Signal Processing 191 (2022) 108346

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Deep Architectures for Image Compression: A Critical Review

Dipti Mishra [a], Satish Kumar Singh [b,*], Rajat Kumar Singh [a]

[a] Department of Electronics and Communication Engineering, Indian Institute of Information Technology Allahabad, Prayagraj, India
[b] Department of Information Technology, Indian Institute of Information Technology Allahabad, Prayagraj, India

ARTICLE INFO

Article history:
Received 15 March 2021
Revised 8 August 2021
Accepted 28 September 2021
Available online 30 September 2021

Keywords:
Image compression
Deep learning
DNN
Review
CNN
Survey

ABSTRACT

Deep learning architectures are now pervasive and filled almost all applications under image processing, computer vision, and biometrics. The attractive property of feature extraction of CNN has solved a lot of conventional image processing problems with much-improved performance & efficiency. The paper aimed to review over a hundred recent state-of-the-art techniques exploiting mostly lossy image compression using deep learning architectures. These deep learning algorithms consists of various architectures like CNN, RNN, GAN, autoencoders and variational autoencoders. We have classified all the algorithms under certain categories for the better and deep understanding. The review is written keeping in mind the contributions of researchers & the challenges faced by them. Various findings for the researchers along with some future directions for a new researcher have been significantly highlighted. Most of the papers reviewed in the compression domain are from the last four years using different methodologies. The review has been summarized by dropping a new outlook for researchers in the realm of image compression.

1. Introduction

In this modern era of big data, the data size issue is a big concern. Because of limited memory space & channel bandwidth, data compression is required to diminish the size of the files required to be stored or transmitted. Data can be in the form of text, image, or video. Various researchers came up with different approaches and built algorithms based on a certain hypothesis, which has already been affirmed to overcome state-of-the-art methods. Some of the conventional approaches for compression purposes use mathematical transforms for energy compaction and spatial frequency isolation by exploiting various types of redundancies present in the image [1]. There is an essential need to compress the size of image files used in areas as mentioned above by exploiting the redundant information present [2,3]. The survey intends to focus on images having applications in various areas, such as natural scenes, remote sensing, biomedical applications, and video processing techniques. In the medical field also, the use of radiography, histo-pathological or whole-slide images, ultrasound, & MRI images creates a problem of large storage and archiving. The fundamental principle is to minimize redundancy & irrelevant information, by exploiting inter-pixel redundancy (using statistical dependency), psycho-visual redundancy (using different sensitivities), & coding redundancy (using fixed or variable length code). Initially in 1991, image compression algorithms, like JPEG [4] & JPEG2000 [5], were developed with some sequential steps using fixed coefficient based transform matrices, i.e., discrete cosine transform (DCT) [6] & discrete wavelet transform (DWT), [7], followed by quantization (scalar & vector) & entropy coding such as Huffman coding, run-length encoding (RLE) schemes [8] or arithmetic coding. Elias, Rissanen & Pasco invented this coding in 1987 [9] to compress the image. Great efforts have been dedicated to the reconstruction of compressed images. Nevertheless, the approaches mentioned above cannot be standardized as optimal as well as flexible for all types of images & applications. Classical compression schemes have been developed to exploit the spatial redundancy present in the images. The discrete cosine transform (DCT) [6], differential pulse code modulation (DPCM) [10] & the entropy coding of images are examples of the statistical approaches. The color & texture of an object in the image is majorly affected by its orientation & illumination. Also, the edges are usually the most important information; therefore, a good compression algorithm should try to minimize edge distortion [11] also. Later on, many techniques like transform picture coding [12], quantization [13], adaptive vector quantization [14], transform coding [15], adaptive transform coding techniques [16] have been reported aiming to provide maximum compression ratio with least distortion. Jayant quantizer [17] & Lloyd max quantization algorithm [18] are the best examples under this category. Subsequently, in 1980, wavelet transforms & neu-

* Corresponding author.
E-mail addresses: dipti.mishra28@gmail.com, rse2017502@iiita.ac.in (D. Mishra), sk.singh@iiita.ac.in (S.K. Singh), rajatsingh@iiita.ac.in (R.K. Singh).
URL: https://www.iiita.ac.in (D. Mishra), https://www.iiita.ac.in (S.K. Singh), https://www.iiita.ac.in (R.K. Singh)

https://doi.org/10.1016/j.sigpro.2021.108346
0165-1684/

ral networks have received significant attention based on the principle of hierarchical multi-resolution decomposition. The idea behind the transform is to decompose the input signal into a low-resolution successively (LR) coarse signal with their detailed information signals [19]. Some of the approaches emphasize generating the perfect reconstruction of the image, i.e., context-based region-of-interest (ROI) based compression schemes. It focuses on image regions, which are content-specific & significant for interpretation by the human visual system (HVS). Such regions are encoded with almost negligible information loss than the other non-significant portions of the image for a better compression ratio. Another ROI-based scheme is JPEG2000, wherein the transformed coefficients for the highlighted regions are processed in a manner to provide higher priority to highlighted regions than the background data. It allows the significant areas (ROI) to be reconstructed back with higher quality compared to the non-significant areas. They include principal component analysis (PCA) [20], clustering algorithms [21], and dictionary learning approaches [22], & many more. Principal component analysis (PCA) has also performed quite well in dimensionality reduction. More recently, WebP algorithm[1] has been proposed to improve image compression rates, especially for the high-resolution (HR) images, that have become more common in recent years.

In the literature, to check the efficacy of compression algorithm & the quality of the image obtained after reconstruction, authors prefer to use mean square error (MSE) [23], peak signal to noise ratio (PSNR) [24], structural similarity (SSIM) [25] and multi-scale structural similarity (MS-SSIM) [26]. The compression techniques utilizing the concept of image segmentation & saliency detection also exploits mean opinion square (MOS) [27], mean intersection over union (mIoU) [28] & mean average precision (mAP) [29] etc.

With the introduction of CNN, a machine can learn the important features, which are optimally used for the efficient representation of the image. For the application of lossy image compression, a neural network is declared to be very substantial than conventional JPEG & JPEG2000 due to the following reasons. Firstly, flexible non-linear synthesis & analysis transformation can be employed with the addition of many convolutional layers. Then, DNN enables the optimization of the "end-to-end" trained encoder-decoder module. Furthermore, recent advancement also proves the efficiency of deep learning in image compression [30,31]. However, the transformation from a traditional handcrafted feature selection system was very slow due to inadequate resources [31]. Summarizing the benefits of DNN based approaches over the conventional algorithm in the current scenario, there is a huge availability of the data required with the fast computational resources. Moreover, researchers are coming out with the different types of CNN architectures working specifically for the task of image compression. The research work on CNNs has been done since the late 70's, & most of them have been applied to lossy image compression [32]. Recently, deep neural networks (DNN) have produced favourable outcomes in varied computer vision tasks [33–35] like image classification [36], object detection [37], semantic segmentation [37], saliency map calculation [38], optical flow fields [39], single-image super-resolution [40,41], & generative models learning [42] with the state-of-the-art performances. Particularly, the autoencoder has been widely used in dimensionality reduction to produce compact representations of images. Thus, it is observed that autoencoders can extract deep compressed representation, also called binary codes from the images, by minimizing loss function & expected to obtain improvement over classical techniques like JPEG & JPEG2000, & so forth. Moreover, JPEG & JPEG2000 did not perform up to the mark for image compression at lower bit rates. It is ineludible in producing artifacts, e.g., blurring, ringing, & blocking. These artifacts arise due to abrupt truncation of high frequencies (edges) & edge discontinuities between neighboring image pixel blocks of size 8×8. For the reduction of the visibility of these artifacts, various compression algorithms have been reported based on convolutional neural networks (CNN) [33], recurrent neural networks (RNN) [43], autoencoders [44], & generative adversarial networks (GAN) [45].

It is necessary to write a survey that compares and summarizes these methods with a simple and integrated view. Already some review papers for image compression are already reported in the literature. Some focused on traditional image compression algorithms [2,46], a few on the mixture of traditional and current neural network approaches [47], some were centralized on deep learning-based image compression schemes [3,48] whereas a few were focused on video coding techniques [49]. In this survey, we focus on learning-based techniques (post deep learning era) & have not included classical image compression algorithms. This review is directed towards the interest of image researchers, who have a keen interest in state-of-the-art deep learning for image compression. This review paper is different from Dhawan [2,46,47], as it focuses only on post-deep learning era-based compression algorithms. It is different from Ma et al. [3,48] as it describes the critical findings in each paper including the compression performance, coding efficiency, computational complexity, model parameters, and applicability. Secondly, it shows a clear-cut distinction for low, medium, and high bit-rate compression algorithms. Thirdly, a new reader can easily find the type, efficiency, and power of various GPU machines used. The paper ended with the idea of the less and highly explored type of coding techniques, which can help a new researcher to start with. Additionally, a direction for the best path is provided based upon certain analysis and observation. Accordingly, we have included the discussion until the latest work is reported. The number of papers reported for different types of architecture is also shown in Fig. 1a. The frequency of papers overgrew in 2018, 2019 & 2020, as displayed in Fig. 1b. The analysis shows that CNN architecture-based "end-to-end" algorithms are reported more as compared to RNN, GAN, auto-encoders & variational autoencoders. Here, we are summarizing the review which aims to:

- manifest the learning-based schemes which have covered almost the complete domain of image synthesis & analysis.
- highlights the critical remarks including comparative analysis for some significant techniques reported so far in the image compression.
- pinpoint the problems faced in the successful implementation of deep learning in image compression.

The review paper is framed, as mentioned here. In Section 2, we have discussed various deep learning architectures for the task of lossless and lossy image compression. Section 3 ends with critical observations and findings based on the discussed techniques. Section 4 throws light on various significant challenges and problems faced by researchers in the mentioned algorithms. Section 5 ends with a conclusion giving future direction & dropping a new outlook for researchers in image compression in the future.

2. Deep architectures for image compression

DNNs are utilized to learn important features from the images & avoid redundant features or information. The layer-wise structure for a simple neural network, CNN and autoencoders are shown in Fig. 2a, Fig. 2b and Fig. 2c, respectively. A DNN consists of an input layer, output layer, and some hidden layers used for feature extraction. Autoencoder is a special type of neural network

[1] https://developers.google.com/speed/webp/.

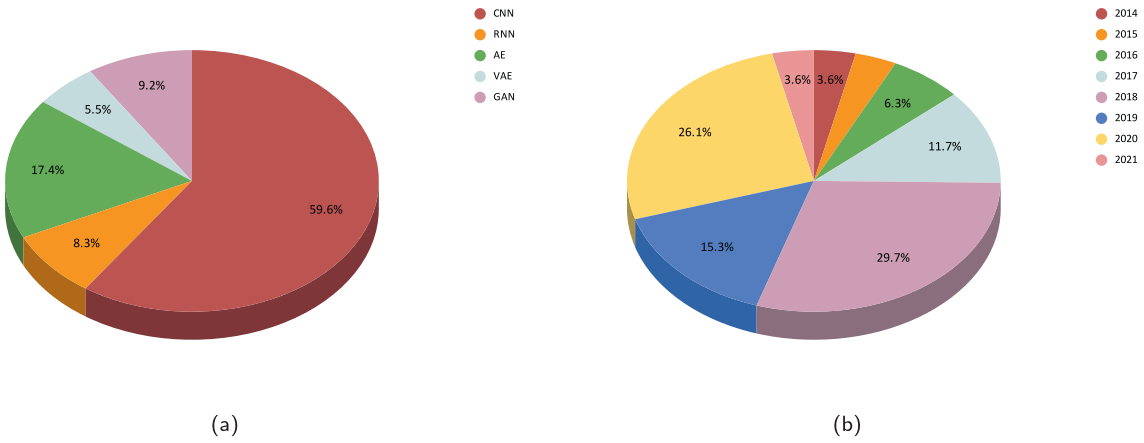


Fig. 1. Summarized report of various Deep Learning based Image Compression Algorithms (a) Deep architectures based image compression algorithms, (b) Year-on-Year reported compression algorithms.

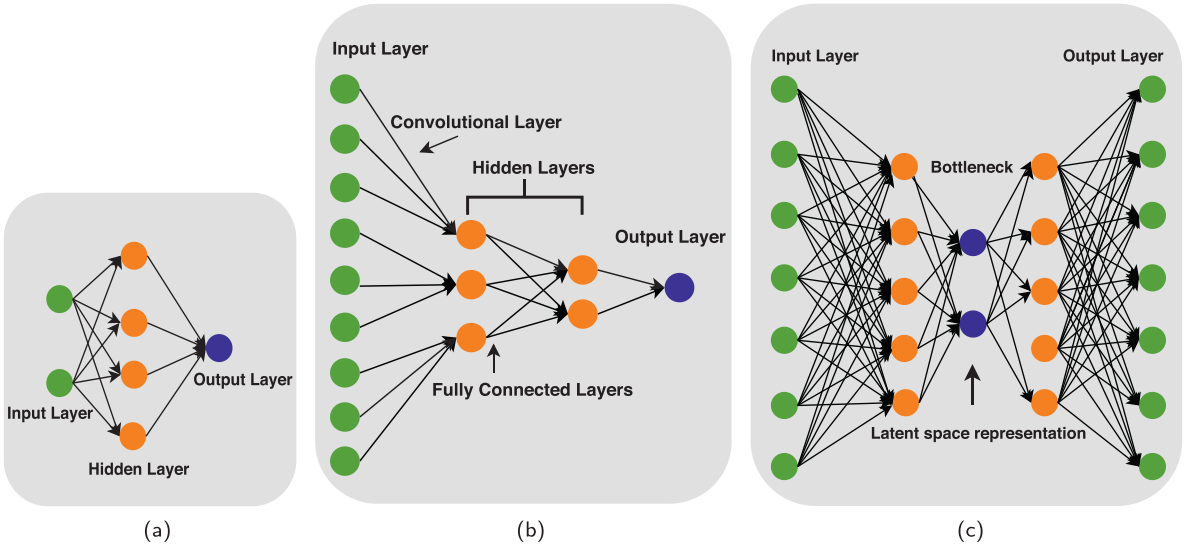


Fig. 2. Description of CNN structure with functionality of the layers (a) A simple neural network (NN) (b) A simple convolution neural Network (CNN) (c) A simple convolutional autoencoder (CAE).

used to learn the mapping between input and latent space representation (bottleneck). The feature extraction by a neural network leads to dimensionality reduction along with the removal of redundant information. After the removal of redundant information, reconstructed image quality is assessed with various performance measures like PSNR & subjective measures SSIM & MS-SSIM. Some of the techniques which we discuss are application-specific, while others depend upon some image processing or signal processing domain knowledge.

It is well known that compression schemes are broadly divided into two types; lossless & lossy compression. Lossy compression involves the loss of information while compressing the data, where the lost data cannot be recovered back. On the contrary, lossless schemes do not allow any loss of information during the compression process. Rate & distortion are the points of concern in lossless & lossy compression schemes, respectively.

2.1. Lossless image compression techniques

The research for providing lossless compression using CNN is still in its infancy, as the loss distorts at every step at the time of encoding information. But still, researchers are trying hard & report some of the medical image compression techniques producing images with minimum loss. During compression, the information loss across non-significant regions in medical images may be considered negligible, as it generally does not affect the final diagnosis of a disease by a health specialist. As a rule, some of the useful lossless techniques are being discussed here.

To access and analyze the medical images, a technique for whole slide imaging (WSI) is available for pathologists & researchers. WSI are very high-resolution images having much finer details whose dimension ranges from few to multiple gigabytes. Initially, Shen and Pan [50], has proposed architecture for lossless compression consisting of two stacked autoencoders (SAE) (based on RBM [51]) for learning distinct features of infected & non-infected malaria cells. The dimensionality reduction for medical images by autoencoder was then followed by Golomb rice encoding. The size of the cell was compressed first by using SAE, and a residual image calculated between the original & reconstructed image is encoded with the Rice coding algorithm. Generally, all the deep CNN approaches were based on either the L_1, L_2, or perceptual loss function. However, these functions failed to retain some important unique details which were important. Zhang and Wu [52] has proposed to exploit L_∞ loss function. To prevent large information loss, which could not be avoided using MSE, L_∞ loss helped in compression artifacts removal. Up to this point in time,

lossless image compression has not achieved much improvement with the aid of deep learning techniques. The L_∞ function posed an over-training constraint not to miss a single small important detail. On the other hand, Mentzer et al. [53] proposed the first lossy compression work for natural images, which was an "end-to-end" trained probabilistic model-based on adaptive entropy coding. The network was able to learn image information with the auxiliary representation. The architecture consisted of three fully parallel feature extractors and predictors for an "end-to-end" training task. The feature extractor helped to generate feature representation while the predictor helped to model those features in the form of an image. Arithmetic coding is then used as entropy coding criteria. In medical images, one cannot afford any significant loss, as it can affect the diagnosis process. So, the constraint, while executing neural image compression, is to maintain the quality of the infected region's semantic portion with the original quality. Later on, Tellez et al. [54] proposed an algorithm to compress the size of histopathological images. The idea was to convert HR images to LR images using unsupervised learning techniques. The semantic region is identified with the help of the Grad-CAM technique [55]. Zhang and Wu [56] proposed a simple predictive encoder and a much complex adaptive CNN soft decoder. Along with soft decoding, hard decoding is also exploited over conventional decoder to obtain high compression gain. The network is optimized using the loss function $L = L_2 + \lambda L_\infty$, where λ is a hyper-parameter. Ma et al. [57] has proposed a wavelet based "end-to-end" optimized lossless as well as lossy model. Later on, Cheng et al. [58] designed a method consisting of generalization of hyperprior from lossy to lossless compression model. The author has tried to model the context model using Gaussian, Laplacian, Cauchy and Logistic distribution, out of which Gaussian mixture model is selected. On the other hand, Schiopu and Munteanu [59] presented a study of intra-prediction methods for lossless compression. The author invented lossless approach to predict samples using deep learning based predictors. The prediction error is modeled by context tree modeling method. A critical analysis of the methods presented is given in Table 1. A detailed comparison of algorithms discussed above with traditional cost in terms of compression performance and computational cost has been shown in Table 2, Table 3 and Table 4, respectively.

2.2. Lossy image compression techniques

All the predictive coding-based approaches discussed till now include negligible or very little information loss during data flow or training in a DNN. Many of the recently reported lossy compression schemes which have shown significant results are discussed here.

2.2.1. ***End-to-end autoencoder like schemes***

It is not effortless to train any neural network with some subnetworks in an "end-to-end" manner. In this section, various "end-to-end" autoencoding schemes have been discussed, aiming to provide the readers with precise classification of the different methodologies used.

In 2006, Hinton and Salakhutdinov [68] stated that image compression could be achieved by converting data from high dimensionality to lower dimensionality. The pre-training procedure involves learning a serial chain of RBM where each RBM contains only one layer for detecting the features. The activation map or activations learned from one RBM acts as input to train the next RBM, and the process goes on. Then, these RBMs are unfolded to form a deep autoencoder type of structure. The network is fine-tuned with the help of back-propagation. With large initial weights, a local minimum is challenging to find, and with small initial weights, gradient dies, which makes the training of autoencoder difficult. With the experiment on 20,000 training images, the author claimed that without pre-training, the reconstructed images contained only average image quality, and with pre-training, the total training time reduces. Pre-training on document dataset also outperforms the state-of-the-art technology like latent semantic analysis (LSA) [69]. Then, Ollivier [70] introduced the concept of autoencoders, also called generative models, which are used to model the relationship between input data samples and latent space as $\mathbf{X} \xrightarrow{f} \mathbf{Y} \xrightarrow{g} \hat{\mathbf{X}}$. Here, the author exploited the autoencoder to compress the data through its generative property as well as to determine the differences between training the autoencoder to decrease the reconstruction error. The work showed an optimization method while generating the images based on feature selection and reducing the code-length. Hence, the objective was to generate the actual images with maximum probability closed to the original while keeping code-length as small as possible. Usually, the reconstruction error is an element-wise measurement such as MSE or binary cross-entropy between a reconstructed image and the input image. This element-wise reconstruction is not a convincing index of the virtue of a learned latent vector. So, in 2015, Larsen et al. [71] contributed by developing an unsupervised scheme to train the weights of encoder-decoder network & utilizing a learned distance measure (Kullback–Leibler (KL) loss & a negative log data-likelihood). On the other hand, Toderici et al. [72] focused on thumbnail images of 32×32 size, which is a left-out part by various researchers. RNN based autoencoder has been used to provide variable-rate encoding. The proposed network consist of encoder (E), Binarizer (B), & Decoder (D), such that the autoencoder for an image $\mathbf{x}$ can be defined as $\mathbf{x}' = D(B(E(\mathbf{x})))$. The feed-forward convolutional & the deconvolutional residual encoder used the L_2 measure distance. Later on, Gregor et al. [73], proposed a recurrent variational autoencoder to exploit unsupervised representation learning with RNN. The notion was to apply a better cost function to train high-level features than other fine details. With the aid of convolutional deep recurrent attentive writer (DRAW) [74], the network generated better samples and likelihoods. For low bit rates, the approach was conceptual and helped to achieve plausible images at reconstruction. Later on, Sento [75] combined the architecture for non-recurrent three-layer autoencoder with Kalman filter for compressing the images. The function of the Kalman filter was to update the weights of the DNN. The filter was optimized and improved the speed of the error convergence with low loss. Extending the work reported in [72], Toderici et al. [76] focused upon thumbnail images, the images on the web page for providing variable rate compression or encoding. Initially, the input images were encoded through the encoder, which was then binarized into a code map, which could be further stored or directly transmitted to a receiver for decoding. The binarizer was used to compensate for quantization noise and has produced a checkerboard type feature map structure. The progressive entropy encoder, which used pixel RNN and binary RNN, was found to be useful in variable-rate decoding of the images. L_1 loss was used to optimize the network for reconstructing the images. With the proposed architecture, $32 \times 32 \times 3$ sized thumbnail patch images are converted into $2 \times 2 \times 32$ binarized form in a single iteration. Each iteration resulted in $1/8$ bits per pixel (bpp), providing the compression ratio of $192 : 1$. The RNN units exploited are LSTM, associative LSTM & gated recurrent units (GRU). The architecture has been shown in Fig. 3.

Inspired by the concept of autoencoders, Theis et al. [77] had proposed an efficient approach to deal with the non-differentiability of the quantization step in a convolutional autoencoder (CAE). A typical CAE is defined by three components: encoder E, decoder D, & probabilistic model P.

Table 1
Overview of lossless image compression based on deep learning techniques.

Paper	DNN	Train/Test	Performance Measures	Parameters, Runtime	Critical Findings/Remarks
Shen et al. [50] 2016	AE	Red Blood Cell Images (PEIR-VM)[a]	bpp=5.1729, PSNR=27.50 dB, 4.9%,9.1%,17% bit saving over CALIC [60], JPEG-LS & JPEG-LM	• 18060	• Trained with cross-entropy loss
				• 30 ms	• For less dataset size, the method is not good due to more weight of intra-image similarity than inter-image similarity • Applicable for small size medical images
Zhang et al. [52] 2018	GAN	DIV2K	PSNR=39.49 dB, SSIM=0.9745	• 6868580	• The network is optimized using weighted sum of MSE, adversarial and L_∞ loss to reduce compression artifacts • Applicable for very less compression ratio & natural images • Surpassed CALIC, JPEG2000, WebP & ARCNN
Mentzer et al. [53] 2018	CNN	Open Images dataset [61]	bpp=2.646, 46%, 7.5% & 4.4% bit saving over PNG [62], JPEG2000 & WebP	• 10 M	• Probabilistic model (L3C) based adaptive entropy coding learns information with auxiliary representation
				• 0.816 s	• Applicable for natural images and code is available at https://github.com/fab-jul/L3C-PyTorch
Tellez et al. [54] 2019	GAN	Camelyon 16[b], Tupac 16[c], Rectum[d]	Predicted the tumor proliferation patch with 80.6% accuracy even after compression	• 0.558 s	• Compressed the size of histo-pathological images using BiGAN before classifying tumor using Grad-CAM [55] to highlight the region of interest • Applicable for medical WSI images and the code implementation is available at https://github.com/davidtellez/neural-image-compression
Zhang et al. [56] 2020	AE	DIV2K	bpp=1.5, PSNR=42.25 dB	• 6868580 (1.08 MB)	• l_∞ constrained soft decoding method to obtain lossless compression with a conventional predictive encoder and much complex adaptive CNN decoder
				• 0.08s on 4 NVIDIA Titan Xp GPUs	• Surpassed JPEG2000, CALIC, WebP & BPG[e] • Applicable for professional applications
Ma et al. [57] 2020	CNN	DIV2K, ImageNet/ Kodak	bpp=0.3, PSNR=31.5 dB, SSIM =8.5 dB	• 1.29 million	• Wavelet like transform, entropy coding and dequantization are used
				• 46.99s on NVIDIA GeForce RTX 2080 Ti GPU	• Both lossless and lossy compression are supported
Cheng et al. [58] 2020	CNN	CLIC/ Kodak	bpp=3.475		• Used L2-norm with loss function which contributes to stable and fast training • Gaussian mixture model is used, while L3C used Logistic mixture model
Schiopu et al. [59] 2020	CNN	4K UHD	bpp=1.952	• 12 minutes on NVIDIA Titan X GPU	• A deep learning-based approach for computing the residual-error for a dual prediction method • An entropy coder performing context-based bit-plane coding to encode the residuals

[a] https://groups.csail.mit.edu/vision/TinyImages [b] https://camelyon16.grand-challenge.org/ [c] https://tupac.tue-image.nl/node/3 [d] https://www.pathologyoutlines.com/topic/colonhistology.html [e] https://bellard.org/bpg/.

$E : \mathbf{R}^N \rightarrow \mathbf{R}^M, D : \mathbf{R}^M \rightarrow \mathbf{R}^N, Q : Z^N \rightarrow [0, 1]$, where M & N are the dimensionalities in original & latent space respectively, & Q is the rounding-based quantization function used. For an input image $\mathbf{x}$, the loss function is defined as $Loss = -log_2 Q([E(\mathbf{x})]) + \beta.d(\mathbf{x}, D([E(\mathbf{x})]))$. Here, β is the trade-off parameter & d is distortion parameter introduced during encoding-decoding mechanism. The first & second term in the above equation are concerned with rate or number of bits & distortion respectively [77]. Toderici et al. [76], instead, uses a random binarization [78], but in this work, a stochastic rounding operation is used for integers and given as: $z \approx \lfloor z \rfloor + \epsilon, \epsilon\epsilon \in [0, 1]$, $P(\epsilon = 1) = z - \lfloor z \rfloor$, where z is the intensity value to be quantized, ϵ is a parameter introduced for rounding operation, $\lfloor y \rfloor$ is the floor operator. For backward propagation, the derivative is given as $\frac{d}{dz}\{z\} : \frac{d}{dz}(\mathbf{E}[\{z\}] = \frac{d}{dz}z = 1$, where $\mathbf{E}$ is the expectation. With this method, the reconstructed images appeared visually more similar or better than JPEG2000 reconstructed images as artifacts produced by JPEG2000 are noisier than CAE. A critical difference of the technique with Balle's was that the latter was actually trained concerning a rate-distortion loss that was meant to optimize performance for adaptive coding. The approach discussed being "end-to-end" could be efficiently optimized for any loss function. But somehow, researchers find it difficult to invent perceptually new relevant metrics which could be logically optimized for image compression. Unfortunately, research to design a perceptually related metric that is suitable for optimization is still in its infancy. Covell et al. [79] has introduced a new approach for achieving high-quality reconstruction basically in high detailing area. The approach enabled to adaptively vary the number of symbols transmission based on content, the context of the image &

Table 2
Compression performance comparison of lossless approaches with traditional codecs (PSNR (dB)/$\|e_\infty\|$).

bpp	CALIC	JPEG2000	WebP	BPG	FLIF [63]	Minnen's [64]	Lee's [65]	DnCNN [66]	Zhang's [56]
					Kodak				
2.61	49.95/1.00	47.96/5.21	45.86/6.33	49.11/4.42	49.10/4.51	49.22/4.89	49.20/5.01	50.15/2.00	50.26/2.00
1.98	45.19/2.00	44.76/8.42	43.74/8.92	45.78/7.08	45.69/7.27	45.94/8.02	45.89/8.24	46.25/4.00	46.42/4.00
1.60	42.32/3.00	42.78/11.58	41.91/11.80	43.78/9.88	43.62/10.27	43.96/11.24	43.91/11.35	43.97/6.00	44.19/5.83
1.35	40.23/4.00	41.20/14.12	40.55/14.42	42.26/12.63	42.01/13.19	42.39/13.95	42.32/14.11	42.35/8.00	42.61/7.29
1.19	38.53/5.00	40.07/15.80	39.53/16.33	41.20/15.58	39.95/16.89	41.38/17.16	41.31/17.47	41.11/9.92	41.38/9.04
1.03	37.17/6.00	38.98/19.46	38.53/19.54	40.14/17.63	39.81/19.15	40.28/19.42	40.22/19.85	39.96/11.84	40.29/10.79
0.91	36.02/7.00	38.11/24.38	37.67/22.50	39.32/20.04	38.94/22.68	39.43/24.68	39.40/25.22	38.98/13.79	39.36/12.38
0.82	35.10/8.00	37.31/27.10	36.99/24.41	38.56/23.39	38.02/26.94	38.64/27.47	38.63/27.62	38.12/15.80	38.55/14.16
					LIVE1 [67]				
2.78	49.93/1.00	47.76/5.51	45.73/6.31	48.99/4.38	49.02/4.24	49.21/4.61	49.18/4.78	50.14/2.00	50.23/2.00
2.15	45.19/2.00	44.36/8.72	43.39/9.38	45.42/7.24	45.37/7.62	45.65/7.91	45.61/7.94	46.12/4.00	46.28/4.00
1.76	42.31/3.00	42.35/11.76	41.41/12.34	43.26/10.34	43.16/10.85	43.49/10.88	43.45/11.02	43.68/6.00	43.92/5.82
1.50	40.20/4.00	40.64/14.51	40.00/14.89	41.72/12.86	41.52/13.48	41.97/12.98	41.91/13.12	42.02/7.82	42.25/7.20
1.31	38.49/5.00	39.41/16.72	38.83/17.80	40.49/15.76	40.21/16.26	40.75/15.92	40.69/15.99	40.63/9.79	b40.91/9.03
1.15	37.13/6.00	38.32/20.66	38.11/19.10	39.47/19.24	39.12/19.81	39.71/19.64	39.67/19.81	39.49.11.88	39.79/10.86
1.03	35.98/7.00	37.39/24.93	36.92/24.41	38.65/20.90	38.20/22.59	38.81/21.48	38.78/21.72	38.45/13.82	38.81/12.20
0.94	35.05/8.00	36.62/26.34	36.55/24.28	37.91/23.93	37.11/25.65	38.12/24.58	38.06/25.23	37.66/15.76	38.05/14.01

Table 3
Compression performance on lossless images taken from CLIC and Kodak datasets (CLICP AND CLICM stands for professional and mobile version, respectively).

Algorithm / Dataset	CLICP	CLICM	KODAK
PNG	4.298	4.374	4.350
JPEG2000	3.403	3.266	3.191
WebP	3.254	3.212	3.206
FLIF [63]	3.141	3.083	2.903
L3C [53]	4.098	3.691	3.547
Cheng et al. [58]	3.647	3.486	3.475

Table 4
Run-time complexity comparison of CNN based lossless algorithms with traditional lossless approaches on 512 × 512 size image; bpsp denotes bits per sub-pixel.

Codec	Encoding Time (s)	Decoding Time (s)	bpsp	Platform
PNG	0.213	6.09×10^{-5}	3.850	CPU
JPEG2000	1.48×10^{-2}	2.26×10^{-4}	2.831	CPU
WebP	0.157	4.12×10^{-2}	2.728	CPU
FLIF [63]	1.72	0.133	2.544	CPU
L3C [53]	0.242	0.374	2.646	GPU

reduced blocking artifacts. Moreover, Agustsson et al. [80] started towards an annealing schedule to begin from a soft discretization to transform into a hard discretization gradually. On the other hand, Dumas et al. [81] proposed a new technique utilizing a stochastic winner-take-all autoencoder, a variant of the autoencoder. Winner-take-all means that all the image patches keep pace with each other to get sparse representation. The highest absolute values were kept as it is; all the other values were made to zero. Then, the stored absolute values were quantized & then compressed uniformly. The sparse feature maps obtained are then uniformly quantized and coded through Huffman coding. However, Liu et al. [82] proposed the approach which differentiates from others in a way that it removed the use of perceptual metric with a perceptual loss & adversarial loss. Again, the approach was a rate-distortion "end-to-end" optimization consisting of a forward encoder, quantizer, and a backward decoder. For fast convergence, transfer learning was used, i.e., a trained network at low compression means less quantization size was used to learn the network at a high compression ratio (more quantization). Since a single autoencoder is tuned for a single unique transform and single quantization step size concerning distortion, So, Dumas et al. [83] claimed that comparable results could be achieved with a single transform. So the author has demonstrated that by learning a single unique transform, rate-distortion performance could be made better than JPEG2000. On the other hand, Zhou et al. [84] inspired by the work of Balle et al. in [85,86] proposed a variational autoencoder consisting of non-linear encoder transform, uniform quantizer, non-linear decoder transform with a post-processing module. The latent space encoding for input

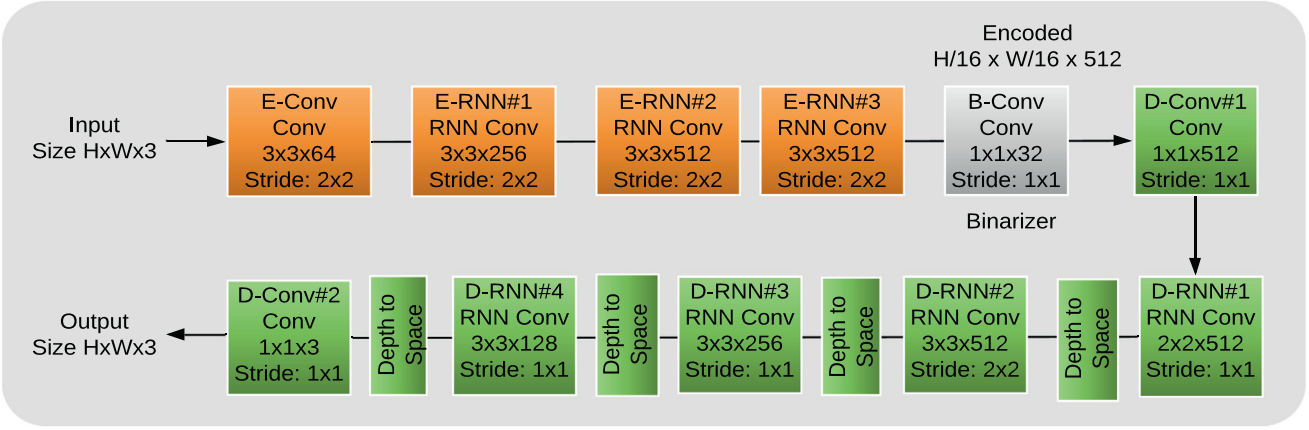


Fig. 3. Full resolution image compression framework using RNN [76].

image $\mathbf{x}$ is $\mathbf{y} = f_E(\mathbf{x})$ is then quantized to have $\hat{\mathbf{y}} = Q(\mathbf{y})$, where Q is the quantization operator. Finally, the transformed decoded image is $\hat{\mathbf{x}} = f_E(\hat{\mathbf{y}})$. Laplacian distribution was used in place of Gaussian distribution as in Balle et al. [86] whose parameters were learned by hyperprior autoencoder & gave an improvement of 0.1 dB at 0.15 bpp. The post-processing module was to reduce blocking artifacts. So the difference between Balle's [85] & Zhou's approach is that the autoencoder with pyramidal encoder & other convolution structures was used to improve the performance. Similarly, Torfason et al. [87] reported work for training a DNN for semantic understanding. The work was inclined towards the choice of a loss function, which is a control switch for minimizing the total bit rate along with reconstruction error. The author proposed a compression algorithm which could be further used for jointly train network for classification and segmentation tasks with image understanding. Moreover, Cheng et al. [88] has exploited convolutional autoencoder in place of transform and inverse transform. Some smooth approximations were incorporated to avoid non-differentiability for a requirement in gradient flow during the training of convolutional autoencoder (CAE). Rounding-based quantization was used as an entropy encoder similar to the convolutional rate-distortion cost function.

$$J(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}) = \left\|\mathbf{x} - \hat{\mathbf{x}}\right\|^2 + \lambda\|\mathbf{y}\|^2 = \left\|\mathbf{x} - g_{\boldsymbol{\phi}}(f_{\boldsymbol{\theta}}(\mathbf{x})) + \mu\right\|^2 + \lambda\|f_{\boldsymbol{\theta}}(\mathbf{x})\|^2 \quad (1)$$

The first term in the equation is the classical mean square error (MSE) distortion between the original images $\mathbf{x}$ and reconstructed images $\hat{\mathbf{x}}$, λ controls the trade-off between the rate and distortion, μ is the uniform noise, $f_{\boldsymbol{\theta}}(\mathbf{x})$ denotes the intensity of the compressed data $\mathbf{y}$, which denotes the number of bits used to encode the compressed data. Moreover, Alexandre et al. [89] had adopted an autoencoder with some skip connections as used in Mentzer et al. [90] by using the concept of importance map from Li et al. [91]. A specific sequence of convolution layers produced the importance map. To counter the non-differentiability of quantization, the technique proposed by Agustsson et al. [80] was used. On the other hand, Chen and He [92] have preferred semantic metric over PSNR & a weighted loss function, which is a combination of MAE, discriminator & semantic loss. The exploited loss function performed well to generate the face images accurately. Then, Ayzik and Avidan [93] proposed a compression algorithm exploiting the idea of correlation learning between encoder input & decoder input. The concept of additional side information is used to feed it along with latent space representation to the decoder to obtain the original image. This process is called distributed source coding (DSC).

The synthetic image is generated after applying DSC on a decoded image and a similar size patch in the original image, which is uploaded on the cloud. Then the final image is evaluated, summing this synthetic image with the originally decoded image. The process is quite complicated and increases the storage space required to store synthetic image & side information. Also, finding and matching the exact location of a particular patch is quite challenging. So the author has assumed that a particular patch would roughly lie at the same location as in the image plane to overcome this problem. Moreover, Hu et al. [96] at first did a wonderful job of summarizing and highlighting the novelties & improvements to be done for image compression methods. The author's significant investigations are that if two models are trained with PSNR, they may show a low level of MS-SSIM at a given bit-rate. Similarly, the models trained with MS-SSIM show the lower value of PSNR. But the author's proposition performed better for both when optimized for PSNR & MS-SSIM. For extremely low bit-rates (bpp<0.1 bpp), Raman et al. [98] proposed a compression framework with a stacked autoencoder and a Switch Prediction Network. In the encoder-decoder architecture, the layer-wise loss is calculated after every max-pooling & un-pooling layer. The network was optimized using the weighted loss function, which is a combination of GAN adversarial loss function, MSE loss calculated after the fourth convolution layer of a pre-trained Alexnet, loss after max-pooling & unpooling layer. The technique proposed by Cheng et al. [99] was based on the selection or development of a better entropy coding technique, which generally affects the optimization of other network parameters positively to obtain better rate-distortion performance. For this, discretized Gaussian mixture likelihoods along with attention models are used to reduce remaining spatial redundancy after quantization operation. The joint rate-distortion is obtained using the Lagrange multiplier. Later on, to avoid the training of one CNN network per bpp, Cai et al. [104] proposed a single CNN to perform compression at different bits per pixel using multiple latent space projections and variable quantization levels. Tucker decomposition is exploited to project the latent space to different rank matrices to get multiple dimensions.

Lee et al. [100] designed the first "end-to-end" optimized image compression approach that outperformed VTM 7.1 which is the latest software for Versatile Video Coding (VVC). The author proposed a JointIQ-Net compression scheme obtaining quality enhancement along with entropy model improvement. The technique incorporated GMM based prior probability modeling for getting entropy minimization. The approach is different from Minen's [64] approach, as it effectively reduces the inter-channel and inter-spatial correlations along with improvement in coding efficiency and enhancement in quality. The BD rate-based coding gain compared to VVC Inta, BPG and Lee's [65] approach have been shown in Fig. 4. Punnappurath and Brown [101] developed a compression scheme of raw image compression to train the network proposed by Toderici et al. [76] using a joint loss function. The raw images are actually unprocessed RGB images. The joint loss function considers image fidelity along with raw reconstruction of the image. Sun et al. [105] reported a technique for reducing the model size of compression model by quantizing the weights and proposing a non-linear memory codebook. Later on, Chen et al. [102] propose to exploit non-local operations to exploit local and global correlation using variational autoencoder (VAE). For the variable bit-rate allocation, an attention mechanism is utilized to generate multi-scale important masks. A critical analysis of the methods discussed above is given in Table 5. The detailed comparison of "end-to-end" coding schemes with conventional codecs in terms of comparison ratio and BD-rate has been shown in Table 6 and Table 7, respectively.

2.2.2. *Down-up sampling based coding*

Down-sampling is generally required when there is a scarcity of transmission bandwidth, which is generally the real scenario. Based upon this fact, Balle et al. [85] proposed an approach for rate-distortion optimization of deep encoders & decoders and chosen to use an adaptive entropy encoding scheme over simple entropy coding. The additive noise is used to combat the quantization loss as compared to the binarized scheme of Toderici et al. [72]. The rounding quantization method proposed by Theis et al. [77] has reduced the computational complexity to some extent. The objective was to minimize a weighted sum of the distortion and rate, $R + \lambda D$, over the parameters of the analysis & synthesis transforms, where R, λ, D are the rate, tuning parameter, and distortion respectively. The optimization was based on MSE with a more flexible transforms cascading of linear convolutions & non-linearities. Since joint non-linearity introduced by generalized divisive normalization (GDN/IGDN) has been exploited inspired by the biological neural system, these transforms were found to be efficient for compression producing better results. Firstly the con-

Table 5
Overview of end-to-end autoencoder like lossy compression schemes.

Paper	DNN	Train/Test	Performance Measures	Parameters, Runtime	Critical Findings/Remarks
Hinton et al. [68] 2006	AE	MNIST[a]	Better than PCA	• 7030	• Provided a better way of initializing the weights than principal component analysis [20] for reducing dimensions of data • Optimizing the weights in nonlinear autoencoders is found to be very difficult. • Pre-training reduce the training time • Deep autoencoders produce low error than the shallower ones.
Ollivier et al. [70] 2014	AE				• Compression of the data through the generative property of autoencoders by minimizing the code-length
Larsen et al. [71] 2016	VAE	Celeba dataset		• 142341692	• Variational autoencoder (VAE) to learn a latent vector of the input image trained with a Kullback - Leibler (KL)loss and a negative log data-likelihood
Toderici et al. [72] 2016	RNN	6 million images from web/ Kodak	bpp=0.5, SSIM=0.77	• 7457472	• Focused on thumbnail images of 32×32 size • Superior over JPEG, JPEG 200 & WebP • Applicable for variable bit-rates
Gregor et al. [73] 2016	VAE	ImageNet	bpp=4.10	• 2988864	• Recurrent variational autoencoder used to learn the latent space representation • DRAW algorithm [74] to model the latent space with a Gaussian input distribution to produce plausible images
Sento et al. [75] 2016	AE	MNIST		• 163618	• Combines the architecture for non- recurrent three-layer autoencoder with Kalman filter for compressing the images
Toderici et al. [76] 2017	RNN	6 million images from web/ kodak	bpp=0.5, PSNR=33.59 dB, SSIM=0.8933, MS-SSIM=0.9877	• 7457472 • 2.69s	• Progressive encoding and variable bit-rate decoding of thumbnail images(32×32) using LSTM • Good performance on low bit-rates compression • Superior over JPEG, JPEG 200 & WebP • Pixel RNN is used for entropy coding
Theis et al. [77] 2017	AE	Random images from flickr.com/ Kodak	bpp=0.4, PSNR=29 dB, SSIM=0.83, MS-SSIM=0.94, High MOS than JPEG & JPEG2000	• 2742112	• Effective approach of rounding based quantization to deal with the non-differentiability of the quantization noise • Residual network is used for training compressive autoencoders with MSE. • A Laplace-smoothed histogram is used as the entropy model • Applicable for variable bit-rates and better than JPEG2000
Covell et al. [79] 2017	RNN	ImageNet/ Kodak	bpp=0.25, PSNR=27 dB	• 4457472	• Use of Stop Code Tolerant (SCT) to train the recurrent CNN • Multi-class method for achieving high-quality reconstruction basically in high detailing area • Overcome JPEG & RNN based autoencoders • Entropy coding is missing
Agustsson et al. [80] 2017	AE	ImageNet/ Kodak	bpp=0.2, MS-SSIM=0.92	• 464154	• Learning of latent space representation jointly optimized for feature representation • Vector quantization over scalar quantization is preferred with variable quantization level • Visually appealing results over JPEG, BPG & JPEG2000
Dumas et al. [81] 2017	AE	ImageNet/ Set14	bpp=0.1, PSNR=31.5 dB		• A new technique utilizing stochastic winner-take-all autoencoder (SWTA-AE) where learning takes place with a global rate-distortion constraint • Better than JPEG
Liu et al. [82] 2018	CNN	CLIC 2018	bpp=0.2, MS-SSIM=0.93, 7.81% BD-Rate reduction over BPG & JPEG2000	• 6 million	• Training of CNN with perceptual loss and adversarial loss to generate sharp details • Outperformed BPG, WebP, JPEG2000 & JPEG • Applicable for low bit-rates
Dumas et al. [83] 2018	AE	ImageNet, BSDS300[94]	bpp=0.2, PSNR=31 dB	• 33152 on NVIDIA GTX 1080	• Recurrent autoencoder for encoding the residual for the recovery of the image based on the concept of spatial adaptive binary coding • Quantization size learning for each feature map • Used of GDN and IGDN to speed up the training process • Surpassed JPEG2000 and H.265 • Applicable for variable bit-rates

(*continued on next page*)

Table 5 (*continued*)

Zhou et al. [84] 2018	VAE	Random images from flickr.com/ CLIC 2018	bpp=0.15, PSNR=30.76 dB, MS-SSIM=0.955	• 46535029	• Variational autoencoder consisting of non-linear encoder transform, uniform quantizer, non-linear decoder transform with a post-processing module • Laplacian distribution used over Gaussian distribution to model the compressed representation • MS-SSIM was used as an effective loss function • Outperformed BPG & H.266
Torfason et al. [87] 2018	GAN	ImageNet	bpp=0.0983, PSNR=28.54 dB, SSIM=0.85, MS-SSIM=0.973	• 25.6 million	• Joint training for compression and classification for semantic understanding
				• GeForce Titan X GPU	• Being robust helped in saving decoding time, space with improvements in image quality • Outperform JPEG & JPEG2000 • High complexity and suitable for low bit-rates
Cheng et al. [88] 2018	AE	ImageNet/ Kodak	bpp=1.2, PSNR=33 dB	• 0.67s on GeForce GTX 1080 GPU	• Convolutional autoencoder in place of transform and inverse transform with some down-upsampling layers to create feature maps with low dimensionality • Principle component analysis (PCA) [20] is utilized to rotate feature maps to obtain more compact representation • Rounding based quantization was used before entropy encoder • Better than JPEG & JPEG2000 • Applicable for low bit-rates
Alexandre et al. [89] 2019	AE	ImageNet/ Kodak	bpp=0.126, PSNR=29.30 dB, MS-SSIM=0.9242	• 950000	• Autoencoder with some skip connections jointly trained for a weighted sum of MSE & MS-SSIM with a rate parameter • The bit allocation was controlled with the help of the importance map and quantization • Applicable for adaptive bit-rates
Ayzik et al. [93] 2020	CNN	KITTI 2012 [95]	bpp=0.025, MS-SSIM=0.925	• 75344	• The side information containing synthetic image is used using DSC to obtain good quality images • High complexity and much storage space is required to store side information and initial decoded image • Succeeded over JPEG2000 & BPG • Applicable for 0.02-2 bpp
Hu et al.[96] 2020	CNN	ImageNet, DIV2K	bpp=0.25, PSNR=30 dB	• 9561472	• Summarized the already reported end-to-end trained compression methods • Proposed a coarse to fine hyperprior based method • Outperformed JPEG, BPG and method in [65,76,86,90,97] • Applicable for variable bit-rates
Dash et al.[98] 2020	CNN	Cityscapes, CLIC 2019	bpp=0.0726, PSNR=23.93 dB, SSIM=0.8118	• 11 million	• Weighted loss function consisting of adversarial, MSE, and layer wise loss is used for optimizing network • Applicable for extremely low bpp (less than 0.1 bpp)
Cheng et al.[99] 2020	CNN	ImageNet/ Kodak	bpp=0.519, PSNR=33.62 dB, MS-SSIM=0.981	• 3689856	• Selection or development of a better entropy coding technique
				• 216s	• Discretized Gaussian mixture likelihoods along with attention models • The joint rate-distortion is obtained using Lagrange multiplier • High run-time complexity due to attention modules • Applicable for large bit-rates
Lee et al.[100] 2020	CNN	Kodak	bpp=0.2, PSNR=31 dB, MS-SSIM=0.7878	• 2 million	• Exploited GMM and introduced JointIQ-Net to outperform VVC Intra scheme, BPG, JPEG2000
Punna et al.[101] 2020	RNN	Images from DSLR Camera/ Google Pixel	PSNR=72.02 dB, MS-SSIM=1.7365	• 0.14 million	• Novel technique for raw image compression utilizing MAE loss for raw images
				• 0.9s on NVIDIA Titan X GPU	
Chen et al.[102] 2021	VAE	MS-COCO, CLIC/ Kodak	bpp=0.2, PSNR=30 dB, MS-SSIM=0.7768	• 262 MB	• Exploited non-local operation to exploit local and global correlation
				• NVIDIA P100 GPU	• Used for variable bit-rate coding
Cai et al.[103] 2020	CNN	AFLW[b]/LFW[c]	bpp=0.075, PSNR=29	• 1.75 million	• A hierarchical distortion loss function to protect both pixel-level fidelity for ROI and structural similarity for the entire image
				• 148.35s on GeForce GTX 980 Ti GPU	• Applicable for very low bit-rate

[a] https://www.yann.lecun.com/exdb/mnist/ [b] https://www.tugraz.at/institute/icg/research/team-bischof/lrs/downloads/aflw/ [c] https://www.cs.umass.edu/lfw/.

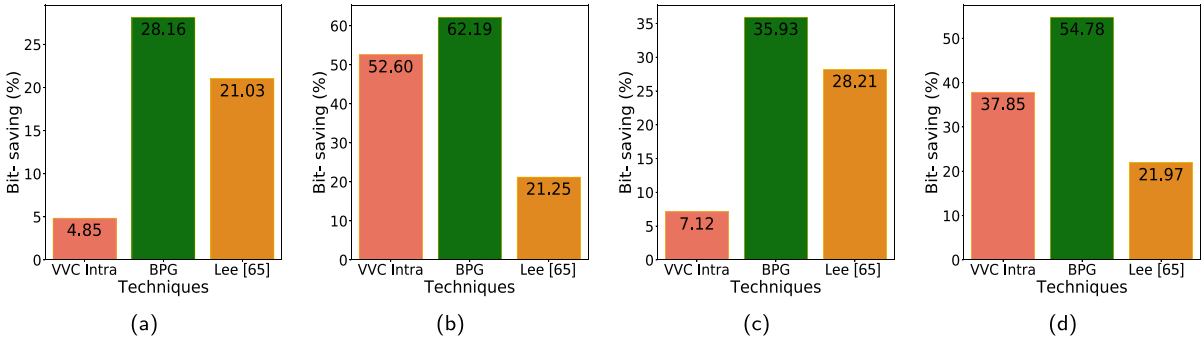


Fig. 4. BD-rate gain (%) obtained for JointIQ-Net approach [100] against the VVC Intra coding, BPG, Lee's [65] approach (a) On CLIC dataset when optimized with MSE (b) On CLIC dataset when optimized with MS-SSIM (c) On Tecnick dataset when optimized with MSE (d) On Tecnick dataset when optimized with MS-SSIM.

Table 6
Compression performance comparison of Dash's [98] "end-to-end" coding scheme with traditional codecs. $FSSIM_c$ refers to feature based similarity which is based on the fact that the HVS uses low-level features to interpret images, PLoss is the perceptual loss, Frechet Inception Distance (FID) is a perceptual quality metric proposed by Heusel et al. [109] which is a measure of how well the generated samples are approximating the real data distribution.

Algorithm/ Parameter	PSNR (dB)	SSIM	$FSSIM_c$	PLoss	FID
			bpp=0.0726		
JPEG2000	23.19	0.6796	0.8491	10.89	159.05
BPG	22.63	0.6411	0.8240	10.19	139.58
Dash et al.[98]	23.93	0.8118	0.9457	4.17	50.47
			bpp=0.0363		
JPEG2000	21.44	0.6002	0.7863	15.68	231.15
Dash et al.[98]	15.60	0.3446	0.6926	10.19	161.24

Table 7
BD-rate comparison with PSNR measuring the distortion level for "end-to-end" coding schemes setting BPG (4:4:4) as the reference.

Algorithm	Kodak	Tecnick	CLIC
JPEG	115.05%	217.84%	120.47%
Toderici's [76]	212.81%	244.26%	NA
Agustsson's [80] (factorized)	32.45%	32.17%	52.11%
Agustsson's [80] (hyperprior)	3.43%	-5.44%	10.15%
Minnen's [64]	-4.80%	-16.95%	-1.06%
Hu's [96]	-9.38%	-16.50%	-13.15%
Lee's [65]	-4.94%	26.84%	18.48%

volution operation on an input image is applied as

$$\mathbf{v}_i^{(k)}(m,n) = \sum_j (\mathbf{h}_{k,ij} * \mathbf{f}_j^{(k)})(m,n) + \mathbf{c}_{k,i} \tag{2}$$

Here, $\mathbf{f}_j^{(k)}$ is the input at ith input channel, kth stage at m,n spatial location, $\mathbf{h}_{k,ij}$ is the kernel, $\mathbf{c}_{k,i}$ is the corresponding bias parameter, $*$ is the $2D$ convolution operator which is then followed by down-sampling operation.

$$\mathbf{w}_i^{(k)}(m,n) = \mathbf{v}_i^{(k)}(s_k m, s_k n), \tag{3}$$

where $\mathbf{w}_i^{(k)}(m,n)$ is the downsampled output, s_k is the down-sampling factor at k^{th} stage. Then GDN operation used to speed up the training process is given as

$$\mathbf{f}_i^{(k+1)}(m,n) = \frac{\mathbf{w}_i^{(k)}(m,n)}{(\beta_{k,i} + \sum_j \gamma_{k,ij}(\mathbf{w}_j^k(m,n))^2)^{1/2}} \tag{4}$$

β and γ are the training parameters used for normalization. At the time of decoding, reverse operations need to be done where IGDN is applied as

$$\hat{\mathbf{w}}_i^{(k)}(m,n) = \hat{\mathbf{f}}_i^{(k)}(m,n).(\hat{\beta}_{k,i} + \sum_j \hat{\gamma}_{k,ij}(\hat{\mathbf{f}}_j^k(m,n))^2)^{1/2} \tag{5}$$

which is followed by upsampling as the reverse of downsampling operation

$$\mathbf{v}_i^{(k+1)}(m,n) = \begin{cases} \hat{\mathbf{w}}_i^{(k+1)}(m/\hat{s}_k, n/\hat{s}_k) & \text{if } m/\hat{s}_k \text{ \& } n/\hat{s}_k \text{ are integers} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

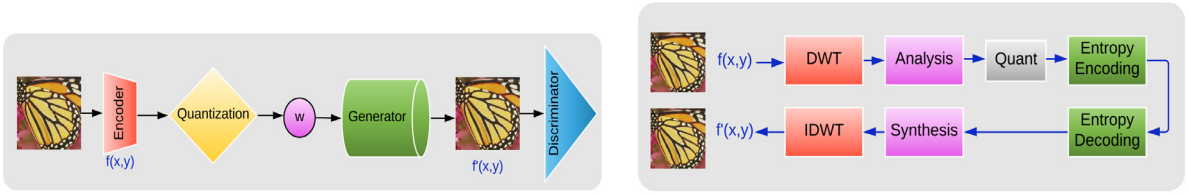
and then final output is obtained using convolution operation which is given as

$$\hat{\mathbf{f}}_i^{(k+1)}(m,n) = \sum_j (\hat{\mathbf{h}}_{k,ij} * \hat{\mathbf{v}}_j^k(m,n) + \hat{\mathbf{c}}_{k,i}) \tag{7}$$

This work extended an approach to rate-distortion optimization to deep encoders and decoders, and from a simple entropy encoding scheme to adaptive entropy coding. This work demonstrates an end-to-end trained image compression and decompression system, which achieves better bit-rate vs. quality trade-offs than established image compression algorithms (like JPEG2000). In addition to showing the efficacy of deep learning for a new application, a vital contribution of the paper is the introduction of a differentiable version of the "rate" function, which the authors show can be used for effective training with different rate-distortion trade-offs. The algorithm is trained with ImageNet [106] dataset giving PSNR=29.31 dB, MS-SSIM=0.9695 at bpp=0.189. It is a boon for the deployment of the proposed method on embedded systems, however, the cost of joint rate-distortion minimization for training is quite high along with 16.01 million training parameters with 0.58 s execution time.

2.2.3. *Saliency based coding*

It is well known that HVS is oriented towards the salient information present in the image. Some algorithms are exploiting this requirement of HVS by focusing on salient regions of an image during compression. Saliency information or region of interest (ROI) allows users to allocate variable bits and thereby helps in reducing the bits required to store the complete image. Based on this attractive idea, various researchers have proposed different algorithms based on CNN training. Like, Prakash et al. [107] provided a saliency-based compression model. The significant regions of the image were coded with a higher bit rate & vice versa while maintaining the optimal compression ratio and quality of the images. The salient regions were identified with the help of CNN producing a multi-structure region of interest (MS-ROI) generating the values ranging from [0,1] (since the sigmoid layer is the last layer in architecture). Here, 1 meant the highest salient region & vice versa. Moreover, the model was able to detect multiple salient regions in a single image, which was a drawback in earlier segmentation and class activation map (CAM) techniques [108]. The detailed comparison of Prakash's algorithm with conventional has been shown in Table 8.



(a) (b)

Fig. 5. (a) Semantic compression network with generative modelling [110], (b) Analysis & synthesis transformation architecture using wavelet transform [111].

Table 8
Compression performance comparison of saliency based approach [107] with traditional codec JPEG. PSNR-S referes to PSNR comparison for saliency region only and VIFP refers to pixel based visual information fidelity.

Codec	PSNR-S	PSNR	PSNR-HVS	PSNR-HVSM	SSIM	MS-SSIM	VIFP
				K odak			
JPEG	33.91	34.70	34.92	42.19	0.969	0.991	0.626
Prakash's [107]	39.16	34.82	35.05	42.33	0.969	0.991	0.629
				M IT Saliency			
JPEG	36.9	31.84	35.91	45.37	0.893	0.982	0.521
Prakash's [107]	40.8	32.16	36.32	45.62	0.917	0.990	0.529

Table 9
Architecture details of the algorithm proposed by Li et al.[91].

Encoder		Decoder	
Layer	Activation Size	Layer	Activation Size
Input	$3 \times 128 \times 128$	Input	$64(128) \times 16 \times 16$
$8 \times 8 \times 128$ conv, pad 2, stride 4	$128 \times 32 \times 32$	$1 \times 1 \times 512$ conv, pad 0, stride 1	$512 \times 16 \times 16$
Residual Block, 128 filters	$128 \times 32 \times 32$	Residual Block, 512 filters	$512 \times 16 \times 16$
$4 \times 4 \times 256$ conv, pad 1, stride 2	$256 \times 16 \times 16$	Residual Block, 512 filters	$512 \times 16 \times 16$
Residual Block, 256 filters	$256 \times 16 \times 16$	Depth to Space, Stride 2	$128 \times 32 \times 32$
Residual Block, 256 filters	$256 \times 16 \times 16$	$3 \times 3 \times 256$ conv, pad 1, stride 1	$256 \times 32 \times 32$
$1 \times 1 \times 64(128)$ conv, pad 0, stride 1	$64(128) \times 16 \times 16$	Residual Block, 256 filters	$256 \times 32 \times 32$
		Depth to Space, Stride 4	$128 \times 32 \times 32$
		$3 \times 3 \times 32$ conv, pad 1, stride 1	$32 \times 128 \times 128$
		$3 \times 3 \times 3$ conv, pad 1, stride 1	$3 \times 128 \times 128$

Moving on, Agustsson et al. [110], proposed an algorithm based on GAN with the generator & multi-scale discriminator (also called decoder) by learning optimal features of the image. The method only needed the storage of the semantic map and preserved region by generating real-looking images. This work proposed to combine a GAN loss with MSE, together with an entropy loss. Although the approach was not very novel, the adoption and combination of existing methods for the proposed solution are impressive. The complete methodology has been shown in Fig. 5. Later on, Li et al. [91] used the concept of spatial information as it is differently localized in different parts of the image. The proposed architecture consists of an encoder, binarizer, and a decoder, which finally calculates a content-specific importance map. The network details of the encoder & decoder network used are given in Table 9. Binarizer helped in quantizing the output of the encoder and was also constrained by the importance map. Since the quantization step is non-differentiable, making it available during back-propagation, a proxy function was used for joint rate-distortion optimization. The mask obtained after applying quantization was based on the allocation of bits to a specific portion of the image.

The compressed representation was further coded with convolutional entropy coder over context adaptive binary arithmetic coding (CABAC) framework [123] and obtained better results. The model, when used with CABAC or CAE, showed better SSIM than JPEG2000. On the other hand, Luo et al. [113] proposed a deep semantic compression model that helped in encoding semantic information with the reduction of computational resources while compressing the image. The encoder, which is a feature detector, tried to reduce the inter-pixel redundancies. Images were first encoded through a feature extraction module, then quantized and encoded into binary codes. Two types of models were proposed, i.e., pre-semantic DeepSic & post-semantic DeepSic. Pre-semantic deepSic was used to keep the salient information in the encoding process. After decoding, the post-semantic deepSic process used the feature stored to predict the class label and reconstruct the original image. Furthermore, Minnen et al. [64] extended own work in [124] and utilized "a single Pixel-CNN layer for modeling autoregressive priors and combine it with hyperprior for boosting rate-distortion performance". The author described a novel model for image compression that uses neural networks. This work for image compression extends the existing Balle's (2018) model by adding two elements: (1) Generalizing the hierarchical GSM model to a Gaussian mixture model (2) adding an autoregressive component. Balle's autoencoder uses Gaussian scale mixtures (GSMs) (which further use neighboring coefficients as context) for entropy encoding of coefficients and encode its latent variables as side information in the bit-stream. The model seems to be effective on the Kodak dataset, by outperforming Ballés method as shown in Fig. 6. Moreover in [115], Akbari et al. stated that the deep semantic segmentation model was used to extract the semantic map using CNN. The image & segmentation map was then used by the next network to get compact information. Then this compact rep-

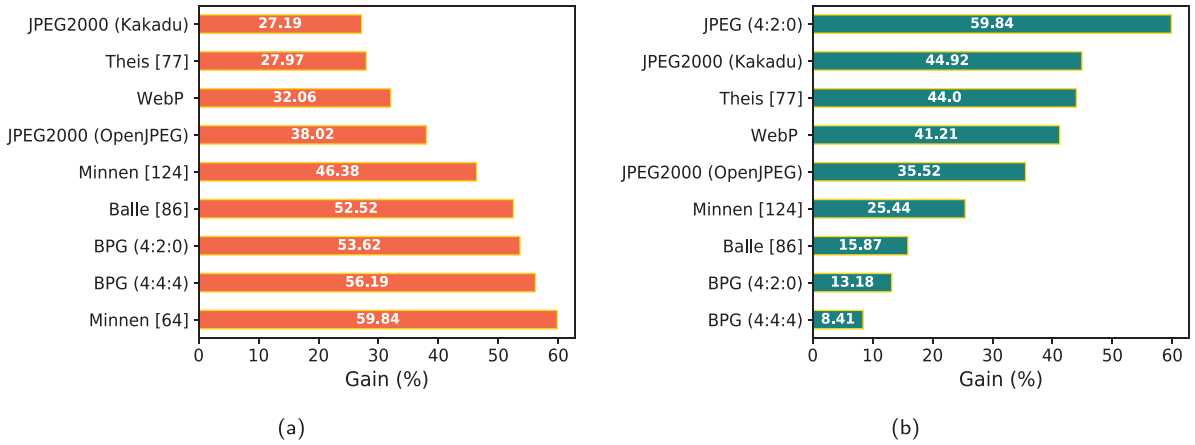


Fig. 6. (a) Average bit-rate savings of each method compared to JPEG (4:2:0) using PSNR on Kodak dataset, (b) Average bit-rate savings of Minnen's approach [64] compared to other codecs on Kodak dataset.

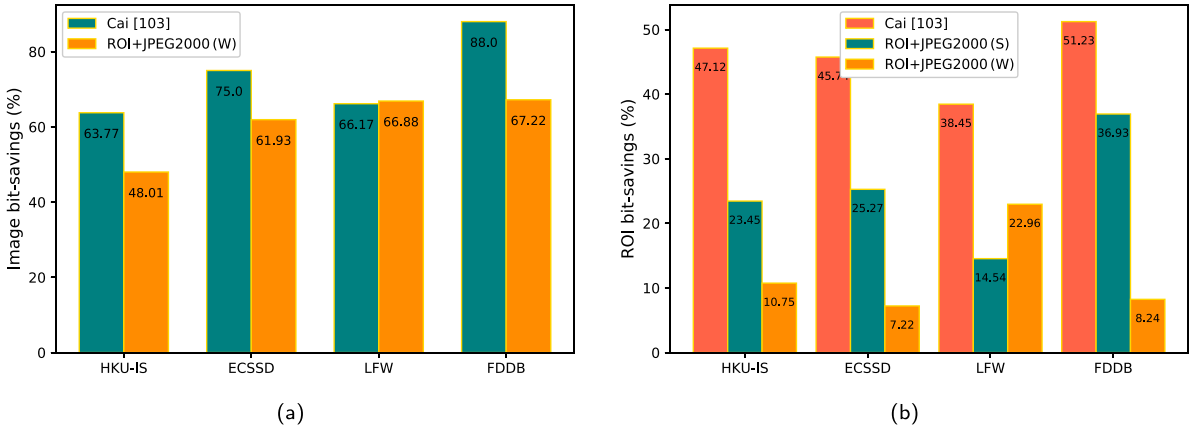


Fig. 7. (a) Bit-saving of Cai's [103] algorithm (relative to ROI+JPEG2000 (W)) for entire image in terms of MS-SSIM, (b) Bit-saving of Cai's [103] algorithm (relative to non-ROI JPG2000) for ROI region in terms of MS-SSIM.

resentation, along with a segmentation map, was used for reconstruction. For further improvement of quality, the data obtained from the segmentation map acted as residual itself, which tried to compensate for the image's loss. The technique proposed by Wang et al. [118] was designed to fulfill the task of image compression as well as classification. Initially, semantic regions were detected and recognized to obtain an importance map based on those semantic regions. An RNN based network was exploited to compress the image based on the importance map. The classification cum compression network-based algorithm has produced good visual quality images. Xue and Su [125] has proposed a post-processing based neural network containing spatial and channel attention modules connected in parallel. Zhou et al. [126] proposed another attention mechanism based end-to-end optimized image compression. Djelouah and Schroers [127] reported an iterative procedure which adapts the latent representation to the specific content we wish to compress while keeping the other parameters fixed. Then, Cheng et al. [128] performed a perceptual quality assessment study for various high resolution images. On the other hand, Cai et al. [103] introduced an ROI encoder/decoder-based image compression scheme to achieve direct rate-distortion optimization. Along with ROI mask prediction, a hierarchical distortion loss function is proposed to achieve pixel-level fidelity and to preserve structural information. The coding gain efficiency or bit saving comparison analysis separately for the entire image and ROI region on HKU-IS [129], ECSSD [130], LFW [131] and FDDB dataset [132] has been shown in Fig. 7, where ROI+JPEG2000 (S) and ROI+JPEG2000 (W) are region-of-interest JPEG2000 saliency-based and non-saliency based coding schemes respectively. The content weighted image compression approach proposed by Li et al. [119] is an extension of own work [91], for allocating the variable bits using importance maps. Channel-wise loss function is also introduced to reduce quantization error. Akutsu and Naruko [120] proposed a region of interest (ROI) prioritized CAE-based compression scheme, in which important maps help to control the allocation of bits according to significant regions. Moreover, Li et al. [121] introduced an efficient and effective entropy modeling technique based on context for the task of image synthesis and analysis. To take care of context, a 3D zigzag scanning with 3D code dividing technique. The technique enabled parallel entropy encoding and decoding. The latent space representation obtained after compression is statistically independent or depends upon some local context.

Due to this, any entropy coding technique fails to consider context while generating codes. For this, Li et al. [122] proposed an improved entropy modeling technique using some non-local operations to incorporate global similarity in the context. A proxy similarity metric and the spatial mask have been proposed for incorporating global similarity. Chen et al. [133] proposed to use a proxy network to mimic the perceptual model by serving as aloss layer of the network. A critical analysis of the methods discussed above is given in Table 10.

Table 10
Overview of Saliency based lossy coding techniques (QF: Quality factor).

Paper	DNN	Train/Test	Performance Measures	Parameters, Runtime	Critical Findings/Remarks
Prakash et al. [107] 2017	CNN	Caltech-256 [112]/ Kodak	QF=50, PSNR=34.82 dB, SSIM=0.969, MS-SSIM=0.991	• 138357544 (93 MB) • 1.2s on Titan X Maxwell GPU	• Deploy the significant regions of the image to provide variable bit-rates compression when trained with MSE • Applicable for adaptive bit-rate applications
Agustsson et al. [110] 2018	GAN	Open Images dataset	bpp=0.035, PSNR=21.80 dB	• 142341692	• GAN based framework was trained with MSE, adversarial and entropy loss • Performance for large images were impressive in terms of mIoU and human opinion • Applicable for very low bit-rate (<0.1 bpp) and better than BPG
Li et al. [91] 2018	CNN	ImageNet/ Kodak	bpp=0.11, PSNR=28 dB, SSIM=0.76	• 257557187	• Importance map was used for saliency guided compression and the spatial allocation of bits • Weighted sum of MSE and SSIM was used as loss function • Significantly outperfomed JPEG & JPEG2000 • Applicable for adaptive bit-rates
Luo et al. [113] 2018	CNN	ImageNet/ Kodak	bpp=0.2, PSNR=25 dB, MS-SSIM=0.87	• 5937024 on NVIDIA Quadro M6000 GPU	• Pre-semantic and post-semantic modules are proposed to preserve salient features and prediction of class label respectively • Superior over JPEG, JPEG2000, Toderici et al. [76] & Rippel et al. [114] when trained with MSE • Variable compression of surveillance streams for future smart cities and IoT
Minnen et al. [64] 2018	CNN	ImageNet/ Tecnick, Kodak	bpp=0.2, PSNR=28 dB		• Autoregressive and Hierarchical Priors are used to model latent space representation
Akbari et al. [115] 2019	CNN	Cityscapes [116], ADE20K [117]/ Kodak	bpp=0.088, PSNR=20.97 dB, SSIM=0.858	• 25912116 • 0.026s	• Trained with weighted sum of L_1 loss, SSIM loss [25], VGG loss, adversarial loss • Applicable for extremely low bit-rates (<0.1 bpp)
Wang et al. [118] 2019	CNN	Caltech-256/ Kodak	bpp=0.25, SSIM=0.89, MS-SSIM=0.93	• 138357544	• Joint network for image compression and classification based-on importance maps • Outperformed JPEG and Toderici et al. [76] for the same compression ratio • Superior over JPEG, BPG and Toderici et al. [76] • Applicable for variable bit-rates
Cai et al. [103] 2020	CNN	AFLW[a]/LFW[b]	bpp=0.075, PSNR=29	• 1.75 million • 148.35s on GeForce GTX 980 Ti GPU	• A hierarchical distortion loss function to protect both pixel-level fidelity for ROI and structural similarity for the entire image • Applicable for very low bit-rate
Li et al. [119] 2020	CNN	Random images from flickr.com/ Kodak	bpp=0.1, PNSR=27.8 MS-SSIM=0.91	• 0.166s on Intel (R) Xeon (R) 64 GB RAM, NVIDIA Titan Xp GPU • More parameters	• Jointly optimized for rate and distortion • Separately trained for MSE and MS-SSIM • Outperformed JPEG, JPEG2000 and other CNN methods
Akutsu et al. [120] 2020	CNN	Road Damage dataset[c]	bpp=0.263, PSNR=27.04 dB, SSIM=0.9700		• A method that adapts image quality for prioritized parts and non-prioritized parts for CAE-based compression • The proposed method uses annotation information for the distortion weights of the MS-SSIM based loss function
Li et al. [121] 2020	CNN	Images from Flickr.com/ Kodak	bpp=0.2, PSNR=30 dB, MS-SSIM=0.96	• 3890000 • 36.603s on NVIDIA Titan Xp	• Joint optimization of entropy coding with image analysis and synthesis
Li et al. [122] 2020	CNN	Images from Flickr.com/ Kodak	bpp=0.2, PSNR=30.5 dB, MS-SSIM=0.965	• 48 ms on NVIDIA Titan Xp GPU	• Context based entropy modelling exploiting non-local operation

[a] https://www.tugraz.at/institute/icg/research/team-bischof/lrs/downloads/aflw/ [b] https://vis-www.cs.umass.edu/lfw/ [c] https://rdd2020.sekilab.global/.

2.2.4. *Scalable image compression*

"Scalable image compression allows recovery of full image with multiple quality level simultaneously by decoding appropriate small subsets of the complete bit-stream, which is called the 'bit-stream scalability'. It is different from other techniques in the sense that the coded bit-stream after scalable coding is partially decodable. The coding is used when the channel capacity is limited or unknown or the decoding speed is not so fast. Various scalable image compression schemes have been discussed here.

The state-of-the-art techniques developed until 2014 reconstruct images based on the quantized or latent space representation with a uniform bit allocation scheme across the entire image. Therefore, Minnen et al. [134] reported an adaptive bit allocation based on visual saliency and context-based prediction. Since the

neural network helps model a non-linear class of functions, the approach helped in compression by mapping the image pixel values to quantized codes and then decode them again back to pixels. The challenge was to design a multi-scale model where lower resolution encoding could help to predict at a higher resolution. So, Ballé et al. [86] introduced "a scale hyperprior as side information for capturing spatial coorelation and then modeling the distribution of the codes as Gaussian scale mixtures (GSM) constrained by the learned scale hyperprior". The hyperprior is formulated for the entropy coder to model the spatial relation between the transformed coefficients. After encoding pixels with a CNN with GDN nonlinearities, the quantized coefficients are entropy encoded, where before the coefficients were independently encoded, the coefficients are jointly modeled using a latent variable model. In particular, the model exploits dependencies in the scale of neighboring coefficients. The additional latent variables are used to represent these scales efficiently. Both the coefficients and the representation of the scales are quantized and encoded in the binary image representation. The work is a step forward for deep image compression, at least when departing from the (Balle et al., 2016, 2017) scheme. Later on, Cai et al. [135] also proposed a scheme based on CNN consisting of multi-scale decomposition transform followed by an adaptive spatial bit rate allocation module. Then the adaptive bit allocation was implemented based on local spatial complexity in different regions of the image. The complete network needs to be trained only once, and it could provide variable rate compression due to multi-scale decomposition. After multi-scale decomposition of images, training of network is accomplished by adding uniform additive noise to avoid quantization's non-differentiability problem. In the same direction, Karkada Ashok and Palani [136] proposed a method that incorporated progressive image decoding as the decoder started decoding, and as soon as more bits were received, the image quality improved. Later on, Li et al. [137] exploited and modified the work of Li et al. [91] to compress the image dataset provided in CLIC 2018. The author has modified the autoencoder network (stacked CNN layers) to generate a 4-bit importance map, which was then finally fed to run-length entropy coding scheme after reordering. Li et al. [91] architecture differed from Ming Li et al. [137] in the fact that between two residual blocks, a $3 \times 3 \times 256$ layer was inserted. Later on, Zhou et al. [138] has proposed a low bit-rate end-to-end trainable image compression framework with a multi-scale and context adaptive entropy model. Recently, in mid 2020's, Wu et al. [139] have proposed an importance map based on variable bit-rate image compression with a new thought of providing at least sufficient bits to non-significant regions also which generally creates significant distortion at low bit-rates. To avoid retraining the compression-decompression network again and again for different bit rates, Chen et al. [140] proposed the notion of introducing quality scaling factors. This scaling is achieved by training a variational autoencoder for high bit-rate and retraining the same for low bit-rates. The author proposed to exploit non-local operation to exploit local and global correlation using VAE. For the variable bit-rate allocation, an attention mechanism is utilized to generate multi-scale importance masks. Yan et al. [141] proposed a semantically scalable coding scheme along with the provision of compression of feature maps. The coding helps in progressive decoding of bit-stream providing coarse-to-fine grained semantic granularities. A critical analysis of the methods discussed above is given in Table 11.

2.2.5. *Variable bit-rate coding*

Most of the deep neural network approaches are optimized for a single compression bit-rate with a dedicated network. However, for real time applications, it is the need to design variable-rate based image compression for high coding performance. However, in practice, it is essential to support the variable-rate compression or meet a target rate with a high-coding performance. Various variable bit-rate coding schemes have been discussed here. Like, Johnston et al. [146] proposed a recurrent autoencoder-based scheme for encoding the residual to reconstruct the image. The author improved the technique proposed by Toderici et al. [76] with the concept of hidden-state priming in RNN while training the network with a weighted SSIM loss function. The network-enabled progressive encoding for improving the image quality by generating more and more binary codes sequentially. Secondly, the network was capable of varying the bit rate based on image contents with the concept of spatial adaptive binary coding. Similarly, Kar et al. [147] proposed a compressor-decompressor model, which is a trained CAE, in which the compressor consisted of down convolution to extract essential features to reduce the bit allocation. Adaptive arithmetic coding, which is not a part of the training, was then applied to the compressed representation. On the other hand, Lee et al. [65] presented their take on a variational image compression model based on Ballé et al. [86] approach. It is a modification by combining the auto-regressive and a hierarchical approach to define the prior. This approach of defining a simplified hyperprior replaced the flow-based density model with a more straightforward Gaussian function. Later on, Choi et al. [148] proposed variable rate image compression in the hierarchy of having one compression module for providing multiple bit-rates. Lagrange multiplier and quantization bin size were the controlling parameters for adjusting dimensions of latent space representation. To obtain variable rate image compression, multiple times training a CNN-based model is required. But scaling or tuning the bottleneck representation causes degradation in compression performance at low bit rates. To combat this, Yang et al. [149] proposed a variable rate RD optimization by adding modulating and demodulating networks while optimizing auto-encoders in early 2020's. The modulating & demodulating networks are inserted before the encoder & decoder, respectively, which improves the RD behavior. Particularly by doing this, the author was successful in achieving variable bit-rate image compression by training the network only once. Sun et al. [150] proposed a semantic compression approach for variable rate compression, where each part of bit-stream denotes a specific object. A critical analysis of the methods discussed above is given in Table 12.

2.2.6. *Hybrid coding techniques*

The hybridization of deep learning and signal domain knowledge always accelerated researchers' interest in different fields of signal and image processing. Here, we are discussing some approaches which have combined the properties of traditional compression algorithm & DNNs to achieve better compression efficiency. However, Jiang et al. [152] introduced a compression scheme that connects the existing coding standards with DNNs as ComCNN & RecCNN, as shown in Fig. 9. ComCNN was used to generate a shrunk form of the original image and to preserve the structural information in the images. RecCNN is used to improve the quality of the image, which was being decoded by the codec. The algorithm outperformed JPEG, JPEG2000 & BPG, however not able to produce much-improved results with BPG as compared with JPEG & JPEG2000. Moreover, it failed to highlight edges and finer details in the image. In the similar way, Zhao et al. [153] proposed a virtual, hybrid technique, which includes training two CNNs with the JPEG codec module. Initially, the feature description neural network (FDNN) was used to get a reduced representation of the images. After that, the JPEG codec was exploited to compress the image further. Then for JPEG decoded output, a third post-processing neural network (PPNN) was exploited to reduce blocking artifacts. An additional network, virtual codec neural network (VCNN), was used for efficient back-propagation while training the network. A post-processing neural network (PPNN) was ex-

Table 11
Overview of scalable lossy compression techniques (CR:Compression ratio).

Paper	DNN	Train/Test	Performance Measures	Parameters, Runtime	Critical Findings/Remarks
Minnen et al. [134] 2017	CNN	6 million public images from Web/ Kodak	bpp=0.3, PSNR=27 dB		• Applicable for variable bit-rates
Balle et al. [86] 2018	CNN	Tecnick [142]/ Kodak	bpp=0.2, PSNR=28 dB	• 4.76 million	• The proposed hyperpriors are especially useful for medium to high bpp and optimized for L2/ PSNR evaluation
				• 76.8ms	• Surpassed JPEG, JPEG2000 & recent state-of-the-art methods
Cai et al. [135] 2018	CNN	Natural images from flickr.com/ Kodak	bpp=0.4, PSNR=32 dB, 45.12% bit saving over JPEG	• 4.76 million on 10 NVIDIA Tesla K80 GPUs	• Multiscale decomposition transform followed by an adaptive spatial bit allocation
				• 96.9 ms	• Laplacian mixture model is used over Gaussian model for quantization • GDN and IGDN are preferred over batch normalization • Surpassed BPG & JPEG2000 • Applicable for variable bit-rates
Ashok et al. [136] 2018	AE	MNIST, CLIC 2018	bpp=0.401	• 4457472	• Architecture for progressive image decoding to provide a variable sized latent factor for variable rate compression
				• 74s on NVIDIA Geforce 920M GPU	• Applicable for low bit-rates and superior than JPEG
Tang et al. [143] 2018	CNN	CLIC 2018, DIV2K	bpp=0.14978, PSNR=30.24 dB	• 4984420 bytes decoder • 30311917 ms on GTX 1080Ti GPU	• Artifact removal technique using multi-scale reshuffling network, over BPG • Applicable for variable bit-rates
Li et al. [137] 2018	AE	CLIC 2018, ImageNet	bpp=0.15, CR=80%	• 2775576	• Exploited and modified the already reported work of Li et al. [91] of the autoencoder network to generate 4-bit importance map • The method has outperformed the results obtained in [91]
Ororbia et al. [144] 2019	RNN	Places-365[a], RAISE-ALL[b]/ Kodak	bpp=0.37, PSNR=28.93 dB, SSIM=0.8425, MS-SSIM=0.9496	• 4457472	• Iterative decoding using trained RNN
					• 0.871, 1.095, 0.971 BD gain over JPEG, JPEG2000 & Toderici et al. [76] • Applicable for variable bit-rates
Wu et al. [139] 2020	GAN	ImageNet/ Kodak	bpp=0.1, PSNR=28 dB, MS-SSIM=0.94	• 10 million	• Importance map based variable bit-rates image compression to provide at least sufficient bits to non-significant regions
				• 50s on GeForce GTX 1080 Ti	• GAN based network needs more time for training • Outperformed JPEG, JPEG2000, BPG & the method mentioned in [90,114] • Applicable for variable bit-rates (0.05)
Chen et al. [140] 2021	VAE	MS-COCO, CLIC 2019/ Kodak	bpp=0.2, PSNR=28.5 dB	• 261.803 MB	• Separately trained for MSE and MS-SSIM respectively • Both time and space efficient • Slight drop in performance at lower bit-rates

[a] https://places2.csail.mit.edu/download.html [a]https://loki.disi.unitn.it/RAISE/download.html.

ploited to get an HR image by reducing artifacts. This technique was proved to be better than Jiang's technique where there are some side effects of training their first module, i.e., ComCNN. Hu et al. [154] combined the properties of the SVAC2 algorithm with the CNN network by giving excellent performance. The raw image was first converted to YUV-420 and given to the SVAC2 encoder. After decoding the YUV420 by SVAC2 decoder, the CNN network was used to filter only the Y channel. Then, chroma interpolation was used to get YUV 444 image to convert it into RGB further. The encoder was designed for two modes- Normal mode, which is used to encode the whole image as an intra image frame (I frame). The second model was designed as a multi-scale residual mode in which a small downsampled image was encoded as I frame and original images were encoded as P frame with the small encoded upscaled image as a reference. The architecture consisted of a feature extraction & reconstruction module. The feature extraction module consisted of the CNN network and reconstruction part, as inspired by ResNet [155]. On the other hand, Liu et al. [156] proposed a technique that tried to exploit the feature extracting property of neural networks with the traditional compression method. JPEG is optimized for the HVS, which is found to be a highly efficient compression framework. The author tried to remove the differences between HVS & DNN in terms of the frequency domain by designing a network for frequency component analysis. JPEG works on the principle to retain low-frequency components and truncate high-frequency details since human eyes or HVS are more sensitive to low frequencies (luminance) than high frequencies (chrominance). However, DNN understands the importance of features differently. The high-frequency features are important in some images, but if these features are truncated, accuracy can be reduced. Therefore, it was very challenging to design the HVS synchronized neural network, which can be optimized to get good quality images. Like Chen et al. [145] proposed soft decoding of JPEG images through a wavelet guided CNN network by introducing a dual pixel-wavelet domain deep CNNs dependent soft decoding network (DPW-SDNet). The block diagram for the proposed technique

Table 12
Overview of varibale bit-rate lossy coding techniques.

Paper	DNN	Train/Test	Performance Measures	Parameters, Runtime	Critical Findings/Remarks
Johnston et al. [146] 2018	RNN	6 million images from web/ Kodak	bpp=0.2340, PSNR=32.99 dB	• 9917504 • NVIDIA Tesla K80 GPU	• First deep learning technique based on spatial contextual entropy model • Use of hidden-state priming in RNN (as Gated Recurrent Unit (GRU) [151]), while training the network on weighted SSIM loss function • The method outperformed BPG, WebP, JPEG & JPEG2000 • Variable bit coding due to spatial adaptive binary coding
Kar et al. [147] 2018	CNN	CBIS-DDSM1[a] and Dream2[b]	bpp=0.1, PSNR=55 dB, SSIM=0.96	• 723092 on NVIDIA Quadro P6000 with 24 GB DDR5 RAM	• Deep convolutional autoencoder model followed by an adaptive arithmetic coding technique • Outperformed JPEG & JPEG2000 at high CR (×3000 (0.04 bpp)) without introducing ringing effects and distortion • Applicable for high compression factor & natural images
Lee et al. [65] 2018	AE	Yahoo Flickr Creative Commons 100 Million dataset[c]/ kodak	bpp=0.3, PSNR=31 dB	• 7.5s	• Combine a hyperprior and the context model to provide variable bit-rates image compression • Replacement of flow-density model with Gaussian model • Used for semantic analysis with improvement over BPG & JPEG
Choi et al. [148] 2019	CNN	ImageNet/ Kodak	bpp=0.2, PSNR=29 dB		• Lagrange multiplier and quantization bin size are used as controlling parameters function
Yang et al. [149] 2020	CNN	CLIC 2019/ Kodak	bpp=0.1, PSNR=25 dB	• 10.27 million	• Additional use of modulating and demodulating networks to obtain variable rate image compression by training the network only once • Outperformed JPEG, JPEG2000 & BPG

[a] https://wiki.cancerimagingarchive.net/display/Public/CBIS-DDSM [b] https://www.synapse.org/#!Synapse:syn3034857/wiki/74404 [c] https://projects.dfki.uni-kl.de/yfcc100m/.

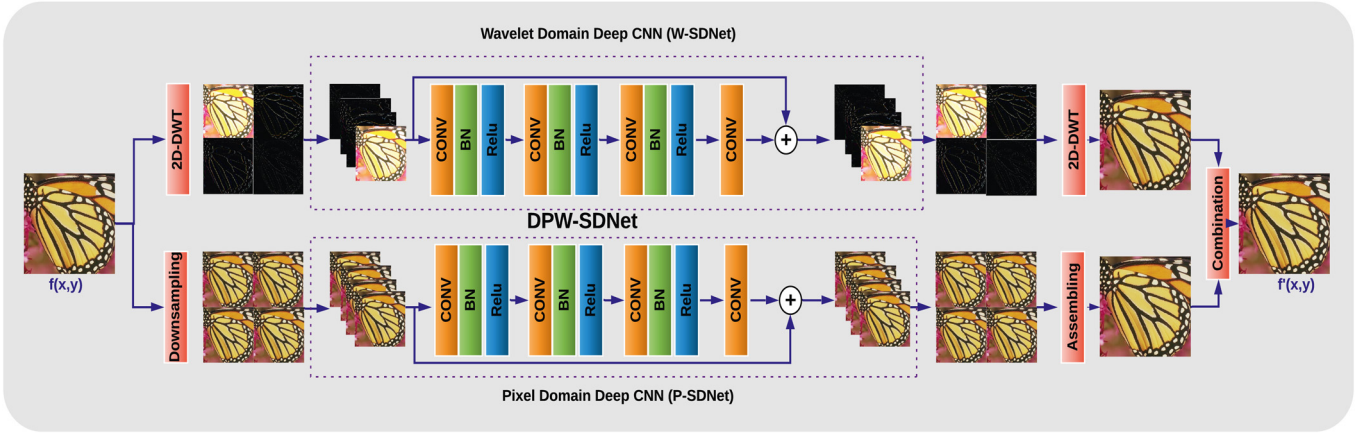


Fig. 8. Detailed Block Diagram of the compression technique in [145].

has been shown in Fig. 8. The network consisted of two different sections pixel & wavelet domain network. The first network, i.e., the pixel domain, helped to remove the blocking & ringing artifacts while the wavelet domain network helped to restore high-frequency components. Moreover, Akyazi and Ebrahimi [111] have used the idea of incorporating wavelet level processing as a pre-processing step with a deep CNN network. Then the preprocessed output is quantized, followed by entropy coding.

Inspired by the method mentioned in [85], the additive noise is used to combat the problem of zero gradients with the quantization step while training the network. The architecture has been shown in Fig. 5.Moreover. Mishra et al. [158] has proposed a wavelet transform-based compression-decompression algorithm to incorporate high-frequency components along with low-frequency components. The incorporation of high frequencies present in the image helped to preserve the fine details like edges, boundaries and reduce blocking artifacts significantly. The algorithm outperformed JPEG, JPEG2000, and other state-of-the-art artifact reduction techniques. A critical analysis of the methods discussed above and their compression performance comparison with traditional codecs has been given in Table 13 and Table 14, respectively.

2.2.7. *Generative modeling based*

Generative adversarial network (GAN) is the breakthrough in the history of DNN, which uses generator & discriminator. The adversarial loss used helps to produce natural and realistic images by taking random noise samples. Like, Snell et al. [159] exploited a loss function that was synchronized with the human vi-

Table 13
Overview of hybrid lossy coding techniques (QF: Quality factor).

Paper	Type	Train/Test	Performance Measures	Parameters, Runtime	Critical Findings/Remarks
Jiang et al. [152] 2017	CNN	BSDS500/ Kodak	bpp=0.204, PSNR=31.14, SSIM=0.851	• 708674 • 0.017s on NVIDIA Tesla K40c GPU	• Two different CNNs with JPEG codec to preserve the structural information and reducing blocking artifacts in the images • Outperformed JPEG, JPEG2000 & BPG, however not able to produce much-improved results with BPG as compared with JPEG & JPEG2000 • Failed to highlight edges and finer details in the image • Applicable for low bit-rates
Zhao et al. [153] 2017	CNN	BSDS500/ Set14 [157]	bpp=0.157, PSNR=31.31 dB, SSIM=0.853	• 19968768	• Countered the non-differentiability problem of quantization • Separately trained the three modules with SSIM over end-to-end network training • Applicable for all bit-rates
Hu et al. [154] 2018	CNN	CLIC 2018, BSDS200[94]	PSNR=30.84 dB		• Combination of SVAC2 algorithm with CNN network • Outperformed JPEG, JPEG2000 & WebP • Applicable for low bit-rates
Liu et al. [156] 2018	CNN	ImageNet	3.5 times high comp rate than JPEG	• 60 million	• Removed the differences between HVS and DNN by designing a network by analysing for high-frequency components • Variable quantization step on the basis of range of importance features
Chen et al. [145] 2018	CNN	BSDS500/ Live1	QF=10, PSNR=29.53 dB, SSIM=0.8210	• 1700736 • 1.2 ∼ 2s on GeForce GTX 1080 Ti GPU	• Soft decoding of JPEG images using wavelet transform and DNN which outperformed JPEG & ARCNN • Applicable for low bit-rates
Akyazi et al. [111] 2019	CNN	CLIC 2018	bpp=0.138, PSNR=23.85 dB, MS-SSIM=0.8817	• 275702	• Incorporated wavelet level processing as a pre-processing step with deep CNN network • Surpassed JPEG & JPEG2000 • Applicable for low bit-rates (<0.15bpp)
Mishra et al. [158] 2020	CNN	ImageNet/ Kodak	bpp=0.2, PSNR=28.8 dB, SSIM=0.82	• 708678 • 7.5 ms on 12 GB GPU Titan X Pascal.	• Single level Haar wavelet is used for obtaining high frequency components • Better artifacts reduction techniqe for low bit-rate compression • Surpassed JPEG, JPEG2000 & various deblocking techniques

Table 14
Compression performance comparison of hybrid coding schemes with traditional codecs.

Algorithm	bpp	PSNR (dB)	MS-SSIM
JPEG	0.155	22.1	0.7568
BPG	0.153	29.2	0.9425
Alexandre et al. [89]	0.126	26.3	0.9242
Akyazi et al. [111]	0.138	23.85	0.8817
Mishra et al. [158]	0.133	26.29	0.9088

sual system (HVS), i.e., MS-SSIM. The MS-SSIM being differentiable could be easily used during back-propagation. This loss provided smooth gradient flow and was inexpensive to compute. The experiments were performed on deterministic and probability autoencoders and compared when using mean absolute error MAE, MSE & MS-SSIM. It was found that MS-SSIM performed better than MSE, MAE, in terms of the visual quality of the image. The method was able to produce quite high-quality, optimized images when undergone super-resolution processing. Similarly, Dosovitskiy and Brox [160] had started the work with the motivation that there is very little research on loss functions that are required for the task of efficient image generation. The author demonstrated that for the perceptual similarity of images obtained, finding only the perfect location of essential objects with sharp edges is not sufficient. Furthermore, the idea was focused on calculating the images' statistical properties and their invariance to irrelevant transformations. Therefore, the author's focus was diverted to find the distance measurement in a feature space. The similarity in features does not always convey image similarity due to the contractive property of feature representations. The author has proposed a particular loss function (DeePSiM), mostly for autoencoders. Usually, MSE is used as the loss function for an autoencoder, but that loss function leads to blurry images since it is a pixel-to-pixel difference. The addition of two new ingredients to the loss function resulted in significantly sharper-looking images. The final loss function was weighted sum of three terms: feature loss L_{feat}, adversarial loss L_{adv}, & pixel space loss L_{img} i.e., $L = \lambda_{feat}L_{feat} + \lambda_{adv}L_{adv} + \lambda_{img}L_{img}$, where λ_{feat}, λ_{adv} & λ_{img} are the weights associated with features, adversarial & pixel loss respectively. The experiments concluded that using the loss in feature space alone would not be enough as that tends to lead to over-pronounced high-frequency components in the image (i.e., too strong edges, corners, other artifacts). To decrease these high-frequency components, a natural image prior was usually used, which is here the adversarial loss (which learns a good prior). The proposed generative scheme was novel and interesting and a breakthrough in image compression, which was almost saturated. Later on, Im et al. [161] proposed a network inspired by Gatys et al. [162] to generate a new image that matches image features & texture features by training layers of the pre-trained CNN. The sequential modeling was executed by GAN, i.e., LAPGAN [163]. The proposed approach was an interface between

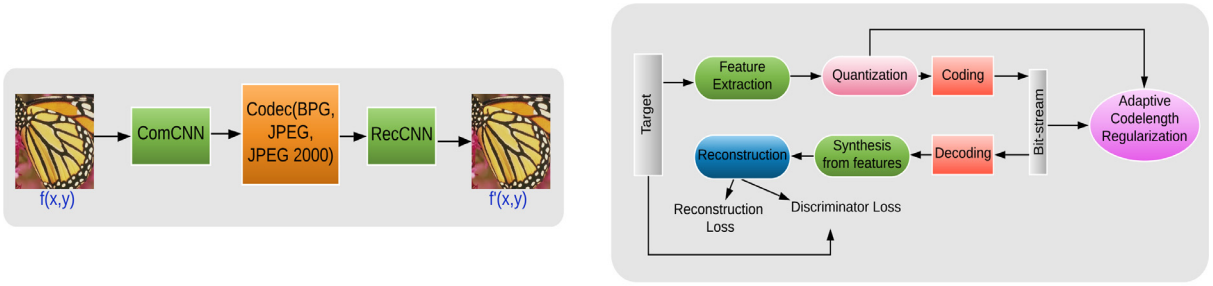


Fig. 9. (a) Hybrid compression framework using CNN and conventional codecs [152], (b) Encoder & decoder compression framework with feature extraction [114].

DRAW [74] & gradient-based optimization. The difference between the proposed GAN & vanilla GAN is that in the former, the generator is provided with noise samples in the form of sequences from the prior distribution. It produced a higher quality image sample than a corresponding single-step GAN model. The results were assessed based on the generative adversarial metric (GAM) [164], which does not compare the adversarial network but with DRAW or VAE. Moreover, Rippel and Bourdev [114] had proposed a compression framework composed of a generator and a dedicated discriminator. The GAN-based architecture has been shown in Fig. 9. For the first model, the wavelet decomposition was exploited as non-linear extractors, individually for each case via some parameterized function. Then inter-scale alignment was used to leverage information shared across different scales. Then, the obtained tensor was quantized for further coding. The feature extractor helped in reducing redundancies through pyramidal decomposition and inter-scale alignment modules. The coding being lossless allowed compressing the quantized tensor through bit-plane decomposition and adaptive arithmetic coding (AAC). The adaptive code length regularization modifies the expected code length to a specified bit rate. The loss calculated at the discriminator forced to generate good quality images by reducing differences between reference and generated images. The loss produced by the discriminator encouraged to generate better reconstructions by decreasing loss between original & recovered images. Apart from these, Tolunay and Ghalayini [165] proposed a deep generative model in such a way that was encoder deficient and having a reversed generator of GAN. The novelty of the approach was latent space representation recovery through the GAN reversal scheme. With the same compression ratio as in JPEG, the proposed scheme outperformed JPEG in terms of SSIM. Still, much research work is needed for designing a more perceptual performance metric for image compression. Then, Kang et al. [97] proposed a network consisting of a generator, discriminator, and a network interfacing both generator and discriminator known as the transformer. The generator tried to produce images based on random noise. The transformer enabled the generated data and training data to be in the same domain, as generated data is in compressed form, and training data is the image, so the task of the transformer was to convert that compressed data into a raw image. A critical analysis of the methods discussed above is given in Table 15. Additionally, we have compared the work in [114] with BPG and the saliency-based algorithms [65,110] as shown in Table 16.

2.2.8. *Intra coding techniques*

The technique is used to predict the future pixels or blocks based on the pattern of the previous two pixels or blocks of an image. This property is majorly used for lossless compression schemes to remove the spatial redundancies. Till now, high-efficiency video coding (HEVC) is the benchmark algorithm for picture compression in this domain. HEVC is efficient than its precursor scheme H.264, with a bit rate saving of approximately 17% as reported in [167]. Dai et al. [168] proposed a residual learning-based CNN model with the provision of having variable filter size. The variable filter size helped speed up the training process with a higher bit rate reduction at less cost of computation. On the other hand, Li et al. [169] proposed a fully connected network to perform intra-prediction in adjacent blocks of an image. Initially, the original images are compressed at different quantization rates. The idea was to utilize the properties of luma & chroma components. We have another technique proposed by Li et al. [170] for compressing videos, especially at low-bit rates. The proposed architecture is comprised of block-wise CNN based on down or up-sampling for rate-distortion optimization. The five-layer CNN module used is found to perform better than super-resolution-based image compression techniques like SRCNN & VDSR. In 2019, Hu et al. [171] also proposed an intra-prediction using the sequential learning capability of RNN. Zhu et al. [172] proposed GAN based intra-prediction coding technique similar to inpainting. Dumas et al. [173] reported a fully connected neural network based intra-prediction technique which use masks of random size for training. On the other hand, Sun et al. [174] proposed intra-prediction technique using multiple neural networks. Sun et al. [175] coined another intra-prediction technique based on neural network with variable block size. A critical analysis of the methods discussed above is given in Table 17.

2.2.9. *Inpainting based coding*

Inpainting is the method of recovering the missing regions of an image from the perspective of image restoration. The missing region enables the system to store less information and that lost information can be produced later with the help of inpainting. In 2017, Baig et al. [176] tried to improve the performance of lossy compression models using the inpainting method. Conventional encoders are directed to reduce the error (residual); however, the residual encoder was used to predict the original data after every residual iteration. The architecture was designed in a way to fill the missing component in the image by exploiting redundancies in adjacent regions. Hence, inpainting was used with compression so that the amount of information that needs to be stored would become less for full image reconstruction. The algorithm is used for partial context inpainting. The network was trained with ImageNet, tested with Kodak and exhibited 63.01% & 17.9% bit saving over JPEG and Toderici [76]. Moreover, a file size reduction of 60% is achieved with the application of this algorithm.

2.2.10. *Colorization based coding*

Colorization is the process of predicting chrominance channels of the images from their luminance counterpart. There are various colour space available like LAB, YUV [177], YCbCr [178,179] etc. For the task of image compression, YCbCr is generally preferred due to the least correlation between its Y, Cb, and Cr channels.

Table 15
Overview of generative modelling (GAN) based lossy compression methods (CR:Compression ratio,QF: Quality factor).

Paper	DNN	Train/Test	Performance Measures	Parameters, Runtime	Critical Findings/Remarks
Snell et al. [159] 2015	AE	80 million Tiny-Images data set[a]/ Set14	PSNR=27.47 dB, SSIM=0.7610		• Trained super-resolution architecture with MS-SSIM based loss function for training as its synchronisation with HVS • The proposed algorithm was less computationally expensive
Dosovitskiy et al. [160] 2016	VAE	ImageNet		• 138357544	• Proposed the DeePSiM loss with a normal autoencoder resulted in sharp contrast reconstructed images • Optimized the network with weighted sum of MSE, feature space and adversarial loss • To reduce the reconstruction images having over-pronounced high-frequency components, adversarial loss is used
Im et al. [161] 2016	GAN	MNIST, CIFAR 10		• 142341692	• The proposed approach was an interface between DRAW [74] and gradient-based optimization • The sequential modeling was accomplished through LAPGAN • The assessment was made using GAM metric
Rippel et al. [114] 2017	GAN	Yahoo Flickr Creative Commons 100 Million dataset[b]	bpp=0.25, SSIM=0.97	• 18.5 ms on a GTX 980 Ti GPU	• Superior over JPEG & WebP • Multiple adversarial training for wavelet pyramidal decomposition and codelength regularization using autoencoder as a generator • Generated image file size of 2.5 times less than JPEG & JPEG2000 and 2 times, 1.7 times small file than WebP and JPEG • Applicable for very low bit-rates
Galteri et al. [166] 2017	GAN	MS-COCO[c]/ Live1	QF=10, PSNR=27.29 dB, SSIM=0.773	• 7128160 • NVIDIA Titan X PASCAL GPU	• Training of network with SSIM and adversarial loss to reduce blurring • Surpassed JPEG and ARCNN • Applicable for average bit-rates
Tolunay et al. [165] 2018	GAN	Celeba dataset	bpp=0.2152, CR=63.34, PSNR=21.92 dB, SSIM=0.7073	• 142341692	• Used GAN was encoder deficient and having reversed generator of GAN • Reverse GAN was optimized using SSIM loss • Latent space representation recovery through GAN reversal scheme with MS-SSIM • Applicable for low bit-rates
Kang et al. [97] 2019	GAN	CIFAR 10[d]	QF=100		• Network consisting of a generator, discriminator and a network interfacing both generating and discriminator known as a transformer to produce images based on random noise • The network was trained with adversarial loss, and random noise was used for quantization

[a] https://groups.csail.mit.edu/vision/TinyImages/ [b] https://projects.dfki.uni-kl.de/yfcc100m/ [c]https://cocodataset.org/#download [d] https://www.cs.toronto.edu/~kriz/cifar.html.

Table 16
Compression performance comparison of Agustsson's approach [110] with BPG and benchmark compression techniques.

Attribute/ Algorithm	BPG	Rippel et al. [114]	Minnen et al. [64]	Agustsson et al. [110]
Learning	No	Yes	Yes	Yes
Arithmetic coding	Adaptive	Adaptive	Adaptive	Static
Context model	CABAC	Autoregressive	Autoregressive	None
Outperformed SOTA in PSNR	No	No	Yes	No
Outperformed SOTA in MS-SSIM	No	No	Yes	No

Compressing a single channel help to reduce the two-third of the space required for the information to be stored. But colorization using compression has been facing several challenges for years and years. In 2017, Baig and Torresani [178] reported that the colorization process requires the color filling in gray pixels based on available chroma or colors. However, compression requires generating colors closed to the ground truth. Colorization helps in compression as only a single channel needs to be processed. The proposed model is a two-way model; the network is first used to generate the colors for each pixel. Then, after that, a process was adopted to enhance the probability of mapping color to a particular pixel within low-cost colorization. The network was trained and tested using ImageNet/ CIFAR 100 available at.[2] It produced multiple colorizations for a gray image having a trade-off between image size and color fidelity. Multiple side information is required to store for each probable resultant. The algorithm has faithfully reconstructed colors better than Cheng and Vishwanathan [180] and He

[2] https://www.cs.toronto.edu/~kriz/cifar.html.

Table 17
Overview of intra-coding lossy techniques (QP: Quantization parameter).

Paper	DNN	Train/Test	Performance Measures	Parameters, Runtime	Critical Findings/Remarks
Dai et al. [168] 2017	CNN	BSDS500	QP=37, PSNR=32.24 dB, 4.6% BD rate over HEVC	• 54512 (214 KB) • 0.9s on NVIDIA Tesla K40C GPU	• Variable filter size used for training CNN with MSE which helped in speeding the training process • Applicable for adaptive bit-rates
Li et al. [169] 2018	CNN	New York city library images	QP=22, 3.4% bit saving than HEVC	• Training on Tesla K40m GPU • Testing on Intel Xeon E7-4870 CPU	• Fully connected network to perform intra-prediction in adjacent blocks of an image • Relatively high complex model and applicable for low bit-rates
Li et al. [170] 2018	CNN	New York city library images	QP=22, 3.4% bit saving than HEVC	• Intel Xeon E7-4870 CPU	• Blockwise CNN based on down or upsampling for rate-distortion optimization • Luma and Chroma based down sampling helped in reducing computational complexity • Applicable for low bit-rates
Hu et al. [171] 2019	RNN	DIV2K	2.5% bit saving than HEVC	• NVIDIA GTX 1080 GPU	• Intra-prediction scheme using the sequential learning capability of RNN • The network was trained with Sum of absolute transformed difference (SATD) loss function • Superior over JPEG & JPEG2000 • Applicable for low bit-rates, flexible for variable block size in HEVC.
Zhu et al. [172] 2019	CNN	UCID		• 167 s on Tesla K80 GPU	• Problem of intra prediction is formulated as an inpainting task
Dumas et al. [173] 2019	CNN	Video Sequences	Gain is on average 0.99% larger than those of prior neural network based methods		• Neural networks trained on undistorted contexts generalize well on distorted contexts, even for severe distortions.
Sun et al. [174] 2020	CNN	4K sequences	2.6%, 3.8%, 3.1% BD-rates for Y, U, and V channels respectively		• Enhances the intra prediction by using multiple neural network modes (NM).
Sun et al. [175] 2020	CNN	Video Sequences		• Intel Core i7- 7820X CPU@3.60GHz with 32GB memory	• Fully NM based intra coding • A coding framework with NM based on the best mode probability analysis

[a] https://digitalcollections.nypl.org/.

et al. [181] with a fast training process. It is found that the algorithm is quite expensive with a model size of 105 MB and 7.2 s execution time. The algorithm is applicable for low bit-rates as generating PSNR=29.15 dB at bpp=0.02.

2.2.11. *Inter-channel correlation based*

Nowadays, the correlation between R, G & B channels for any colored images is also being used to achieve image compression. This is also one of the right choices as an image compression researcher. In 2018, Cui and Steinbach [182] focused on the problem of blocking & ringing artifacts at low bit rates. With the previous studies, it has already been known that the G channel has higher quality at a low bit rate. So a network was designed to exploit this quality of the G channel to guide the reconstruction of R & B channels. The decoder side was designed for exploiting inter-channel correlation. The decoded image was split into R, G & B. The architecture consisted of 3 stages, & the channel-wise loss function was used to optimize the post-processing network. The network is trained and tested with CLIC 2018 dataset producing PSNR=33 dB, MS-SSIM=0.97 at bpp=0.12. It is the first deep CNN and post-processing approach that exploited inter-channel correlation while decoding. The algorithm is applicable for very low bit-rates with 1787904 trainable parameters and 14.7 s execution time.

2.2.12. *Simultaneous compression & retrieval*

These are the techniques in which the same network is used to provide image compression and distortion less retrieval. It isn't straightforward to obtain the exact mirror image of the original image, however, DNNs somehow succeeded in achieving good performance in this direction also. Like in 2017, Zhang et al. [183] used DNN, which was trained for extracting features and generate latent space. The image encoder & feature extractor were combined and fine-tuned with triple weighted loss function for the application of content-based image retrieval (CBIR) coding [184]. The algorithm generated a file of small size with a high CR (5.3), bpp=0.18, PSNR=23.39 dB. The total number of trainable parameters is 138344128 making the approach quite expensive.

2.2.13. *Transform coding*

Transform is the first and foremost step for the benchmark image compression standard JPEG algorithm, which is used to remove the inter-pixel redundancies. Likewise, it is a perfect idea to combine the feature extracting property of CNN with the DCT transformation. Likewise, in 2018, Liu et al. [185] came up with learning-based DCT. The DCT coefficients were quantized using multi-level quantization than conventional binary quantization. L_1 norm has been exploited for quantization coefficients required for obtaining rate-distortion optimization. The DCT, quantization, and transform coding were integrated into a deep network itself. The work presented a block-based transform, which increased the quality of reconstructed images when checked with the BD rate by claiming that training-based transform is much better than fixed DCT. The network is trained with UCID [186] dataset, tested with Kodak and generated PSNR=31.95 dB at bpp=0.272, 38.03% & 56.66% BD-rate reduction over JPEG & Toderici [76] respectively. It is applicable for low bit-rates with 2806176 parameters. Later on, in 2019,

Tan et al. [187] proposed a hybrid technique of reconstructing DCT downsampled images using a super-resolution network.

2.2.14. *In-loop & out-of-loop filtering*

The quantization mechanism, followed by the truncation of DCT coefficients, generally produces the blocking artifacts. So in loop or post-filtering is a better choice to reduce these artifacts, thereby increasing the compression efficiency. Also, for reducing compression artifacts, various de-blocking filtering techniques have been proposed. They generally helped to increase the quality of reconstructed images by reducing blocking & ringing artifacts.

In the direction of reducing this challenge, Dong et al. [188] proposed an "easy-to-hard" transfer learning approach of training a shallower network and then finetune it using a deeper network. However, the author observed that "it is difficult to train a network containing more than four layers for low-level vision applications". Moreover, Cavigelli et al. [190] stated that it is generally known fact that the differences between the original & reconstructed image at higher compression rates are quite visible due to the specific artifacts. The probability of artifacts generation existed during compression as the image was first divided into blocks or tiles as in JPEG. The idea was basically to focus on removing ringing, and blocking. On the other hand, Galteri et al. [166] proposed an artifacts reduction technique based on residual networks using GAN. Several schemes were proposed for better image quality [77,85,90,114] in which Mentzer et al. [90] technique of using context model with pixel-CNN [91] for entropy coding achieved good results.

Batch normalization, which is also a type of regulation, was used to recalculate and shift the mean and variance of intermediate features. YCoCg color space was used for BPG with L_1 loss. The proposed scheme proved to be removing artifacts for BPG compressed images. On the other hand, Aytekin et al. [192] focused on the binarization part of variable bit neural network image compression. Several artifacts removal techniques are available in literature based on deep learning, GANs, RNNs, CNNs & residual networks like presented in work of Galteri et al. in 2018 [166], Oord et al. [202], Dahl et al. [203], Santurkar et al. [204], Baig et al. [176], Yu et al. [205], Svoboda et al. [206], Jain and Seung [207], Zhang et al. [208], Mao et al. [209] and Theis et al. [77]. Maleki et al. [194] proposed technique was based on the LAB space, which was chosen to reduce artifacts. The concept of block CNN was to divide an image into 8×8 blocks & analyze each block separately. The author observed that the residual estimation was based on context, and the artifacts depend upon the location of the pixel in the block. Similarly, Gonzalez et al. [210] proposed a probabilistic data-fitting based on the noisy, compressed image. It is a joint denoising and decompression method to fit data, latent prior representation, and quantization process. It is a bayesian scheme used to combat noise and overcome JPEG2000. The scheme is applicable for large bit-rates producing PSNR=39.52 dB, SSIM=0.9241 at bpp=2 by training the network on random images from the Web. Later on, Kirmemis et al. [196] introduced a compression artifact removal scheme by applying a post-processing module on BPG compressed images. The author was succeeded by proposing the network where the compressing and decompressing of images depends on the availability of required computational resources and achieved excellent results. On the other hand, Tang and Luo [143] proposed the algorithm to remove compression artifacts in reconstructed images after compression using the concept of a multi-scale reshuffling network. The up-scaling & down-scaling network was separately used in encoder & decoder. The deep network used to extract important features & multi-scale shuffling network was utilized to get HR images from their LR version. Initially, the images were converted from RGB to YCbCr. The compression network was trained with L_1 loss function. Moreover, Ororbia et al. [144] proposed an iterative decoding using trained RNN and found it to produce better perceptual quality images than JPEG and JPEG2000. Later on, to combat the problem of high-frequency components using wavelet transform, Ma et al. [211] proposed an efficient method of CNN-based arithmetic coding. The method exploited the correlation among the individual wavelet coefficients and also among sub-bands using RNN followed by CNN-based arithmetic coding. For further improvement in the quality of the images obtained after decompression, a second CNN module for post-processing is introduced. The network is trained with the DIV2K dataset and exhibits PSNR=32 dB and 14.66% saving over JPEG2000 at bpp=0.3 (low-bit rate). Kim et al. [212] reported a pseudo blind method to find the quality factor of a given compressed image and then selecting a Inception module based network which is trained with similar quality factor. In early 2020's Jin et al. [199] exploited the frequency characteristics of the image to reduce the artifacts after decompression with residual learning to speed up the training. Very interestingly, Liu et al. [200] have presented a comprehensive survey of traditional and learning-based compression artifacts removal techniques. The author investigated the existing state-of-the-art techniques on the Large-Scale Ideal Ultra high-definition 4K (LIU4K) dataset. Basically, the survey considered four types of artifact removal schemes namely filter-based methods, probabilistic prior-based methods, learning-based JPEG artifacts removal methods, and learning-based loop filter methods. Later on, Li et al. [213] reported a single model for handling a wide variety of quality factors using restoration and global branch method, which helped to remove ringing artifacts. Recently, Yeh et al. [201] proposed a blocking artifact reduction technique exploiting multiple loss functions at various multiple layers instead of using loss functions only at the output obtained at the last layer. The multi-scale fusion model is learned for image super-resolution and blocking artifacts estimation. The principle is to calculate the blocking artifacts by estimating the difference in artifacts of the original image and the artifacts of its downsampled forms, which relatively contains fewer artifacts. A critical analysis of the methods discussed above is given in Table 18. Moreover, the computational cost comparison of various end-to-end coding schemes, down-up sampling methods, scalable image compression schemes, variable bit-rate coding, hybrid schemes, generative modeling with the traditional codecs has been shown in Table 19. The critical analysis based on coding gain efficiency in terms of BD rate has been shown in Fig. 10. The computational cost of some of the benchmark algorithms under various categories is summarized in Fig. 11. Based on the discussion so far, we have classified the algorithms according to coding bandwidth availability, As shown in Fig. 12, a summary based on the compression performance along with their applicability on available bandwidth (extremely low, low, mid, high bit-rates) has been presented. The figure also throws light on the techniques which are very less and highly explored, so that a new researcher can orient the work in an innovative and less explored directions.

3. Critical findings and discussion

The paper presented a comprehensive survey of the best deep learning architectures for image compression. All the architectures are optimal for a specific application in the image compression domain. Even though deep learning architectures have covered and succeeded in achieving excellent results in almost all applications related to image processing, there is much scope for further advancements and improvements. Various techniques have been discussed here based on some principle, in which some techniques can be better for some specific scenario, while some may work better for another task. As there is a "no-free-lunch theorem", we can't give the best rating for any specific algorithm. However, ac-

Table 18
Overview of in-loop and out-of-loop filtering lossy coding techniques (QF: Quality factor).

Paper	DNN	Train/Test	Performance Measures	Parameters, Runtime	Critical Findings/Remarks
Dong et al. [188] 2015	CNN	BSDS500 [94]/ Live1	QF=10, PSNR=28.98 dB, SSIM=0.8217	• 5 million	• "Easy-to-hard" approach for low-level vision problems • Outperformed JPEG and method in [41,189]
Cavigelli et al. [190] 2017	CNN	BSDS500/ Live1	bpp=0.4, QF=10, PSNR=29.44 dB, SSIM =0.833, applicable from QF= 40-76	• 5144000 on 2 NVIDIA Titan X Maxwell GPUs • 2.42s on NVIDIA GTX 1080	• Deep CNN with some residual connections to provide super-resolution • Training on multi-scale edge aware loss function as inspired by [188,191] surpassed JPEG and ARCNN. • Use of multiscale MSE + edge-aware loss function over MSE to avoid artifacts
Galteri et al. [166] 2017	CNN	MS-COCO[a]/ Live1	QF=10, PSNR=27.29 dB, SSIM=0.773	• 7128160 • NVIDIA Titan X PASCAL GPU	• Training of CNN on SSIM and adversarial loss to reduce blurring • Surpassed JPEG and ARCNN • Applicable for average bit-rates
Mentzer et al. [90] 2018	CNN	ImageNet/ Kodak	bpp=0.15, MS-SSIM=0.93	• 9561472	• 3D-CNN is exploited for learning a conditional probability model for a multiresidual-block-based network • Use of weighted normalization in training and residual pathway • Outperformed JPEG, JPEG2000, & BPG • Applicable for variable bit-rates
Aytekin et al. [192] 2018	CNN	CLIC 2018	bpp=0.148, PSNR=27.92 dB	• 1368000	• Variable bit-rates image compression by adding random noise to solve the non-differentiablilty problem of quantization • A deblocking network similar to U-Net [193] was used to reduce blocking artifacts • L_2 normalization used to train decoder • A new entropy loss function for auto-encoder is also designed • Applicable for variable bit-rates
Maleki et al. [194] 2018	CNN	PASCAL VOC [195]/ Kodak	bpp=0.5, PSNR=32 dB, SSIM=0.87	• 7865420	• Block Processing for calculating residual and reducing artifacts • Applicable for low bit-rates
Kirmemis et al. [196] 2018	CNN	CLIC 2018	bpp=0.15, QP=40, PSNR=30.14 dB, MS-SSIM=0.948	• 1.35 million • 723s on i7-3630QM 2.40 GHz CPU	• The method formulated as linear programming problem found to be suitable over EDSR [197] or SR-ResNet [198] • SELU activation function is used over ReLU for faster learning • Applicable for low bit-rates & surpassed BPG
Tang et al. [143] 2018	CNN	CLIC 2018, DIV2K	bpp=0.14978, PSNR=30.24 dB	• 4984420 bytes decoder • 30311917 ms on GTX 1080Ti GPU	• Artifact removal technique using multi-scale reshuffling network, over BPG •
Ororbia et al. [144] 2019	RNN	Places-365[b], RAISE-ALL[c]/ Kodak	bpp=0.37, PSNR=28.93 dB, SSIM=0.8425, MS-SSIM=0.9496	• 4457472	• Applicable for variable bit-rates • Iterative decoding using trained RNN • 0.871, 1.095, 0.971 BD gain over JPEG, JPEG2000 & Toderici et al. [76]
Jin et al. [199] 2020	CNN	BSDS500/ Live1	bpp=0.1, PSNR=28.41 dB, SSIM=0.7493	• NVIDIA Titan X GPU with 12GB RAM	• Applicable for variable bit-rates • Surpassed JPEG and other denoising and restoration methods
Liu et al. [200] 2020	CNN	LIU 4K	QF=10, PSNR=32.33 dB, PSNRB=32.30 dB, SSIM=0.8520, MS-SSIM=0.9513	• 9400180, 22.67 MB • 32.48ms	• A critical analysis of compression artifact reduction techniques
Yeh et al. [201] 2021	CNN	BSDS500, DIV2K/ Live1	QF=10, PSNR=29.30 dB, SSIM = 0.8179, PSNRB=29.73 dB	• 625806 • 0.68s on 16 GB NVIDIA GeForce GTX 1080 GPU	• Used multi-loss function at multiple layers of CNN network • Good artifact reduction technique

[a] https://cocodataset.org/#download [b] https://places2.csail.mit.edu/download.html [c] https://loki.disi.unitn.it/RAISE/download.html.

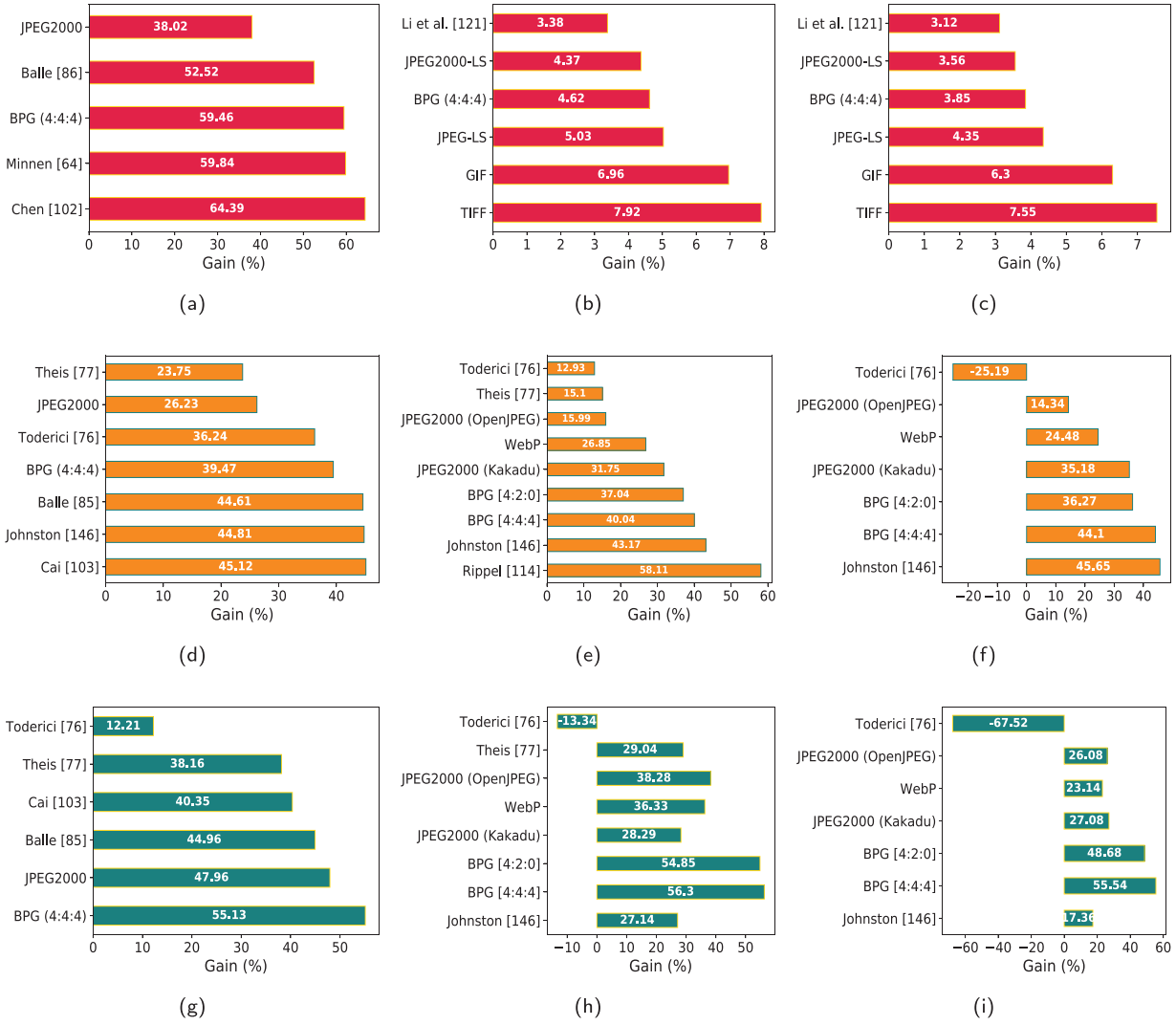


Fig. 10. (a) Coding gain efficiency (BD rate) of various methods considering JPEG as reference (anchor), (b) Bit-rate comparison of Li's [121] approach with lossless image compression standards on the Kodak dataset, (c) Bit-rate comparison of Li's [121] approach with lossless image compression standards on the Tecnick dataset, (d) MS-SSIM coding performance comparison (BDBR-MS-SSIM) of different methods compared to JPEG (anchor) on Kodak, (e) Bit-saving (BD) wrt. MS-SSIM for various schemes against JPEG (anchor) for Kodak dataset, (f) Bit-saving (BD) wrt. MS-SSIM for various schemes against JPEG for Tecnick dataset, (g) PSNR coding performance comparison (BDBR-PSNR) results of different methods compared to JPEG (anchor) on Kodak, (h) Bit-saving (BD) wrt. PSNR for various schemes against JPEG (anchor) for Kodak dataset, (i) Bit-saving (BD) wrt. PSNR for various schemes against JPEG (anchor) for Tecnick dataset.

Table 19
Run-time complexity comparison of various benchmark DNN approaches with standard codecs.

Algorithm	Encoding Time (ms)	Decoding Time (ms)
JPEG	18.6	13.0
JPEG2000	367.4	80.4
BPG	449.7	220.9
WebP	67.0	83.7
Toderici's [76]	1606.9	1079.3
Balle's [85]	242.12	338.09
Balle's [86]	64.7	12.1
Rippel's [114]	8.6	9.9
Mishra's [158]	3.5	4
Cai's [103]	75.12	73.23
Li's [119]	24	32
Li's [121]	74	984
Cai's [135]	79.5	17.4
Ashok's [136]	42	32

cording to the resources available and the type of functionality required for a particular application, we can diligently choose one of them. We have highlighted some critical observations based upon certain experimental results and limitations, which are discussed below.

3.1. On the basis of architecture used

- CNN based networks [53,64,82,86,91–93,96,98–100,103,104,107,111,113,115,118–122,134,135,143,145,147–149,152–154,156,158,166,168–170,182,185,188,190,194,199–201] which use simple feature learning for the better reconstruction of data are highly efficient and faster methods.
- RNN based networks [72,73,76,79,101,144,146,171] uses iterative training for sequential modelling of data. These are basically efficient for audio, text, or 1D signal processing where future samples are likely to be predicted. However, it was observed that prolonged training in an RNN network leads to distorted behavior as training starts to follow gradient ascent-like behavior rather than gradient descent. That is why only a few methods for image compression are available using RNN based architectures.

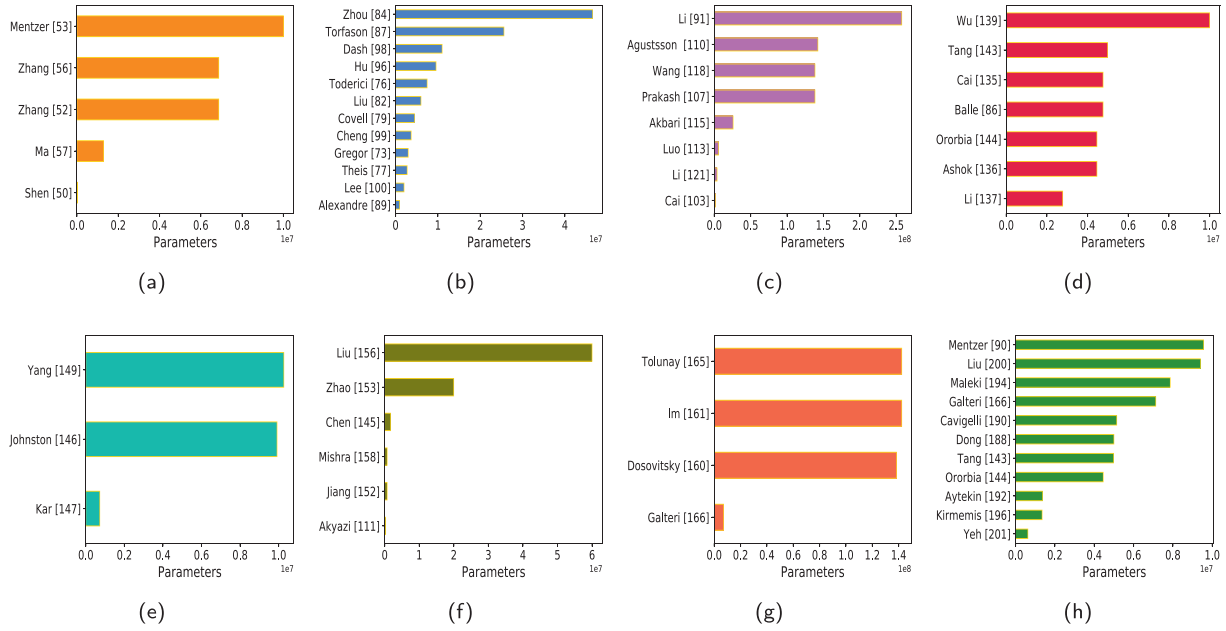


Fig. 11. Training parameter comparison for compression schemes discussed under various categories (a) Lossless compression schemes (b) End-to-end coding schemes (c) Saliency based coding schemes (d) Scalable compression schemes (e) Variable bit-rate coding schemes (f) Hybrid coding schemes (g) Generative modelling schemes (h) In-loop & Out-of-loop filtering based schemes.

- GAN based compression framework [52,54,87,97,110,114,139,161, 165,166] effectively boosts up the compression efficiency but at the cost of heavy computations. Also, sometimes adversarial training seems to fabricate/generate some synthetic features at the time of the generation of the image, which is not desirable.

3.2. On the basis of loss function, utility and application

- The lossless approaches discussed so far in which information loss is not permissible are widely used in medical and satellite applications [50,52–54,56,214].
- Some of the models being of very lightweight have found their utility in mobile applications [75,80,89,93, 103,111,137,147,152,166,201].
- CNN based approaches [88,101,107,114,115,119,122,135,145,152, 168,200,201,215–217][215–217] being very fast and highly effective, can be used for real time professional applications.
- Optimizing the compression network with a weighted sum of two or three loss function specifically L_∞ [52,56] as one of the constraints proved to increase the compression gain and performance with the least information loss.
- Much improvement in compression efficiency both in terms of rate and distortion is achieved, if the compression network is optimized on the subjective metric (SSIM, MS-SSIM, or weighted sum) over objective metric (MSE, PSNR) [52,77,89,92,98,115,119,121,122,146,160]. This is because subjective measures are synchronized with HVS, and the end-user for assessing any compression algorithm is human only.
- Saliency based compression approaches [64,91,103,107,110,113, 115,118–122,218][218] scalable image compression algorithms [86,134–137,139,140,143,144], variable bit-rate coding schemes [65,146–149] with "end-to-end" training aid in generating images at extremely low bit rate with the improvement in subjective measures SSIM & MS-SSIM. But, computational cost increases due to extra trainable parameters on account of an additional arrangement of saliency prediction network.
- There is very little research in the area of exploiting DNN based colorization for compression as the researchers are facing difficulty to produce true color images [178,179]. As the color of reconstructed images does not exactly match with the ground truth, but rather have the ability to generate multiple plausible colorizations for a grayscale decompressed image.
- Earlier, "end-to-end" training requirement in hybrid approaches [111,145,152–154,156,158] (conventional module with DNN) was found to be difficult, but now the researchers have broken this wall also.
- Compression using inter-channel correlation [182], intra-coding [168–171], in-painting [176], and transform coding [185,187] using DNN is one of the new, exciting problem and can be explored more in future.

3.3. On the basis of compression performance, computation cost and execution time

- It was observed from Table 2 that, Zhang's [56] approach is giving the best improvement in terms of PSNR along with minimum reconstruction loss.
- From Table 4, it can be deduced that the encoding time for neural network approaches is quite less than traditional codecs, however, the decoding time is still high.
- Table 6 signifies that both at low and high bit-rates, the algorithm has performed best in terms of objective and subjective quality metrics.
- Table 7 shows that when assessing with BD-rate, Minnen's [64], Hu's [96] and Lee's [65] approach being based on neural networks have shown better performance as compared to BPG, especially on high-resolution Tecnick dataset.
- Fig. 4 shows the BD rate gain obtained for JointIQ-Net [100] approach against VVC-Intra, BPG and Lee's [65] approaches. It is inferred that compared to training with MSE, the gain obtained is better in the case of training with MS-SSIM loss function on CLIC and Tecnick datasets.

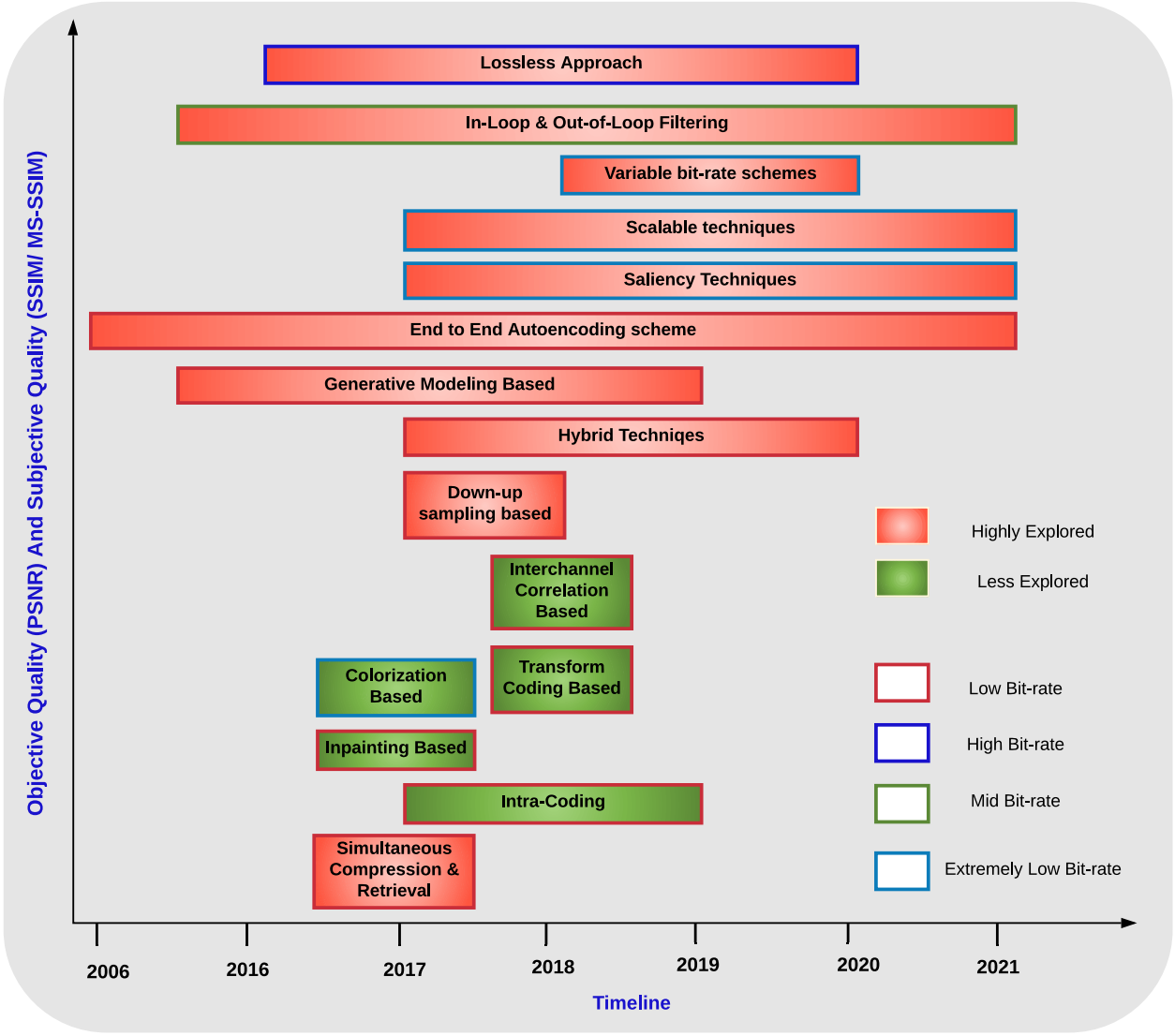


Fig. 12. Roadmap/summary on the basis of survey done on deep learning based compression schemes.

- Table 8 shows the compression performance comparison of Prakash's approach [107] with JPEG and have shown quite better results when assessed with PSNR, PSNR-HVS, PSNR-HVSM, SSIM, MS-SSIM, and VIFP, hereby making it a good saliency approach.
- Fig. 6 signifies the superiority of Minnen's approach [64] over traditional codecs like JPEG, JPEG2000 (OpenJPEG and Kakadu), BPG (4:2:0), BPG (4:4:4) and neural network approaches [77,86,124].
- Fig. 7 a shows bit-saving of Cai's algorithm [103] with respect to the region of interest JPEG2000, where ROI based Cai's has been proved to perform quite better than traditional ROI based JPEG2000 approach, on HKU-IS, ECSSD, LFW, and FDDB datasets respectively. Fig. 10 a shows bit-saving of Cai's algorithm with respect to non-ROI JPEG2000, again the Cai's [103] algorithm is proved to perform quite better than the other two approaches.
- It is observed from Table 14 that among all schemes shown, Alexandre et al. [89], performed best.
- From Table 16 it is concluded that Minnen's [64] algorithm is flexible, robust, and adaptive. Especially, it is superior to other approaches (highlighted in Table) in terms of both objective and subjective evaluation.
- Fig. 10 signifies the BD-rate, BD-PSNR, and BD-MSSSIM evaluation of some benchmark algorithms against traditional codecs JPEG. The algorithm with higher bit-saving percentage namely Chen's [102], Cai's [103], Rippel's [114], Johnston's [146], shows the maximum improvement. However, in terms of BD-PSNR, BPG (4:2:0) and BPG (4:4:4) are still the best approaches. So, there is a scope of improvement for designing an efficient algorithm that can enhance objective performance along with subjective performance. Among lossless approaches, still Tagged Image File Format (TIFF) is the best technique, and neural network approaches need an improvement in the direction of lossless image compression.
- Table 19 shows the computational cost comparison of various benchmark neural network approaches with traditional codecs JPEG, JPEG2000, BPG, and WebP. It is inferred that traditional codecs take a lot of time to encode and decode an image.
- Fig. 11 shows the computational cost of some benchmark algorithms in terms of training parameters. Mentzer's [53], Zhou's [84], Li's [91], Wu's [139], Yang's [149], Liu's [156], Tolunay's [165] and Mentzer's [90] among lossless, end-to-end coding, saliency based coding, scalable compression, variable bit-rate coding, hybrid coding, generative modelling and in-loop/out-of-loop filtering techniques respectively are compu-

tationally expensive. However, Shen's [50], Alexandre's [89], Cai's [103], Li's [137], Kar's [147], Akyazi's [111], Galteri's [166], Yeh's [201] are quite light model techniques under the same categories quoted above.

Based on the comprehensive discussion so far, the algorithms exploiting a perceptual quality metric (like SSIM) or weighted loss metric (like MSE+SSIM & many more) may work better for the requirement of reconstruction of sharp images or HVS synchronized images. Among the various methods reported so far, in our opinion, saliency prediction-based "end-to-end" coding techniques, which lead to the basis of non-uniform or variation allocation of data bits, are most efficient and promising and should be explored more. End-to-end autoencoding schemes are found to be quite flexible than other schemes, which can be easily tuned. The methods using the colorization technique which is not much explored also have a better scope for the improvement in compression efficiency. In fact, this sub-problem of colorizing the gray image, to produce true color exactly same as ground truth is quite interesting but challenging. Intra-coding-based image compression is emerging now, which can be a better option for the researchers to incline towards lossless compression. Exploiting the intra-channel correlation property between the channels of color space is also an attractive direction for reducing storage space & bandwidth for compression. Much effort is still required for inpainting-based compression algorithms. Downsampling and upsampling are used for low bit-rate coding applications. Upsampling methods are very fast in terms of encoding time but provide low compression efficiency. So there is a trade-off between compression speed and compression performance. Post filtering techniques may highly improve the compression efficiency.

4. Challenges

It has been clear that image compression using feature extraction-based deep learning techniques exhibited to give better results and efficiency than traditional image compression methods as since they perform quite better and require no side information as every time the network is learnt according to data given. Lossy image compression approaches discussed so far efficiently meet the data storage issue & bandwidth requirement. But still, there are some challenges on the broader side discussed separately, which are faced by the image compression researchers.

- Technically, the resolution of an object is greatly affected by its orientation & illumination, so the edges are usually the most important means of recognition. Hence a good image compression algorithm should be developed to minimize edge distortion and produce sharp images (sharpness).
- The present perceptual sound metrics do not correctly correlate with the human vision, or not correctly synchronized with HVS. Moreover, they correlate particular types of distortions. The challenge is to coin such a capable metric or loss function synchronized with the human rating that could check all types of distortions on optimization.
- Deep architectures being highly flexible than traditional handcrafted systems have shown quite efficient results in image compression. But again, there is a trade-off between efficiency & computational cost (i.e., the number of training parameters) which need to be optimized. It is the need to design compact network to make it compatible to use in real time scenario.
- More effort is required to find optimal features to be used depending on the task & to optimize rate-distortion efficiently.

5. Conclusion

This paper presents a categorical systematic review of deep learning-based image compression methods. These methods have evolved from several perspectives, including the type of DNN used, the approach used and real-time applicability, etc. We have summarized the milestones, typical methods and highlighted their contributions, strengths, and weaknesses. The unique feature extraction property of CNN is comparatively quite better than any signal processing algorithm as in former the parameters to be used are learned using a huge data and the models in later are dependent on prior knowledge of data. The coding efficiency of the neural network-based approach for far samples can be better utilized by a wide receptive field provided in CNN. On the other hand, traditional codecs can only give better efficiency for neighboring samples, not for far samples. The traditional codecs can only perform better towards the evaluation of HVS, however, DNN can reconstruct spatial, textural, and structural features for HVS along with computer vision analysis. Accordingly, it can be stated that DNNs have increased the performance efficiency of the image compression system which is already discussed. However, the joint rate-distortion optimization problem, which is found in traditional image compression algorithms, has not been solved through DNNs also. Much effort and research are required in this direction also. Although for deep learning methods, training requires massive training data but gathering such data nowadays is not a challenge because most of the data today is multimedia data. The deep learning architecture also requires enormous computing power & the availability of GPU-enabled computers is widespread now. Hence, after a critical discussion and analysis shown, it can be concluded that deep learning-based image compression algorithms are convincing and far better than traditional handcrafted feature-based systems. Based on our evaluation and analysis, critical findings, and challenges discussed, we have tried to build a bridge for the current and new researchers to work in the image compression domain.

Declaration of Competing Interest

None.

References

[1] R.C. Gonzalez, R.E. Woods, Digital Image Processing (3rd Edition), Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006.
[2] S. Dhawan, A review of image compression and comparison of its algorithms, Int. J. Electron. Commun. Technol. IJECT 2 (1) (2011) 22–26.
[3] S. Ma, X. Zhang, C. Jia, Z. Zhao, S. Wang, S. Wanga, Image and video compression with neural networks: a review, IEEE Trans. Circuits Syst. Video Technol. (2019).
[4] G.K. Wallace, The JPEG still picture compression standard, IEEE Trans. Consum. Electron. 38 (1) (1992) xviii–xxxiv.
[5] C. Christopoulos, A. Skodras, T. Ebrahimi, The JPEG2000 still image coding system: an overview, IEEE Trans. Consum. Electron. 46 (4) (2000) 1103–1127.
[6] N. Ahmed, T. Natarajan, K.R. Rao, Discrete cosine transform, IEEE Trans. Comput. 100 (1) (1974) 90–93.
[7] A. Haar, Zur Theorie der orthogonalen Funktionensysteme, Math. Ann. 69 (3) (1910) 331–371.
[8] A. Robinson, C. Cherry, Results of a prototype television bandwidth compression scheme, Proc. IEEE 55 (3) (1967) 356–364.
[9] D.J. MacKay, D.J. Mac Kay, Information Theory, Inference, and Learning Algorithms, Cambridge university press, 2003.
[10] C.C. Cutler, Differential quantization of communication signals, 1952, US Patent 2,605,361.
[11] A. Lewis, G. Knowles, VLSI architecture for 2D Daubechies wavelet transform without multipliers, Electron. Lett. 27 (2) (1991) 171–173.
[12] P.A. Wintz, Transform picture coding, Proc. IEEE 60 (7) (1972) 809–820.
[13] R.M. Gray, D.L. Neuhoff, Quantization, IEEE Trans. Inf. Theory 44 (6) (1998) 2325–2383.
[14] R.M. Gray, Vector quantization, Read. Speech Recognit. 1 (2) (1990) 75–100.
[15] V.K. Goyal, Theoretical foundations of transform coding, IEEE Signal Process. Mag. 18 (5) (2001) 9–21.

[16] A.G. Tescher, R.V. Cox, An adaptive transform coding algorithm, Technical Report, AEROSPACE CORP EL SEGUNDO CA ENGINEERING SCIENCE OPERATIONS, 1976.
[17] N. Jayant, Adaptive quantization with a one-word memory, Bell Syst. Tech. J. 52 (7) (1973) 1119–1144.
[18] S. Lloyd, Least squares quantization in PCM, IEEE Trans. Inf. Theory 28 (2) (1982) 129–137.
[19] J. Max, Quantizing for minimum distortion, IRE Trans. Inf. Theory 6 (1) (1960) 7–12.
[20] K. Pearson, LIII. On lines and planes of closest fit to systems of points in space, Lond. Edinb. Dublin Philos. Mag. J. Sci. 2 (11) (1901) 559–572.
[21] J. MacQueen, et al., Some methods for classification and analysis of multivariate observations, in: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1, Oakland, CA, USA, 1967, pp. 281–297.
[22] J. Ziv, A. Lempel, Compression of individual sequences via variable-rate coding, IEEE Trans. Inf. Theory 24 (5) (1978) 530–536.
[23] J.R. Thompson, Some shrinkage techniques for estimating the mean, J. Am. Stat. Assoc. 63 (321) (1968) 113–122.
[24] D. Salomon, Data Compression: The Complete Reference, Springer Science & Business Media, 2004.
[25] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, et al., Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.
[26] Z. Wang, E.P. Simoncelli, A.C. Bovik, Multiscale structural similarity for image quality assessment, in: Proceedings of the Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003, 2, IEEE, 2003, pp. 1398–1402.
[27] R.C. Streijl, S. Winkler, D.S. Hands, Mean opinion score (MOS) revisited: methods and applications, limitations and alternatives, Multimed. Syst. 22 (2) (2016) 213–227.
[28] M. Everingham, L. Van Gool, C.K. Williams, J. Winn, A. Zisserman, The PASCAL Visual Object Classes (VOC) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.
[29] Y. Yue, T. Finley, F. Radlinski, T. Joachims, A support vector method for optimizing average precision, in: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2007, pp. 271–278.
[30] J. Xie, L. Xu, E. Chen, Image Denoising and inpainting with deep neural networks, in: Proceedings of the Advances in Neural Information Processing Systems, 2012, pp. 341–349.
[31] A. Selimovic, B. Meden, P. Peer, A. Hladnik, Analysis of content-aware image compression with VGG16, in: Proceedings of the IEEE International Work Conference on Bioinspired Intelligence (IWOBI), IEEE, 2018, pp. 1–7.
[32] K. Fukushima, Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position, Biol. Cybern. 36 (4) (1980) 193–202.
[33] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, in: Proceedings of the Advances in Neural Information Processing Systems, 2015, pp. 91–99.
[34] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.
[35] L. Cavigelli, M. Magno, L. Benini, Accelerating real-time embedded scene labeling with convolutional networks, in: Proceedings of the 52nd Annual Design Automation Conference, ACM, 2015, p. 108.
[36] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in: Proceedings of the Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[37] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580–587.
[38] R. Zhao, W. Ouyang, H. Li, X. Wang, Saliency detection by multi-context deep learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1265–1274.
[39] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, T. Brox, FlowNet: learning optical flow with convolutional networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2758–2766.
[40] C. Dong, C.C. Loy, K. He, X. Tang, Learning a deep convolutional network for image super-resolution, in: European Conference on Computer Vision, Springer, 2014, pp. 184–199.
[41] C. Dong, C.C. Loy, K. He, X. Tang, Image super-resolution using deep convolutional networks, IEEE Trans. Pattern Anal. Mach. Intell. 38 (2) (2016) 295–307.
[42] P. Vincent, H. Larochelle, Y. Bengio, P.-A. Manzagol, Extracting and composing robust features with denoising autoencoders, in: Proceedings of the 25th International Conference on Machine Learning, ACM, 2008, pp. 1096–1103.
[43] D.E. Rumelhart, G.E. Hinton, R.J. Williams, et al., Learning representations by back-propagating errors, Cognit. Model. 5 (3) (1986) 1.
[44] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning internal representations by error propagation, Technical Report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.
[45] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in Neural Information Processing Systems, 2014, pp. 2672–2680.
[46] C.K. Parmar, K. Pancholi, A review on image compression techniques, J. Inf. Knowl. Res. Electr. Eng. 2 (2) (2015) 281–284.
[47] M.I. Patel, S. Suthar, J. Thakar, Survey on image compression using machine learning and deep learning, in: Proceedings of the International Conference on Intelligent Computing and Control Systems (ICCS), IEEE, 2019, pp. 1103–1105.
[48] D. Liu, Y. Li, J. Lin, H. Li, F. Wu, Deep learning-based video coding: a review and a case study, ACM Comput. Surv. (CSUR) 53 (1) (2020) 1–35.
[49] Y. Zhang, S. Kwong, S. Wang, Machine learning based video coding optimizations: a survey, Inf. Sci. 506 (2020) 395–423.
[50] H. Shen, W.D. Pan, Y. Dong, M. Alim, Lossless compression of curated erythrocyte images using deep autoencoders for malaria infection diagnosis, in: Proceedings of the Picture Coding Symposium (PCS), IEEE, 2016, pp. 1–5.
[51] G.E. Hinton, T.J. Sejnowski, et al., Learning and relearning in Boltzmann machines, Parallel Distrib. Process. Explor. Microstruct. Cognit. 1 (282-317) (1986) 2.
[52] X. Zhang, X. Wu, Near-lossless L-infinity constrained multi-rate image decompression via deep neural network (2018).
[53] F. Mentzer, E. Agustsson, M. Tschannen, R. Timofte, L. Van Gool, Practical full resolution learned lossless image compression, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
[54] D. Tellez, G. Litjens, J. van der Laak, F. Ciompi, Neural image compression for gigapixel histopathology image analysis, IEEE Trans. Pattern Anal. Mach. Intell. 43 (2) (2021) 567–578, doi:10.1109/TPAMI.2019.2936841.
[55] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad–CAM: visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.
[56] X. Zhang, X. Wu, Ultra high fidelity deep image decompression with l-constrained compression, IEEE Trans. Image Process. 30 (2020) 963–975.
[57] H. Ma, D. Liu, N. Yan, H. Li, F. Wu, End-to-end optimized versatile image compression with wavelet-like transform, IEEE Trans. Pattern Anal. Mach. Intell. (2020), doi:10.1109/TPAMI.2020.3026003.
[58] Z. Cheng, H. Sun, M. Takeuchi, J. Katto, Learned lossless image compression with a hyperprior and discretized gaussian mixture likelihoods, in: Proceedings of the ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020, pp. 2158–2162.
[59] I. Schiopu, A. Munteanu, Deep-learning-based lossless image coding, IEEE Trans. Circuits Syst. Video Technol. 30 (7) (2019) 1829–1842.
[60] X. Wu, N. Memon, Context-based lossless interband compression-extending CALIC, IEEE Trans. Image Process. 9 (6) (2000) 994–1001.
[61] A. Kuznetsova, H. Rom, N. Alldrin, J.R.R. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, T. Duerig, V. Ferrari, The open images dataset V4: unified image classification, object detection, and visual relationship detection at scale, CoRR abs/1811.00982 (2018).
[62] G. Roelofs, Linux gazette: history of the portable network graphics (png) format, Linux J. 1997 (36es) (1997) 19–es.
[63] J. Sneyers, P. Wuille, FLIF: free lossless image format based on MANIAC compression, in: Proceedings of the IEEE International Conference on Image Processing (ICIP), IEEE, 2016, pp. 66–70.
[64] D. Minnen, J. Ballé, G.D. Toderici, Joint autoregressive and hierarchical priors for learned image compression, in: S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett (Eds.), Proceedings of the Advances in Neural Information Processing Systems 31, Curran Associates, Inc., 2018, pp. 10771–10780.
[65] J. Lee, S. Cho, S.-K. Beack, Context-adaptive entropy model for end-to-end optimized image compression, in: Proceedings of the International Conference on Learning Representations, 2019.
[66] W. Zuo, K. Zhang, L. Zhang, Convolutional neural networks for image denoising and restoration, in: Denoising of Photographic Images and Video, Springer, 2018, pp. 93–123.
[67] H.R. Sheikh, Z. Wang, L. Cormack, A.C. Bovik, LIVE image quality assessment database release 2 (2005), 2005.
[68] G.E. Hinton, R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (5786) (2006) 504–507.
[69] T.K. Landauer, P.W. Foltz, D. Laham, An introduction to latent semantic analysis, Discourse Process. 25 (2-3) (1998) 259–284.
[70] Y. Ollivier, Auto-encoders: reconstruction versus compression, CoRR abs/1403.7752 (2014).
[71] A.B.L. Larsen, S.K. Sønderby, H. Larochelle, O. Winther, Autoencoding beyond pixels using a learned similarity metric, in: M.F. Balcan, K.Q. Weinberger (Eds.), Proceedings of The 33rd International Conference on Machine Learning, Proc. of Machine Learning Research, 48, PMLR, New York, New York, USA, 2016, pp. 1558–1566.
[72] G. Toderici, S.M. O'Malley, S.J. Hwang, D. Vincent, D. Minnen, S. Baluja, M. Covell, R. Sukthankar, Variable rate image compression with recurrent neural networks, CoRR abs/1511.06085 (2016).
[73] K. Gregor, F. Besse, D.J. Rezende, I. Danihelka, D. Wierstra, Towards conceptual compression, in: Proceedings of the Advances In Neural Information Processing Systems, 2016, pp. 3549–3557.
[74] K. Gregor, I. Danihelka, A. Graves, D. Wierstra, DRAW: a recurrent neural network for image generation, in: Proceedings of the ICML, 2015.
[75] A. Sento, Image compression with auto-encoder algorithm using Deep Neural Network (DNN), in: Proceedings of the Management and Innovation Technology International Conference (MITicon), IEEE, 2016, pp. MIT–99.
[76] G. Toderici, D. Vincent, N. Johnston, S. Jin Hwang, D. Minnen, J. Shor, M. Covell, Full resolution image compression with recurrent neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5306–5314.

[77] L. Theis, W. Shi, A. Cunningham, F. Huszár, Lossy image compression with compressive autoencoders, in: Proceedings of the International Conference on Learning Representations, 2017.
[78] R.J. Williams, Simple statistical gradient-following algorithms for connectionist reinforcement learning, Mach. Learn. 8 (3-4) (1992) 229–256.
[79] M. Covell, N. Johnston, D. Minnen, S. Jin Hwang, J. Shor, S. Singh, D. Vincent, G. Toderici, Target-quality image compression with recurrent, convolutional neural networks (2017).
[80] E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, L.V. Gool, Soft-to-hard vector quantization for end-to-end learning compressible representations, in: Proceedings of the Advances in Neural Information Processing Systems, 2017, pp. 1141–1151.
[81] T. Dumas, A. Roumy, C. Guillemot, Image compression with Stochastic winner-take-all auto-encoder, in: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2017, pp. 1512–1516.
[82] H. Liu, T. Chen, Q. Shen, T. Yue, Z. Ma, Deep image compression via end–to-end learning, Comput. Vis. Pattern Recognit. (2018).
[83] T. Dumas, A. Roumy, C. Guillemot, Autoencoder based image compression: can the learning be quantization independent? in: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2018, pp. 1188–1192.
[84] L. Zhou, C. Cai, Y. Gao, S. Su, J. Wu, Variational autoencoder for low bit-rate image compression, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2018.
[85] J. Balle, V. Laparra, E.P. Simoncelli, End-to-end optimized image compression, in: Proceedings of the International Conference on Learning Representations, 2017.
[86] J. Ballé, D. Minnen, S. Singh, S.J. Hwang, N. Johnston, Variational image compression with a scale hyperprior, in: Proceedings of the International Conference on Learning Representations, 2018.
[87] R. Torfason, F. Mentzer, E. Ágústsson, M. Tschannen, R. Timofte, L.V. Gool, Towards Image understanding from deep compression without decoding, in: Proceedings of the International Conference on Learning Representations, 2018.
[88] Z. Cheng, H. Sun, M. Takeuchi, J. Katto, Deep convolutional autoencoder-based lossy image compression, in: Proceedings of the Picture Coding Symposium (PCS), IEEE, 2018, pp. 253–257.
[89] D. Alexandre, C.-P. Chang, W.-H. Peng, H.-M. Hang, An autoencoder-based learned image compressor: description of challenge proposal by NCTU, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 2539–2542.
[90] F. Mentzer, E. Agustsson, M. Tschannen, R. Timofte, L. Van Gool, Conditional probability models for deep image compression, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4394–4402.
[91] M. Li, W. Zuo, S. Gu, D. Zhao, D. Zhang, Learning convolutional networks for content-weighted image compression, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3214–3223.
[92] Z. Chen, T. He, Learning based facial image compression with semantic fidelity metric, Neurocomputing 338 (2019) 16–25.
[93] S. Ayzik, S. Avidan, Deep image compression using decoder side information, in: Proceedings of the 16th European Conference on Computer Vision–ECCV 2020, Glasgow, UK, August 23–28, 2020, Proc., Part XVII 16, Springer, 2020, pp. 699–714.
[94] D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: Proceedings of the 8th International Conference on Computer Vision, 2, 2001, pp. 416–423.
[95] A. Geiger, P. Lenz, C. Stiller, R. Urtasun, The KITTI vision benchmark suite, https://www.cvlibs.net/datasets/kitti(2015).
[96] Y. Hu, W. Yang, Z. Ma, J. Liu, Learning end-to-end lossy image compression: a benchmark, IEEE Trans. Pattern Anal. Mach. Intell. (2021), doi:10.1109/TPAMI.2021.3065339.
[97] B. Kang, S. Tripathi, T. Nguyen, Toward joint image generation and compression using generative adversarial networks (2019).
[98] S.K. Raman, A. Ramesh, V. Naganoor, S. Dash, G. Kumaravelu, H. Lee, CompressNet: generative compression at extremely low bitrates, in: Proceedings of the IEEE Winter Conf. on Applications of Computer Vision, 2020, pp. 2325–2333.
[99] Z. Cheng, H. Sun, M. Takeuchi, 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.
[100] J. Lee, S. Cho, M. Kim, An end-to-end joint learning scheme of image compression and quality enhancement with improved entropy minimization, arXiv preprint arXiv:1912.12817(2020).
[101] A. Punnappurath, M.S. Brown, Learning raw image reconstruction-aware deep image compressors, IEEE Trans. Pattern Anal. Mach. Intell. 42 (4) (2019) 1013–1019.
[102] T. Chen, H. Liu, Z. Ma, Q. Shen, X. Cao, Y. Wang, End-to-end learnt image compression via non-local attention optimization and improved context modeling, IEEE Trans. Image Process. Publ. IEEE Signal Process. Soc. 30 (2021) 3179–3191, doi:10.1109/TIP.2021.3058615.
[103] C. Cai, L. Chen, X. Zhang, Z. Gao, End-to-end optimized ROI image compression, IEEE Trans. Image Process. 29 (2020) 3442–3457.
[104] J. Cai, Z. Cao, L. Zhang, Learning a single tucker decomposition network for lossy image compression with multiple bits-per-pixel rates, IEEE Trans. Image Process. 29 (2020) 3612–3625.
[105] H. Sun, Z. Cheng, M. Takeuchi, J. Katto, End-to-end learned image compression with fixed point weight quantization, in: Proceedings of the IEEE International Conference on Image Processing (ICIP), IEEE, 2020, pp. 3359–3363.
[106] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: a large-scale hierarchical image database, in: Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 248–255.
[107] A. Prakash, N. Moran, S. Garber, A. DiLillo, J. Storer, Semantic perceptual image compression using deep convolution networks, in: Proceedings of the Data Compression Conference (DCC), IEEE, 2017, pp. 250–259.
[108] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2921–2929.
[109] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, Gans trained by a two time-scale update rule converge to a local nash equilibrium, Adv. Neural Inf. Process. Syst. 30 (2017).
[110] E. Agustsson, M. Tschannen, F. Mentzer, R. Timofte, L. van Gool, Generative adversarial networks for extreme learned image compression, in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 221–231.
[111] P. Akyazi, T. Ebrahimi, Learning-based image compression using convolutional autoencoder and wavelet decomposition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2019. 0–0
[112] G. Griffin, A. Holub, P. Perona, Caltech-256 object category dataset(2007).
[113] S. Luo, Y. Yang, Y. Yin, C. Shen, Y. Zhao, M. Song, DeepSIC: deep semantic image compression, in: Proceedings of the International Conference on Neural Information Processing, Springer, 2018, pp. 96–106.
[114] O. Rippel, L. Bourdev, Real-time adaptive image compression, in: Proceedings of the 34th International Conference on Machine Learning-Volume 70, JMLR. org, 2017, pp. 2922–2930.
[115] M. Akbari, J. Liang, J. Han, DSSLIC: deep semantic segmentation-based layered image compression, in: Proceedings of the ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019, pp. 2042–2046.
[116] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3213–3223.
[117] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, A. Torralba, Scene parsing through ADE20K dataset, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.
[118] C. Wang, Y. Han, W. Wang, An end-to-end deep learning image compression framework based on semantic analysis, Appl. Sci. 9 (17) (2019) 3580.
[119] M. Li, W. Zuo, S. Gu, J. You, D. Zhang, Learning content-weighted deep image compression, IEEE Trans. Pattern Anal. Mach. Intell. (2020) 1-1.
[120] H. Akutsu, T. Naruko, End-to-end learned RoI image compression, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2019.
[121] M. Li, K. Ma, J. You, D. Zhang, W. Zuo, Efficient and effective context-based convolutional entropy modeling for image compression, IEEE Trans. Image Process. 29 (2020) 5900–5911.
[122] M. Li, K. Zhang, W. Zuo, R. Timofte, D. Zhang, Learning context-based non-local entropy modeling for image compression, arXiv preprint arXiv:2005.04661(2020b).
[123] D. Marpe, H. Schwarz, T. Wiegand, Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard, IEEE Trans. Circuits Syst. Video Technol. 13 (7) (2003) 620–636.
[124] D. Minnen, G. Toderici, S. Singh, S.J. Hwang, M. Covell, Image-dependent local entropy models for learned image compression, in: Proceedings of the 25th IEEE International Conference on Image Processing (ICIP), IEEE, 2018, pp. 430–434.
[125] Y. Xue, J. Su, Attention based image compression post-processing convolutional neural network, in: Proceedings of the CVPR Workshops, 2019, p. 0.
[126] L. Zhou, Z. Sun, X. Wu, J. Wu, End-to-end optimized image compression with attention mechanism, in: Proceedings of the CVPR workshops, 2019, p. 0.
[127] J.C.M.S.A. Djelouah, C. Schroers, Content adaptive optimization for neural image compression, in: Proceedings of the CVPR, 2019.
[128] Z. Cheng, P. Akyazi, H. Sun, J. Katto, T. Ebrahimi, Perceptual quality study on deep learning based image compression, in: Proceedings of the IEEE International Conference on Image Processing (ICIP), IEEE, 2019, pp. 719–723.
[129] G. Li, Y. Yu, Visual saliency based on multiscale deep features, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 5455–5463.
[130] J. Shi, Q. Yan, L. Xu, J. Jia, Hierarchical image saliency detection on extended CSSD, IEEE Trans. Pattern Anal. Mach. Intell. 38 (4) (2015) 717–729.
[131] G.B. Huang, M. Ramesh, T. Berg, E. Learned-Miller, Labeled faces in the wild: a database for studying face recognition in unconstrained environments, Technical Report, 07-49, University of Massachusetts, Amherst, 2007.
[132] V. Jain, E. Learned-Miller, FDDB: a benchmark for face detection in unconstrained settings, Technical Report, UM-CS-2010-009, University of Massachusetts, Amherst, 2010.
[133] L.-H. Chen, C.G. Bampis, Z. Li, A. Norkin, A.C. Bovik, ProxIQA: a proxy approach to perceptual optimization of learned image compression, IEEE Trans. Image Process. 30 (2020) 360–373.

[134] D. Minnen, G. Toderici, M. Covell, T. Chinen, N. Johnston, J. Shor, S.J. Hwang, D. Vincent, S. Singh, Spatially adaptive image compression using a tiled deep network, in: Proceedings of the IEEE International Conference on Image Processing (ICIP), IEEE, 2017, pp. 2796–2800.
[135] C. Cai, L. Chen, X. Zhang, Z. Gao, Efficient variable rate image compression with multi-scale decomposition network, IEEE Trans. Circuits Syst. Video Technol. 29 (12) (2018) 3687–3700, doi:10.1109/TCSVT.2018.2880492.
[136] A. Karkada Ashok, N. Palani, Autoencoders with variable sized latent vector for image compression, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2018.
[137] M. Li, J. Hu, C. Xia, Y. Zhang, An implementation of picture compression with a CNN-based auto-encoder, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2018.
[138] J. Zhou, S. Wen, A. Nakagawa, K. Kazui, Z. Tan, Multi-scale and context-adaptive entropy model for image compression, arXiv preprint arXiv:1910.07844(2019).
[139] L. Wu, K. Huang, H. Shen, A GAN-based tunable image compression system, " in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2020, pp. 2334–2342.
[140] T. Chen, H. Liu, Z. Ma, Q. Shen, X. Cao, Y. Wang, End-to-end learnt image compression via non-local attention optimization and improved context modeling, IEEE Trans. Image Proc. (2021), doi:10.1109/TIP.2021.3058615. 1–1
[141] N. Yan, D. Liu, H. Li, F. Wu, Semantically scalable image coding with compression of feature maps, in: Proceedings of the IEEE International Conference on Image Processing (ICIP), IEEE, 2020, pp. 3114–3118.
[142] N. Asuni, A. Giachetti, TESTIMAGES: a large-scale archive for testing visual devices and basic image processing algorithms, in: A. Giachetti (Ed.), Proceedings of the Smart Tools and Apps for Graphics - Eurographics Italian Chapter Conference, The Eurographics Association, 2014, doi:10.2312/stag.20141242.
[143] Z. Tang, L. Luo, Compression artifact removal using multi-scale reshuffling convolutional network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2018.
[144] A.G. Ororbia, A. Mali, J. Wu, S. O'Connell, W. Dreese, D. Miller, C.L. Giles, Learned neural iterative decoding for lossy image compression systems, in: Proceedings of the Data Compression Conference (DCC), IEEE, 2019, pp. 3–12.
[145] H. Chen, X. He, L. Qing, S. Xiong, T.Q. Nguyen, DPW-SDNet: dual pixel-wavelet domain deep CNNs for soft decoding of JPEG-compressed images, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 711–720.
[146] N. Johnston, D. Vincent, D. Minnen, M. Covell, S. Singh, T. Chinen, S. Jin Hwang, J. Shor, G. Toderici, Improved lossy image compression with priming and spatially adaptive bit rates for recurrent networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4385–4393.
[147] A. Kar, S. Phani Krishna Karri, N. Ghosh, R. Sethuraman, D. Sheet, Fully convolutional model for variable bit length and lossy high density compression of mammograms, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2018.
[148] Y. Choi, M. El-Khamy, J. Lee, Variable rate deep image compression with a conditional autoencoder, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2019.
[149] F. Yang, L. Herranz, J. van de Weijer, J.A.I. Guitián, A.M. López, M.G. Mozerov, Variable rate deep image compression with modulated autoencoder, IEEE Signal Process. Lett. 27 (2020) 331–335.
[150] S. Sun, T. He, Z. Chen, Semantic structured image coding framework for multiple intelligent applications, IEEE Trans. Circuits Syst. Video Technol. (2020).
[151] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder–decoder for statistical machine translation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Doha, Qatar, 2014, pp. 1724–1734, doi:10.3115/v1/D14-1179.
[152] F. Jiang, W. Tao, S. Liu, J. Ren, X. Guo, D. Zhao, An end-to-end compression framework based on convolutional neural networks, IEEE Trans. Circuits Syst. Video Technol. 28 (10) (2017) 3007–3018.
[153] L. Zhao, H. Bai, A. Wang, Y. Zhao, Learning a virtual codec based on deep convolutional neural network to compress image, J. Vis. Commun. Image Represent. 63 (2019) 102589, doi:10.1016/j.jvcir.2019.102589.
[154] J. Hu, M. Li, C. Xia, Y. Zhang, Combine traditional compression method with convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2018.
[155] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[156] Z. Liu, T. Liu, W. Wen, L. Jiang, J. Xu, Y. Wang, G. Quan, DeepN-JPEG: a deep neural network favorable JPEG-based image compression framework, in: Proceedings of the 55th Annual Design Automation Conference, ACM, 2018, p. 18.
[157] R. Zeyde, M. Elad, M. Protter, On single image scale-up using sparse-representations, in: Proceedings of the International Conference on Curves and Surfaces, Springer, 2010, pp. 711–730.
[158] D. Mishra, S.K. Singh, R.K. Singh, Wavelet-based deep auto encoder-decoder (wdaed)-based image compression, IEEE Trans. Circuits Syst. Video Technol. 31 (4) (2021) 1452–1462, doi:10.1109/TCSVT.2020.3010627.
[159] J. Snell, K. Ridgeway, R. Liao, B.D. Roads, M.C. Mozer, R.S. Zemel, Learning to generate images with perceptual similarity metrics, in: Proceedings of the IEEE International Conference on Image Processing (ICIP), IEEE, 2017, pp. 4277–4281.
[160] A. Dosovitskiy, T. Brox, Generating images with perceptual similarity metrics based on deep networks, in: Proceedings of the Advances in Neural Information Processing Systems, 2016, pp. 658–666.
[161] D.J. Im, C.D. Kim, H. Jiang, R. Memisevic, Generating images with recurrent adversarial networks, CoRR abs/1602.05110 (2016).
[162] L. Gatys, A. Ecker, M. Bethge, A neural algorithm of artistic style, arXiv (2015). 10.1167/16.12.326.
[163] E.L. Denton, S. Chintala, R. Fergus, et al., Deep generative image models using a Laplacian pyramid of adversarial networks, in: Proceedings of the Advances in Neural Information Processing Systems, 2015, pp. 1486–1494.
[164] D.J. Im, C.D. Kim, H. Jiang, R. Memisevic, Generative adversarial metric(2016).
[165] E.M. Tolunay, A. Ghalayini, Generative neural network based image compression(2018).
[166] L. Galteri, L. Seidenari, M. Bertini, A. Del Bimbo, Deep generative adversarial compression artifact removal, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 4826–4835.
[167] T. Nguyen, D. Marpe, Objective performance evaluation of the HEVC main still picture profile, IEEE Trans. Circuits Syst. Video Technol. 25 (5) (2014) 790–797.
[168] Y. Dai, D. Liu, F. Wu, A convolutional neural network approach for post-processing in HEVC intra coding, in: Proceedings of the International Conference on Multimedia Modeling, Springer, 2017, pp. 28–39.
[169] J. Li, B. Li, J. Xu, R. Xiong, W. Gao, Fully connected network-based intra prediction for image coding, IEEE Trans. Image Process. 27 (7) (2018) 3236–3247.
[170] Y. Li, D. Liu, H. Li, L. Li, F. Wu, H. Zhang, H. Yang, Convolutional neural network-based block up-sampling for intra frame coding, IEEE Trans. Circuits Syst. Video Technol. 28 (9) (2018) 2316–2330.
[171] Y. Hu, W. Yang, M. Li, J. Liu, Progressive spatial recurrent neural network for intra prediction, IEEE Trans. Multimed. 21 (12) (2019) 3024–3037.
[172] L. Zhu, S. Kwong, Y. Zhang, S. Wang, X. Wang, Generative adversarial network-based intra prediction for video coding, IEEE Trans. Multimed. 22 (1) (2019) 45–58.
[173] T. Dumas, A. Roumy, C. Guillemot, Context-adaptive neural network-based prediction for image compression, IEEE Trans. Image Process. 29 (2019) 679–693.
[174] H. Sun, Z. Cheng, M. Takeuchi, J. Katto, Enhanced intra prediction for video coding by using multiple neural networks, IEEE Trans. Multimed. 22 (11) (2020) 2764–2779.
[175] H. Sun, L. Yu, J. Katto, Fully neural network mode based intra prediction of variable block size, in: Proceedings of the IEEE International Conference on Visual Communications and Image Processing (VCIP), IEEE, 2020, pp. 21–24.
[176] M.H. Baig, V. Koltun, L. Torresani, Learning to inpaint for image compression, in: Proceedings of the Advances in Neural Information Processing Systems, 2017, pp. 1246–1255.
[177] D. Varga, T. Szirányi, Fully automatic image colorization based on convolutional neural network, in: Proceedings of the 23rd International Conference on Pattern Recognition (ICPR), IEEE, 2016, pp. 3691–3696.
[178] M.H. Baig, L. Torresani, Multiple hypothesis colorization and its application to image compression, Comput. Vis. Image Underst. 164 (2017) 111–123.
[179] M.H. Baig, L. Torresani, Colorization for image compression, CoRR abs/1606.06314 (2016).
[180] L. Cheng, S. Vishwanathan, Learning to compress images and videos, in: Proceedings of the 24th International Conference on Machine Learning, ACM, 2007, pp. 161–168.
[181] X. He, M. Ji, H. Bao, A unified active and semi-supervised learning framework for image compression, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 65–72.
[182] K. Cui, E. Steinbach, Decoder side image quality enhancement exploiting inter-channel correlation in a 3-stage CNN: submission to CLIC 2018, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 2571–2574.
[183] Q. Zhang, D. Liu, H. Li, Deep network-based image coding for simultaneous compression and retrieval, in: Proceedings of the IEEE International Conference on Image Processing (ICIP), IEEE, 2017, pp. 405–409.
[184] J. Eakins, M. Graham, Content-based image retrieval(1999).
[185] D. Liu, H. Ma, Z. Xiong, F. Wu, CNN-based DCT-like transform for image compression, in: Proceedings of the International Conference on Multimedia Modeling, Springer, 2018, pp. 61–72.
[186] G. Schaefer, M. Stich, UCID: an uncompressed color image database, in: Proceedings of the Storage and Retrieval Methods and Applications for Multimedia 2004, 5307, International Society for Optics and Photonics, 2003, pp. 472–480.
[187] Y. Tan, J. Cai, S. Zhang, W. Zhong, L. Ye, Image compression algorithms based on super-resolution reconstruction technology, in: Proceedings of the IEEE 4th International Conference on Image, Vision and Computing (ICIVC), IEEE, 2019, pp. 162–166.
[188] C. Dong, Y. Deng, C. Change Loy, X. Tang, Compression artifacts reduction by a deep convolutional network, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 576–584.
[189] A. Foi, V. Katkovnik, K. Egiazarian, Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images, IEEE Trans. Image Process. 16 (5) (2007) 1395–1411.
[190] L. Cavigelli, P. Hager, L. Benini, CAS-CNN: a deep convolutional neural network for image compression artifact suppression, in: Proceedings of the International Joint Conference on Neural Networks (IJCNN), IEEE, 2017, pp. 752–759.

[191] P. Svoboda, M. Hradiš, D. Bařina, P. Zemčík, Compression artifacts removal using convolutional neural networks, J. WSCG 24 (2) (2016) 63–72.
[192] C. Aytekin, X. Ni, F. Cricri, J. Lainema, E. Aksu, M. Hannuksela, Block-optimized variable bit rate neural image compression, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2018.
[193] O. Ronneberger, P. Fischer, T. Brox, U-Net: convolutional networks for biomedical image segmentation, in: Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2015, pp. 234–241.
[194] D. Maleki, S. Nadalian, M.M. Derakhshani, M.A. Sadeghi, BlockCNN: a deep network for artifact removal and image compression, in: Proceedings of the CVPR Workshops, 2018, pp. 2555–2558.
[195] M. Everingham, S.M.A. Eslami, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The pascal visual object classes challenge: a retrospective, Int. J. Comput. Vis. 111 (1) (2015) 98–136.
[196] O. Kirmemis, G. Bakar, A. Murat Tekalp, Learned compression artifact removal by deep residual networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2018.
[197] B. Lim, S. Son, H. Kim, S. Nah, K. Mu Lee, Enhanced deep residual networks for single image super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017, pp. 136–144.
[198] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al., Photo-realistic single image super-resolution using a generative adversarial network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4681–4690.
[199] Z. Jin, M.Z. Iqbal, D. Bobkov, W. Zou, X. Li, E. Steinbach, A flexible deep CNN framework for image restoration, IEEE Trans. Multimed. 22 (4) (2020) 1055–1068.
[200] J. Liu, D. Liu, W. Yang, S. Xia, X. Zhang, Y. Dai, A comprehensive benchmark for single image compression artifact reduction, IEEE Trans. Image Process. 29 (2020) 7845–7860.
[201] C.-H. Yeh, C.-H. Lin, M.-H. Lin, L.-W. Kang, C.-H. Huang, M.-J. Chen, Deep learning-based compressed image artifacts reduction based on multi-scale image fusion, Inf. Fusion 67 (2021) 195–207.
[202] A. Van Oord, N. Kalchbrenner, K. Kavukcuoglu, Pixel recurrent neural networks, in: Proceedings of the International Conference on Machine Learning, PMLR, 2016, pp. 1747–1756.
[203] R. Dahl, M. Norouzi, J. Shlens, Pixel recursive super resolution, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 5439–5448.
[204] S. Santurkar, D. Budden, N. Shavit, Generative compression, in: Proceedings of the Picture Coding Symposium (PCS), IEEE, 2018, pp. 258–262.
[205] K. Yu, C. Dong, C.C. Loy, X. Tang, Deep convolution networks for compression artifacts reduction, arXiv preprint arXiv:1608.02778(2016).
[206] P. Svoboda, M. Hradis, D. Barina, P. Zemcik, Compression artifacts removal using convolutional neural networks, arXiv preprint arXiv:1605.00366(2016).
[207] V. Jain, S. Seung, Natural image denoising with convolutional networks, in: Proceedings of the Advances in Neural Information Processing Systems, 2009, pp. 769–776.
[208] K. Zhang, W. Zuo, Y. Chen, D. Meng, L. Zhang, Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising, IEEE Trans. Image Process. 26 (7) (2017) 3142–3155.
[209] X. Mao, C. Shen, Y.B. Yang, Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections, in: Proceedings of the Advances in Neural Information Processing Systems, 2016, pp. 2802–2810.
[210] M. Gonzalez, J. Preciozzi, P. Muse, A. Almansa, Joint denoising and decompression using CNN regularization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2018.
[211] H. Ma, D. Liu, R. Xiong, F. Wu, A CNN-based image compression scheme compatible with JPEG-2000, in: Proceedings of the IEEE International Conference on Image Processing (ICIP), IEEE, 2019, pp. 704–708.
[212] Y. Kim, J.W. Soh, J. Park, B. Ahn, H.-S. Lee, Y.S. Moon, N.I. Cho, A pseudo-blind convolutional neural network for the reduction of compression artifacts, IEEE Trans. Circuits Syst. Video Technol. 30 (4) (2019) 1121–1135.
[213] J. Li, Y. Wang, H. Xie, K.-K. Ma, Learning a single model with a wide range of quality factors for JPEG image artifacts removal, IEEE Trans. Image Process. 29 (2020) 8842–8854.
[214] D. Mishra, S. K. Singh, R. K. Singh, Lossy Medical Image Compression using Residual Learning-based Dual Autoencoder Model, 2020 IEEE 7th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON) (2020) 1–5, doi:10.1109/UPCON50219.2020.9376417.
[215] Y. Kim, J. W. Soh, N. I. Cho, AGARNet: Adaptively Gated JPEG Compression Artifacts Removal Network for a Wide Range Quality Factor, IEEE Access 8 (2020) 20160–20170.
[216] D. Mishra, S. K. Singh, R. K. Singh, K. Preetham, Edge-Aware Image Compression using Deep Learning-based Super-resolution Network, arXiv preprint arXiv:2104.04926 (2021).
[217] H. Son, T. Kim, H. Lee, S. Lee, Enhanced Standard Compatible Image Compression Framework based on Auxiliary Codec Networks, arXiv preprint arXiv:2009.14754, (2020).
[218] D. Mishra, S. K. Singh, R. K. Singh, D. Kedia, Multi-scale network (MsSG-CNN) for joint image and saliency map learning-based compression, Neurocomputing 460 (2021) 95–105, doi:10.1016/j.neucom.2021.07.012.

Dipti Mishra, (Graduate Student Member IEEE) is currently pursuing Ph.D. from Indian Institute of Information Technology Allahabad, India where she is associated with Computer Vision and Biometrics Lab (CVBL), IIIT Allahabad, India. She received her B.Tech and M.Tech in Electronics & Communication Engineering from Dr. APJ Abdul Kalam Technical University, Lucknow, India in 2010 and Jaypee Institute of Information Technology, Noida, India in 2015 respectively. She has over 5 years of experience in academic and research organization. Her current research interests lies in image processing, image compression, signal processing, pattern recognition, machine learning and deep learning.

Dr. Satish Kumar Singh is an Associate Professor with the Indian Institute of Information Technology Allahabad, Prayagraj, India. He has over 16 years of experience in academic and research. He has authored over 70 publications in reputed International Journals and Conference proceedings. He is a fellow Institution of Engineers, India (IEI), fellow Institution of Electronics and Telecommunication Engineers (IETE), Senior member Institute of Electrical and Electronics Engineers (IEEE) and member International Association of Pattern Recognition Research (IAPR). Currently, he is serving as the Chair of IEEE Uttar Pradesh Section and Secretary, Signal Processing Society Chapter, IEEE Uttar Pradesh Section as well. He is also the Associate Editor, IET Image Processing, IJPRAI and lead guest editor in Neural Computing and Applications, Special Issue on Computer Vision, Image Processing, and Guest Editor of Springer Nature Computer Science Special issues on recent trends in computer vision and progress in image processing. His current research interests include machine learning and deep learning, digital image processing, image compression, signal processing,

Dr. Rajat Kumar Singh (Senior Member IEEE) received B.Tech. degree in Electronics and Instrumentation Engineering from BIET, Jhansi, India, in 1999, M.Tech. degree in Communication Engineering from BITS, Pilani, India, in 2001, and Ph.D. degree from Indian Institute of Technology Kanpur (IITK), India, in 2007, with a focus on the architecture of optical packet switching incorporating various buffering techniques. He is currently working as an Associate Professor with the Department of Electronics and Communication Engineering at the Indian Institute of Information Technology, Allahabad, India. His current research interests are in the areas of Optical Networking and Switching, Wireless Sensor Network, and Image Processing. He has published various research articles in different Journals/Conferences of repute like IEEE, Springer, Elsevier, etc.