03	-	−13.77	−51.22	28.09
	HoneyBee	−21.63	-	−4.13	−42.87	21.61
	Jockey	104.82	-	56.38	6.97	20.83
	ReadySetGo	2.77	-	89.06	−7.32	25.03
	ShakeNDry	−20.94	-	−35.10	−32.82	26.70
	YachtRide	−3.83	-	−21.85	−42.17	23.08
	Average	**8.05**	**-**	**3.71**	**−30.12**	**25.17**
JCT-VC Class B	BasketballDrive	15.47	-	-	−34.98	22.82
	BQTerrace	15.08	-	-	−22.52	24.03
	Cactus	−21.40	-	-	−43.63	25.09
	Kimono	−2.67	-	-	−46.79	24.69
	ParkScene	−20.17	-	-	−39.31	28.29
	Average	**−2.74**	**-**	**-**	**−37.44**	**24.98**
JCT-VC Class C	BasketballDrill	5.54	17.97	-	−18.45	27.86
	BQMall	4.84	−38.59	-	−20.33	23.97
	PartyScene	−23.60	−6.53	-	−30.29	21.67
	RaceHorses (480p)	−14.29	41.07	-	−25.45	15.61
	Average	**−6.88**	**3.48**	**-**	**−23.63**	**17.82**
JCT-VC Class D	BasketballPass	0.67	−44.96	-	−36.24	31.56
	BlowingBubbles	−29.38	−22.92	-	−39.84	24.06
	BQSquare	−25.50	−39.60	-	−97.56	24.50
	RaceHorses (240p)	−19.82	12.6	-	−36.59	14.93
	Average	**−18.51**	**−23.72**	**-**	**−52.56**	**23.76**
Average on all videos		**−2.94**	**-**	**-**	**−35.14**	**22.93**

Table 1: The BDBR calculated by MS-SSIM

Dataset	Video	DVC 11	HLVC 12	Proposed DCNN
UVG	Beauty	−39.63	−48.48	0.87
	Bosphorus	17.57	−23.16	0.90
	HoneyBee	24.53	−26.63	0.81
	Jockey	90.02	105.21	0.78
	ReadySetGo	9.03	26.69	0.89
	ShakeNDry	−25.07	−26.88	-ve
	YachtRide	−14.19	−16.34	-ve
	Average	**8.89**	**−1.37**	**0.69**
JCT-VC Class B	BasketballDrive	35.24	13.21	-ve

	BQTerrace	2.28	−4.56	0.90
	Cactus	−5.19	−29.09	-ve
	Kimono	−10.79	−18.71	0.84
	ParkScene	−11.63	−19.59	-ve
	Average	**1.98**	**−11.75**	**0.30**
JCT-VC Class C	BasketballDrill	18.03	−3.67	0.90
	BQMall	62.28	13.68	0.88
	PartyScene	8.61	2.08	0.89
	RaceHorses (480p)	14.61	19.25	-ve
	Average	**25.88**	**7.83**	**0.66**
JCT-VC Class D	BasketballPass	42.34	−3.44	-ve
	BlowingBubbles	−12.15	−19.19	-ve
	BQSquare	22.01	−19.10	0.90
	RaceHorses (240p)	9.18	−8.55	-ve
	Average	15.34	−12.57	**0.2**
Average on all videos		**11.85**	**−4.36**	**0.46**

Table 2: The BDBR calculated by PSNR

We compared our proposed technique to the learning-based video codecs [11,12,13,14] in Table 1 in terms of MS-SSIM and [11,12] in terms of PSNR. When tested by both MSSIM and PSNR, our technique outperforms H.264 (UVG dataset & JCT-VC dataset) Fig: 2(a) & (b) and Fig: 2(c) & (d). Meanwhile, when compared to the following tables with respect to PSNR and MS-SSIM, our technique provides equivalent or superior compression performance.

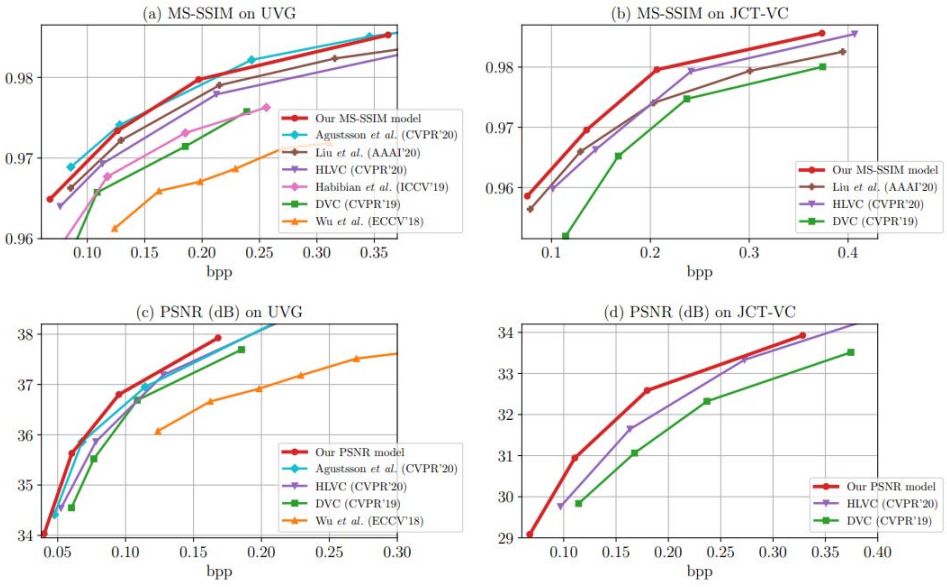

Fig: 2(a) & 2(b): Performance of proposed model based on MS-SSIM

Fig: 2(c)&(d): Performance of proposed model based on PSNR

6. Training and Testing

Our proposed DCNN technique trained the vimeo 90k dataset [15] which is recently built for evaluating different video processing tasks. such as video denoising and video super-resolution. It consists of 89,800 independent clips that are different from each other in context. The whole system is implemented based on Tensor flow and it takes about 7 days to train the whole network using gpu and tested the Ultra-Video-Group dataset (UVG), [16] and Joint Collaborative Team –Video Coding dataset (JCT-VC), [17].

7. Conclusion

We propose, End-to-End DCNN-based efficient framework ofvideo compression in this paper. In this case, the proposed framework combines the benefits of standard and deep neural network-based models. And we show how our DCNN technique out-performs both widely used classical video compression standards and more present deep learning-based video compression solution. Because our proposed DCNN model improves video quality while using low bit rates, it has a higher compression ratio and lower error rates.

References:

[1].K.Siva Kumar, Dr.K.Janaki, ”A Review on Video Compression Approaches and Utilization of Deep Learning Techniques”, International Journal of Advanced Research in Engineering and Technology (IJARET), IAEMA Publication, 1211-1218, Volume 11, Issue 9, 2020.

[2].Raz Birman,Yoram Segal &OferHadar, “Overview of Research in the field of Video Compression using Deep Neural Networks “, 79:11699–11722, Springer, Multimedia Tools and Applications (2020) .

[3].A. Ranjan and M. J. Black. Optical flow estimation using aspatial pyramid network. In CVPR, volume 2, page 2. IEEE,2017.

[4].Jingjing Dai,Oscar C. Au, WenYang, Chao Pang, Feng Zou and Yu Liu,” Motion VectorCoding based on Predictor Selectionand Boundary-matching Estimation”, IEEE International Workshop on Multimedia Signal Processing, 2009.

[5].Wenbo Bao, Wei-Sheng Lai, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang, “MEMC-Net: Motion Estimation and Motion Compensation Driven Neural Network for Video Interpolation and Enhancement”, Volume: 43 Issue: 3, IEEE Transactions on Pattern, 2021.

[6].XiangJiWu ,Ziwen Zhang , Jie Feng , Lei Zhou , Junmin Wu, Tucodec Inc,” End-to-end Optimized Video Compression with MV-Residual Prediction”, IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2020.

[7].Tong Chen, Haojie Liu, Qiu Shen, Tao Yue, Xun Cao, Zhan Ma,” Deep Coder: A Deep Neural Network Based Video Compression”, IEEE Visual Communications and Image Processing (VCIP),2017.

[8].Zhibo Chen, Tianyu He, Xin Jin, Feng Wu, “Learning for Video Compression”, IEEE Transactions on Circuits and Systems for Video Technology , 566 – 576 Volume: 30, Issue: 2, Feb. 2020.

[9]. J. Balle, V. Laparra, and E. P. Simoncelli, “End-to-end optimized image compression”, arXivpreprintarXiv:1611.01704, 2016.

[10].J. Balle, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, “Variational image compression with a scale hyper prior” ,arXivpreprintarXiv:1802.01436, 2018.

[11]. G. Lu, W. Ouyang, D. Xu, X. Zhang, C. Cai, and Z. Gao, “DVC: An end-to-end deep video “compression framework,” in Proceedings of theIEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 11 006–11 015, 2019.

[12]. Ren Yang, Fabian Mentzer, Luc Van Gool“Learning for Video Compression with Hierarchical Quality and Recurrent Enhancement”, 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Year: 2020.

[13]. Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, “Learning image and video compression through spatial-temporal energy compaction,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 10 071–10 080.

[14]. A. Habibian, T. van Rozendaal, J. M. Tomczak, and T. S. Cohen, “Video compression with rate-distortion auto encoders,” in Proceedings of the IEEE International Conference of Computer Vision (ICCV), 2019

[15]. For Training (vimeo 90k dataset):http://data.csail.mit.edu/tofu/dataset.html.

[16]. For Testing (UVG data set): http://ultravideo.fi/#testsequences.

[17]. For Testing (JCT-VC data set): https://github.com/remega/video_database/tree/master/videos.

1 Dr.M.Sreelatha

Designation:

Professor & Head, Dept. of CSE.

Qualifications:

Ph.D(Computer Science & Systems Engineering) from Andhra University, Visakhapatnam.

M.Tech (Computer Science & Engineering) from NIT, Warangal.

B.Tech(Computer Science & Engineering) from ANU,AP.

2 Dr.R.LakshmiTulasi

Designation:

Professor

Qualifications:

Ph.D (CSE) in 2017 from JNTUH, Hyderabad.

M.Tech(CSE) in 2005 from JNTUCEA, Anantapur.

B.Tech(CSE) in 2000 from Bapatla Engineering College .

3 Sri.K.Siva Kumar

Designation:

Assistant Professor

Qualifications:

Ph.D in CSE [Pursuing] - Visvesvaraya Technological University, State Govt of Karnataka, India.

M.Tech(CSE) in 2012 from JNTUK, Andhra Pradesh.

B.Tech(CSE) in 2005 from JNTUH, Andhra Pradesh.