Visual Intelligence

REVIEW Open Access

Overview of intelligent video coding: from model-based to learning-based approaches

Siwei Ma[1*], Junlong Gao[1], Ruofan Wang[1], Jianhui Chang[1], Qi Mao[2], Zhimeng Huang[1] and Chuanmin Jia[3]

Abstract

Intelligent video coding (IVC), which dates back to the late 1980s with the concept of encoding videos with knowledge and semantics, includes visual content compact representation models and methods enabling structural, detailed descriptions of visual information at different granularity levels (i.e., block, mesh, region, and object) and in different areas. It aims to support and facilitate a wide range of applications, such as visual media coding, content broadcasting, and ubiquitous multimedia computing. We present a high-level overview of the IVC technology from model-based coding (MBC) to learning-based coding (LBC). MBC mainly adopts a manually designed coding scheme to explicitly decompose videos to be coded into blocks or semantic components. Thanks to emerging deep learning technologies such as neural networks and generative models, LBC has become a rising topic in the coding area. In this paper, we first review the classical MBC approaches, followed by the LBC approaches for image and video data. We also discuss and overview our recent attempts at neural coding approaches, which are inspiring for both academic research and industrial implementation. Some critical yet less studied issues are discussed at the end of this paper.

1 Introduction

Digital image/video coding has boomed with the digitalization of information since the late 1950s, as the data size of the original digitalized image or video data increases dramatically and reaches beyond the capability of storage and transmission. During the early stages of image coding, removing spatial statistical redundancy was the main means of image compression, such as Huffman coding [1] and Run-length coding [2]. The concept of transform coding, which transforms the spatial domain into the frequency domain for compression, was first proposed in the late 1960s, including the Fourier transform [3] and Hadamard transform [4]. Later the discrete cosine transform (DCT) was designed for image coding in 1974 by Ahmed et al. [5]. In the case of video, there is significant temporal redundancy in addition to spatial redundancy, which can be reduced by applying temporal prediction. Several early prediction-based coding techniques were introduced during the 1970s, including differential pulse-code modulation (DPCM) [6], frame difference coding [7], and block-based motion prediction [8]. A prototype of a hybrid prediction/transform coding scheme [9] was first proposed in 1979 by Netravali and Stuller, who combined motion compensation with transform coding techniques, commonly referred to as "the first generation" coding scheme. An overview of the historical development of the first-generation methods is provided in [10].

After several decades of development, hybrid prediction/transform coding methods have achieved great success. Various coding standards have been developed and are widely used in a variety of applications, such as MPEG-1/2/4 (Moving Picture Experts Group), H.261/2/3, and H.264/AVC (Advanced Video Coding) [11], as well as AVS (Audio and Video Coding Standard in China) [12–15], H.265/HEVC (High Efficiency Video Coding) [16], and H.266/VVC (Versatile Video Coding) [17]. In [11, 16–26],

[*]Correspondence: swma@pku.edu.cn
[1]National Engineering Research Center of Visual Technology, School of Computer Science, Peking University, Beijing, 100871, China
Full list of author information is available at the end of the article

Springer

the traditional hybrid coding methods have been well reviewed from the historical pulse code modulation (PCM), DPCM coding to HEVC, three-dimensional video (3DV) coding, and VVC.

With the huge number of mobile devices, surveillance cameras, and other video capture devices, the volume of video data is increasing significantly. In the coming era of big data, image and video processing will require more efficient and effective coding techniques. Nevertheless, researchers in this field have also acknowledged the difficulty of further improving performance under the traditional hybrid coding framework. One reason for the performance improvement limitation is that the traditional coding methods only consider the signal properties of images and videos and the room left for improvement is increasingly squeezed with the constraint of objective quality measurement, e.g. peak signal noise ratio (PSNR). As such, many novel coding methods that incorporate the properties of the human visual system (HVS), referred to as the second-generation coding methods [27–30], have demonstrated a higher compression ratio over traditional coding methods while maintaining comparable subjective image quality. Compared to the first-generation coding methods, these methods are more dependent on the structural object-related model than on the source signal. From Musmann's viewpoint [31], model-based coding (MBC) is composed of the first-generation and second-generation methods, which are based on a signal source or structural object-related models. MBC arose and attracted the interest of researchers, research has advanced greatly in this field, and some exciting results have been achieved. For example, in [32], a background picture model-based surveillance video coding method shows at least twice the compression ratio on surveillance videos of the AVC high profile. Moreover, other model-based coding methods display great potential nowadays and achieve obvious improvement over the traditional hybrid coding methods, such as geometric partition video coding [33] and segmentation-based coding [34]. Some MBC methods were also introduced into various coding standards, such as MPEG-4/7, AVS2, HEVC, and VVC. The developments in MBC have been well reviewed in [35–42].

Although MBC aims to improve coding efficiency, many challenging problems still limit the effectiveness of the coding process, such as manually designed coding paradigms based on expert knowledge. During the last few years, neural networks, such as convolutional neural networks (CNNs), have demonstrated considerable potential in a variety of fields, including image and video understanding, processing, and compression. In terms of the compression task, neural networks perform transform coding by mapping pixel data into quantized latent representations first and then converting them back again into pixels. Such a nonlinear transform holds the potential to map pixels to a more compact latent representation than the transforms of the preceding codecs. Moreover, the parameters in neural networks can be well trained based on massive image and video samples, which facilitates the model to alleviate its reliance on manually designed modules. Considering these excellent characteristics, learning-based coding (LBC) has been recognized as a promising solution for image and video coding.

In this paper, we will present an overview of intelligent video coding (IVC) development from MBC to LBC, in which the two technologies encode videos leveraging knowledge in different manners. The technical roadmap of IVC methods is summarized in Fig. 1. The similarity between MBC and LBC is that similar components, such as transform, quantization, and entropy coding, are adopted to construct the framework to exploit the correlation of textural content and remove redundancy. The difference lies in that the former relies on manually designed modules, while the latter relies on a data-driven strategy or components using machine learning. The rest of the paper is organized as follows. In Sect. 2, a brief introduction to the history of MBC is provided. Section 3 provides an overview of recent advancements in learning-based approaches for visual signal compression, including learned image compression and learning-based video coding. Section 4 introduces our previous attempts and understanding of IVC. In Sect. 5, we discuss the future directions of IVC, specifically from the perspectives of standardized potentials, data security, and generalization. Section 6 concludes this paper.

2 Model-based coding

MBC focuses on modeling and coding the structural visual information in the images and videos. The history of MBC can be traced back to the 1950s [43]. In [43], Schreiber et al. proposed a Synthetic Highs coding scheme, where the image content is divided into textures and edges, and they are coded by different approaches, e.g. using statistical coding methods for textures and visual model-based coding methods for edges, which was the predecessor of the current HVS model-based perceptual coding methods. In [38], Pearson clarified the term "model" in MBC, which is explained as object-related models and developed from the source model in signal processing, as shown in Fig. 2. A video sequence containing one or more moving objects is analyzed to yield information about the size, location, and motion of the objects, which is employed to synthesize a model of each object as animation data. The animation data are coded and transmitted to the decoder. Moreover, the residual pixel data, comprising the difference between the original video sequence and the sequence derived from the animated model, are also transmitted to the decoder. The decoder adopts the animation data to synthesize the model, which is subsequently accompanied by the residual

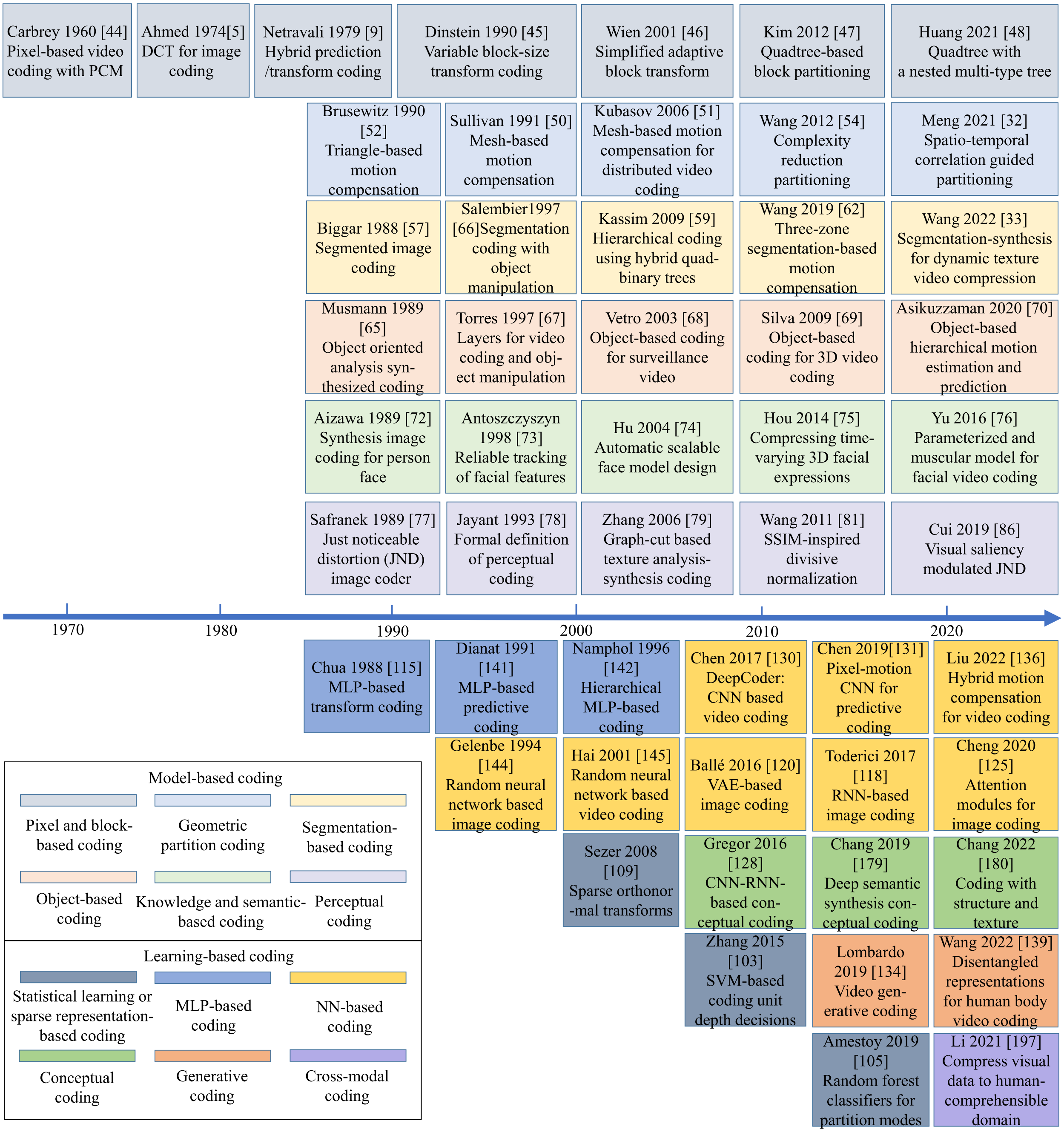

Carbrey 1960 [44] Pixel-based video coding with PCM
Ahmed 1974[5] DCT for image coding
Netravali 1979 [9] Hybrid prediction /transform coding
Dinstein 1990 [45] Variable block-size transform coding
Wien 2001 [46] Simplified adaptive block transform
Kim 2012 [47] Quadtree-based block partitioning
Huang 2021 [48] Quadtree with a nested multi-type tree
Brusewitz 1990 [52] Triangle-based motion compensation
Sullivan 1991 [50] Mesh-based motion compensation
Kubasov 2006 [51] Mesh-based motion compensation for distributed video coding
Wang 2012 [54] Complexity reduction partitioning
Meng 2021 [32] Spatio-temporal correlation guided partitioning
Biggar 1988 [57] Segmented image coding
Salembier1997 [66]Segmentation coding with object manipulation
Kassim 2009 [59] Hierarchical coding using hybrid quad-binary trees
Wang 2019 [62] Three-zone segmentation-based motion compensation
Wang 2022 [33] Segmentation-synthesis for dynamic texture video compression
Musmann 1989 [65] Object oriented analysis syn-thesized coding
Torres 1997 [67] Layers for video coding and obj-ect manipulation
Vetro 2003 [68] Object-based coding for surveillance video
Silva 2009 [69] Object-based coding for 3D video coding
Asikuzzaman 2020 [70] Object-based hierarchical motion estimation and prediction
Aizawa 1989 [72] Synthesis image coding for person face
Antoszczyszyn 1998 [73] Reliable tracking of facial features
Hu 2004 [74] Automatic scalable face model design
Hou 2014 [75] Compressing time-varying 3D facial expressions
Yu 2016 [76] Parameterized and muscular model for facial video coding
Safranek 1989 [77] Just noticeable distortion (JND) image coder
Jayant 1993 [78] Formal definition of perceptual coding
Zhang 2006 [79] Graph-cut based texture analysis-synthesis coding
Wang 2011 [81] SSIM-inspired divisive normalization
Cui 2019 [86] Visual saliency modulated JND
1970
1980
1990
2000
2010
2020
Chua 1988 [115] MLP-based transform coding
Dianat 1991 [141] MLP-based predictive coding
Namphol 1996 [142] Hierarchical MLP-based coding
Chen 2017 [130] DeepCoder: CNN based video coding
Chen 2019[131] Pixel-motion CNN for predictive coding
Liu 2022 [136] Hybrid motion compensation for video coding
Gelenbe 1994 [144] Random neural network based image coding
Hai 2001 [145] Random neural network based video coding
Ballé 2016 [120] VAE-based image coding
Toderici 2017 [118] RNN-based image coding
Cheng 2020 [125] Attention modules for image coding
Sezer 2008 [109] Sparse orthonor -mal transforms
Gregor 2016 [128] CNN-RNN-based con-ceptual coding
Chang 2019 [179] Deep semantic synthesis con-ceptual coding
Chang 2022 [180] Coding with structure and texture
Zhang 2015 [103] SVM-based coding unit depth decisions
Lombardo 2019 [134] Video gen-erative coding
Wang 2022 [139] Disentangled representations for human body video coding
Amestoy 2019 [105] Random forest classifiers for partition modes
Li 2021 [197] Compress visual data to human-comprehensible domain
Model-based coding
Pixel and block-based coding
Geometric partition coding
Segmentation-based coding
Object-based coding
Knowledge and semantic-based coding
Perceptual coding
Learning-based coding
Statistical learning or sparse representation-based coding
MLP-based coding
NN-based coding
Conceptual coding
Generative coding
Cross-modal coding

Figure 1 The technical roadmap of intelligent video coding methods including model-based and learning-based compression algorithms

pixel data to reconstruct the image sequence. From Musmann's viewpoint [31], MBC includes pixel MBC, block motion MBC, and object MBC, i.e. the first-generation and second-generation methods. In this paper, we would follow Musmann's viewpoint and provide MBC classification to present the historical development of the model from the signal source to the object and the content understanding of the objects, as summarized in Table 1. From Table 1, it is observed that the evolution of MBC, from the statistical pixel and block to the geometric partition and structural segmentation, and from the content-aware object to the understanding of the content including knowledge, semantics, and the knowledge of HVS. Moreover, many coding standards based on MBC have been developed, such as

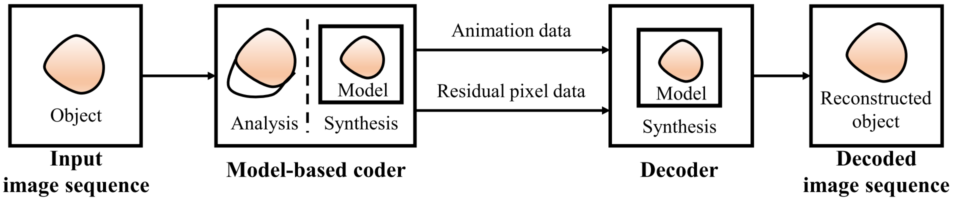


Figure 2 Principle of model-based coding (MBC) from [38]

Table 1 Classification of MBC approaches

Model-based coding	Example of manually designed models	Example of coding schemes
Pixel-based coding	Markov signal model, Gaussian model	PCM
Block-based coding	Block-based translation model	MPEG-x, H.26x series of standards
Geometric partition coding	Triangle, mesh-based model	Mesh-based coding
Segmentation-based coding	Region-based motion model	Contour coding
Object-based coding	Object-based motion model	MPEG-4 object-based coding
Knowledge-based coding	Facial model	Facial model-based image coding
Semantic-based coding	Facial expression model	Model-based facial expression coding
Perceptual coding	HVS model	Texture analysis and synthesis

MPEG-4/7. In this section, we will give a brief introduction to the methods and standards based on MBC.

2.1 Model-based coding methods

In the historical evolution of MBC, pixel model-based video coding, e.g. PCM [44], was later ever used for early memory and computation resource-limited applications, and it was replaced with block-based motion model coding later [45–48]. However, the rectangular partition of block-based coding is rigid and inefficient for modeling irregular visual signals. As a variation of the block-based motion model, more flexible geometric partitions were proposed for motion compensation, including deformable blocks [49], meshes [50, 51] and triangles [52], and they were also studied for H.264/AVC [53, 54], HEVC [55] and VVC [33]. Although geometric partitions are flexible, they are also constrained by their fixed patterns. Therefore, a more flexible and finer-grained partition is based on the input signal itself, such as contour and segmentation, rather than pre-defined geometric partitions. Graham proposed a two-dimensional contour coding in [56], which can be viewed as a predecessor of segmentation coding, and Biggar first formally utilized a segmented image coder with better performance than the transform coder in [57]. Since then, a variety of studies on segmentation-based coding have been performed, including segmentation-based coding [34, 58–62] and segmentation methods [63, 64].

MBC methods mentioned above explore flexible and fine-grained partitions without considering the knowledge of objects or scenes in the world. Since different classes of objects or scenes always exhibit different kinds of appearance and motion patterns, modeling such patterns as knowledge and combining them into coding can further improve the compression ratio for particular image classes. The higher performance also comes with costs that modeling and combining knowledge require considerable manpower for manual design, and knowledge of an object or a scene cannot always be transferred to that of others, resulting in potential limitations on wild scenarios. In the following part of our paper, we review the development of MBC methods using knowledge. Accompanying the emergence of segmentation-based coding, object-based coding is a further prolongation of segmentation coding, where the segmentation may represent one identified object [65–67]. In [65], three parameter sets were used to define the motion, shape, and color of an object, which can be used to reconstruct an image by the model-based image synthesis method. In [66], a generic object-based coding algorithm was proposed relying on the definition of a spatial and temporal segmentation of the sequences. Moreover, object-based coding is further applied to special videos, such as surveillance video or 3D video [68, 69], and motion compensation for codecs [70]. Based on the knowledge of the known objects, knowledge and semantic-based coding methods were developed, such as parameterized modeling for the facial animation [71–76]. Modeling the scene or image content directly is difficult and restricted in wild scenarios; in contrast, perceptual coding [77–86] attempts to incorporate the vision model into the coder by using the knowledge of HVS [87]. In [87], a nonlinear mathematical HVS model was proposed for image compression, which was developed from the psycho-visual and physiological characteristics of the HVS, and a reduced achromatic model was developed as a nonlinear filter fol-

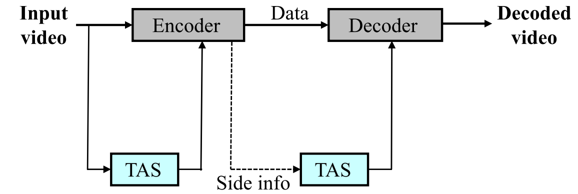


Figure 3 Diagram of the texture analysis and synthesis (TAS) encoder, and texture synthesis (TS) decoder

lowed by a bandpass spatial filter. Texture analysis and synthesis coding, as described in Fig. 3, as cross research of perceptual coding and segmentation coding [79, 88–92], incorporates a texture analyzer in the encoder and a texture synthesizer in the decoder to incorporate the texture information into the coding process.

To achieve higher efficiency compression of audio-visual information with a relatively low bit rate, significant efforts have been devoted by some standardization organizations. MPEG started to develop the international standard MPEG-4 in 1993 [93]. MPEG-4 is based on object-based coding, which concentrates on analyzing and synthesizing the objects in an image [66], which has several advantages over block-oriented schemes, e.g. adaptation to the local image characteristics and object motion compensation as opposed to blockwise motion compensation. In MPEG-4, each picture is considered to be consisting of temporal instances of objects that undergo a variety of changes. Therefore, the concepts of video objects, as well as their temporal instances of video object planes, are introduced in MPEG-4. Specifically, in MPEG-4, each video object is encoded separately and multiplexed into a single bitstream that can be accessed by the users. The encoder sends the video objects and the information about the scene composition for storage and transmission. On the decoder side, the coded data are de-multiplexed and decoded separately, and then the reconstructed objects fuse to the final decoded frame.

MPEG-7 [94] is another standardized attempt at content description representation, which is a multimedia content description standard and was released in 2001. It is different from the previous formats MPEG-1/2/4 in that it does not deal with the coding of moving pictures and audios. MPEG-7 addresses how humans expect to interact with computer systems for it develops rich descriptions that reflect those expectations. It uses XML Schema as the language of choice for content description, allowing fast and efficient searching for material that is of user interest.

Except for MPEG-4/7, some novel MBC methods are explored in other video coding standards, such as screen video coding in HEVC [95] and scene video coding in AVS2 [96]. Screen video refers to the consecutive images generated or rendered by computers or some other electronic devices, and the video may contain computer-generated screen content and natural images/videos. In [95], two new coding tools, residual scalar quantization (RSQ) and base colors and index map (BCIM) were proposed for screen video coding. RSQ directly quantizes the intraprediction residual without applying a transform since screen content often has high contrast and sharp edges. In BCIM, a base color table is created first by clustering. Then, each sample in the block will be quantized to the nearest base color and recorded in the index map. The scene video is captured in specific scenes, such as surveillance video and videos from classrooms, homes, and courts, which are characterized by temporally stable backgrounds. Regarding scene video coding, background modeling schemes [32, 97] were proposed to achieve more accurate prediction without dependence on foreground segmentation. Based on these methods, AVS2 proposed a background-picture-model-based coding method to achieve higher compression performance [96].

3 Learning-based coding

MBC relies on manually designed modules where the components are heavily engineered to fit together. Such a design results in the structure of the signal being manually engineered and thus the capability of MBC to eliminate the redundancy is limited. The motivation of LBC is that with similar components to MBC, LBC models are trained using the massive image and video samples to determine the coding strategy automatically and alleviate the dependence on manually designed coding paradigms based on expert knowledge. With an automatic coding strategy, LBC enables the structure to be automatically discovered to eliminate redundancy more efficiently, which displays the great potential to achieve a better coding performance. In general, the similarity between MBC and LBC is that they share similar components to remove the redundancy in the signal, and the difference is that the former relies on manually designed modules and the latter relies on a data-driven strategy or modules using machine learning. In the literature, numerous LBC approaches have been proposed for coding. LBC can be grouped into three categories, namely statistical learning, sparse representation, and deep learning-based methods.

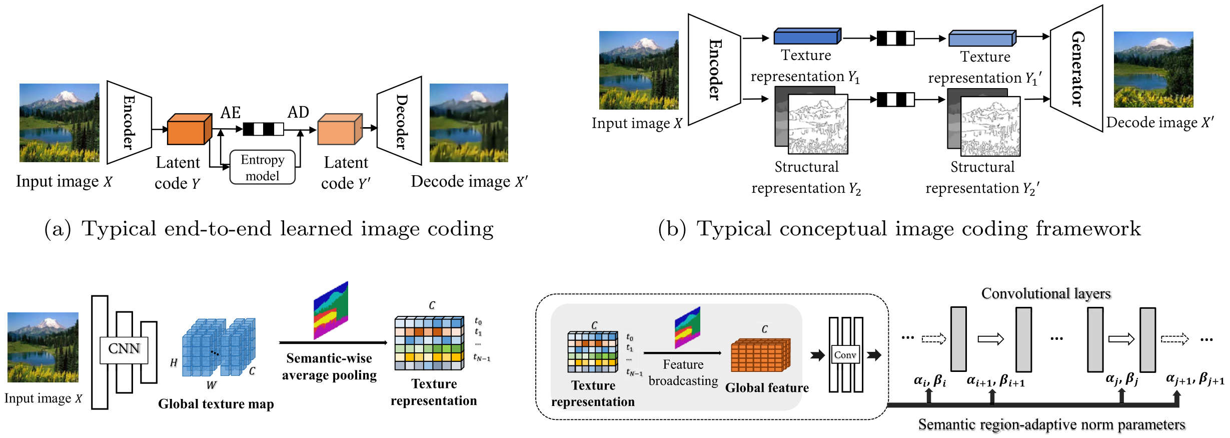
Statistical learning is incorporated into image/video compression to reduce coding complexity or improve the compression performance, such as support vector machine (SVM) [98], Bayesian decision [99], random forest [100], decision tree [101], and AdaBoost [102]. SVM was used as a classifier to determine the early splitting or pruning of a coding unit (CU) [103]. In [104], the Bayesian decision rule was employed with skip states to early terminate the binary-tree (BT) and extended quadtree (EQT) partition. In [105], a random forest classifier was used to determine the most likely partition modes. A fast intra-coding scheme was proposed in [106], where a low complexity coding tree unit (CTU) structure was derived with a decision tree, and the optimal intra mode was decided with the gradient descent principle. AdaBoost is incorporated in [107] as a classifier for CU partition determination. Although these methods are data-driven to discover the best strategy for compression, they are adopted as complex classifiers using manually designed features for coding standards and thus are limited to the scarcity of generalization caused by manually designed features.

A sparse representation of a signal consists of a linear combination of relatively few base elements in a basis or an overcomplete dictionary. Signals that are represented sparsely are termed compressible under the learnable dictionary. Some research efforts were dedicated to learning dictionaries to adapt to a signal class for image compression [108–110]. Bryt and Elad employed a K-SVD (singular value decomposition) dictionary-based facial image codec. They trained K-SVD dictionaries for predefined image patches. The encoding is based on sparse coding of each image patch with the trained dictionary, and the decoding is a simple reconstruction of the patches by the linear combination of atoms. Sezer et al. [109] adopted a concatenation of orthogonal bases as the dictionary, where each basis is selected to encode any given image block of fixed size. Zepeda et al. [110] proposed an iteration-tuned and aligned dictionary (ITAD)-based image [111] codec for particular image classes, such as facial images. ITAD is used as a transform to code image blocks taken over a regular grid. Although some encouraging results were achieved, sparse representation-based coding is designed for particular image classes due to the nature of sparse representation, and thus hard to generalize to wild images encountered in practical scenarios.

Recently, neural networks have been widely explored in image/video coding, which is termed deep learning-based coding. Deep learning-based coding has some advantages over statistical learning and sparse representation-based coding. First, neural networks can mine the underlying characteristics of data and exploit the spatial correlation of textural content, and learn the features adaptively rather than manually designed features. Second, with massive training data, deep learning-based coding can be generalized to wild images and videos. In the following part of this article, we introduce the history of deep learning-based image and video coding methods, which mainly originated in the late 1980s and are based on neural network techniques. Some representative works are listed in Table 2.

Table 2 Representative works of deep learning-based image and video coding

Category	Technique	Method	Highlights	Venue
Image coding	MLP	NNTIC [115]	Neural network	IJCTA 1988
		ICBP [116]	Image patches	ACS 1989
		IDC [117]	Dimension reduction, entropy coding	JCNN 1989
	RNN	ICRNN [118]	Scaled-additive coding	CVPR 2017
		SAIC [119]	Spatially adaptive prediction	ICIP 2017
	VAE	EIC [120]	End-to-end training	arxiv 2016
		NTIC [121]	Transform & quantize method	PCS 2016
		IDCNN [122]	Context model of entropy coding	NIPS 2018
		LIC [123]	Hyperprior	ICLR 2018
		VAIC [124]	Pyramidal feature fusion	CVPRW 2018
		GMLA [125]	Gaussian mixture model	CVPR 2020
	GAN	RAIC [126]	Generative network	ICML 2017
		LFIC [127]	Light field	TCS 2018
		TCC [128]	Conceptual compression	NIPS 2016
		ELIC [129]	Extreme compression	CVPRW 2018
Video coding	VAE	DeepCoder [130]	Feature prediction	VCIP 2017
		LVC [131]	Predictive coding	TCSVT 2019
		LVR [132]	LSTM predictor	ICML 2015
		RDAVC [133]	R-D Autoencoder	ICCV 2019
		DGVC [134]	Local & global feature	NIPS 2019
		CSRVC [135]	Spatiotemporal RNN	TCSVT 2021
		HMC [136]	Compound spatiotemporal representation	TCSVT 2022
	GAN	ADLVC [137]	Feature prediction	CVPRW 2020
		LSVC [135]	ConvLSTM	CVPR 2021
		NTHSVC [138]	3D keypoint extractor	CVPR 2021
		DHBC [139]	Contrastive learning	ICME 2022



(a) Typical end-to-end learned image coding

(b) Typical conceptual image coding framework

(c) A typical texture modeling process of conceptual image coding

(d) A typical image synthesis process of conceptual image coding

Figure 4 The frameworks of the typical end-to-end learned image coding and conceptual coding, as well as the exemplar texture modeling and image synthesis process of conceptual coding

Interested readers may refer to existing reviews for related literature [112–114].

3.1 Learning to compress still images

Multilayer perceptron (MLP) [140] includes an input layer of neurons, several hidden layers of neurons, and an output layer of neurons. This structure provides evidence for scenarios such as dimension reduction and data compression. Chua et al. [115] proposed an end-to-end image compression framework based on the compact representation of the neural network and leveraging high parallelism. The following work [116] trained a fully connected network to compress each 8 × 8 patch of the input image with back propagation. Sonehara et al. [117] proposed a dimension-reduction network to compress the image. In addition, the framework used quantization and entropy coding as individual modules. Furthermore, the MLP-based predictive image coding algorithm [141] was used to exploit the spatial context information. To reduce training time, the nested training algorithm (NTA) was proposed for image compression [142] with an MLP-based hierarchical neural network. A new class of random neural networks [143] was introduced in 1989. Different from MLP, signals in random neural network methods are in the spatial domain. Some researchers have considered the combination of the random neural network and image compression. Gelenbe et al. [144] applied a random neural network in the image compression task, which was further improved in [145] by integrating the wavelet domain of images.

The recurrent neural network (RNN) includes a class of neural networks with memory modules to store recent information. Toderici et al. [118] proposed an RNN-based image compression framework by utilizing a scaled-additive module for coding. Minnen et al. [119] presented a spatially adaptive image compression framework that divided the image into tiles for better coding efficiency.

With the development of CNNs, many deep learning-based frameworks outperform traditional algorithms in both low-level and high-level computer vision tasks [146]. Under the scalar quantization assumption, Ballé et al. [120, 121] introduced an end-to-end optimized neural framework for image compression based on CNNs in 2016. A typical end-to-end learned image coding is illustrated in Fig. 4 (a). During training, Ballé et al. added an i.i.d uniform noise to simulate the quantized operation and replace the stochastic gradient descent approach to avoid zero derivatives. The joint rate-distortion optimization problem can be cast in the context of variational auto-encoders (VAE) [147]. The following work extended the compression model by using scale hyperpriors for entropy estimation [122], which achieved better performance compared with HEVC. Minnen et al. [123] enhanced the context model of entropy coding for end-to-end optimized image compression. Cheng et al. [125] proposed discretized Gaussian mixture likelihoods and attention modules to further improve the performance.

Generative adversarial networks (GAN) are developing rapidly in the application of deep neural networks. Rippel and Bourdev [126] proposed an integrated and well-optimized GAN-based image compression. Inspired by the advances in GAN-based view synthesis, light field (LF) image compression can achieve significant coding gain by generating the missing views using the sampled context views in LF [119]. In addition, Gregor et al. [148] introduced a homogeneous deep generative model DRAW to their coding framework. Different from previous works, Gregor et al. aimed at conceptual compression by generating the image semantic information as much as possible [128]. Agustsson et al. [129] built an extreme image compression system using unconditional and conditional GANs, outperforming all other codecs under low bit-rate conditions. Agustsson et al. [149] proposed using learned perceptual image patch similarity (LPIPS) [150] as the metric for generator training, which further improves the subjective quality of the reconstructed image.

3.2 Learning-based video coding

In this section, we review the development of learning-based video coding. First, we introduce pure learning-based video coding methods. Second, a combination of deep learning and the hybrid video coding framework is presented. Third, we compare these two coding architectures.

Similar to learning-based image coding frameworks, many novel video coding frameworks are built on neural network models to reduce temporal redundancies. As a natural extension of learning-based image coding methods, 3D auto-encoders are proposed to encode the quantized spatiotemporal features with an embedded temporal conditional entropy model. Chen et al. [130] proposed DeepCoder, which combines several CNN networks with a low-profile x264 encoder for video compression. Wu et al. [151] later applied an RNN-based video interpolation module and combined it with a residual coding module for inter-frame coding. Inspired by the prediction for future frames of generative models [152], Srivastava et al. [132] proposed utilizing the long short-term memory (LSTM) encoder-decoder framework to learn video representations, which can be utilized to predict future video frames. Different from Ranzato's work [152] which predicts one future frame, this model can predict a long future sequence into the future. Agustsson et al. [153] further presented a scale-space flow generation and trilinear warping method for motion compensation. Habibian et al. [133] utilized the rate-distortion auto-encoder to directly exploit spatiotemporal redundancy in a group of pictures (GoP) with a temporal conditional entropy model. Lombardo et al. [134] followed the VAE-based image compression framework and encoded this representation according to predictions of the sequential network. With the emergence of GANs, using an auto-encoder combined with adversarial training has been regarded as a promising method. Wang et al [138] demonstrated the use of a novel subject-agnostic face reenactment method for video conferencing, achieving an order of magnitude bandwidth savings over the H.264 standard. With the advantage of adversarial training, at a lower bitrate, different from VAE-based video coding methods that tend to reconstruct blurry videos, GAN-based coding models reconstruct the video with a pleasing perceptual quality.

Following hybrid video coding systems, recent studies have demonstrated the effectiveness of deep learning models from five main modules, i.e. intra-prediction, inter-prediction, quantization, entropy coding, and loop filtering. For intra-prediction, Cui et al. [154] proposed an intra-prediction convolutional neural network (IPCNN) to improve the intra-prediction efficiency. Instead of using CNN, Li et al. [155] proposed a fully connected network (IPFCN) for intra-prediction. In [156], Li et al. explored CNN-based down/up-sampling techniques as a new intra-prediction mode for HEVC. To alleviate the effects of compression noise on the upsampling CNN, Feng et al. [157] designed a dual-network-based superresolution strategy by bridging the low-resolution image and upsampling network using an enhancement network. Inter-prediction is realized by motion estimation on previously coded frames against the current frame in hybrid video coding. Huo et al. [158] utilized variable-filter-size residue-learning CNN (VRCNN) to refine motion compensation for inter-prediction improvement [159]. Yan et al. [160] proposed a fractional pixel reference generation CNN (FRCNN) to predict the fractional pixels for fractional-pixel motion compensation in inter-prediction. Instead of dealing with fractional pixels, some works [161, 162] have directly explored the inter-prediction block generation using CNN-based frame rate up conversion (FRUC). In addition to FRUC, the two nearest bi-directional reference frames in the reference list are utilized as input for the network in [163]. Regarding the limitation of the traditional bidirectional prediction using a simple average of two prediction hypotheses, [161, 164] further improved its efficiency by leveraging a six-layer CNN with a 13 x 13 receptive field size to infer the inter-prediction block in a nonlinear fashion. Utilizing compressed optical flows to directly specify motion is also effective for inter-frame prediction. In addition, bi-directional motion was studied in [165, 166] by additionally exploring the future frames. Both long-term and bi-directional predictions attempt to better characterize complex motion to improve the coding efficiency. Liu et al. [135] used a pyramid optical flow decoder for multi-scale compressed optical flow estimation and applied a progressive refinement strategy with joint feature and pixel domain motion compensation. Zhao et al. [167] adopted previously reconstructed frames, optical flow-based prediction, and a background reference frame to infer the foreground objects of the frame to be coded. In video coding, quantization and entropy coding are the lossy and lossless compression procedures, respectively. In [168], Alam et al. proposed a two-step quantization strategy using neural networks. After quantization, the syntax elements including coding modes and transform coefficients will be fed into the entropy coding engine to further remove their statistical redundancy. Song et al. [169] improved the performance of context-adaptive binary arithmetic coding (CABAC) on compressing the syntax elements of 35 intra-prediction modes by leveraging CNN to directly predict the probability distribution of intra modes instead of the manually designed context models, where CABAC is adopted in HEVC as entropy coding. Loop filtering was proposed to remove compression artifacts. Zhang et al. [170] established a residual highway convolutional neural network (RHCNN) for loop filtering in HEVC. By leveraging the coherence of the spatial and temporal adaptations, Jia et al. [171] improved the performance of a CNN-based loop filter, and designed a spatial-temporal residue network (STResNet)-based loop filter. Moreover, Jia et al. further improved the filtering performance by introducing a content-aware CNN-based loop filter in [172]. More in-loop filters that work with neural networks can be found in [173, 174]. Beyond in-loop filters, some post-filtering algorithms [175, 176] have been proposed to improve the quality of decoded video and images by reducing compression artifacts.

Pure learning-based video coding methods and combined deep learning and hybrid video coding methods have their advantages and disadvantages. The current performance of pure learning-based video coding is developing rapidly and is competitive with traditional video coding. There still exists room for performance improvement. However, the decoding complexity is relatively high, different models are relatively independent, and the bitstreams cannot be interconnected. Combined deep learning and hybrid video coding methods are built upon the traditional hybrid video coding, which has been well developed for several decades, and thus, the performance starting point is relatively higher than that of pure learning-based video coding, which is trained from scratch. However, the combined video coding only replaces some modules by deep learning from hybrid video coding, resulting in different modules that cannot be optimized jointly to achieve higher performance.

3.3 Learning-based coding standards

To enable interoperability between devices manufactured and services provided by different companies, a series of standards targeting intelligent visual data coding have been investigated in the past several years. Several standardization organizations including ISO/IEC (International Organization for Standardization/International Electrotechnical Commission), JPEG (Joint Photographic Experts Group)/MPEG, ITU-T (International Telecommunication Union Telecommunication Standardization Sector), VCEG (Video Coding Experts Group), JVET (Joint Video Experts Team), AVS, IEEE DCSC (Data Compression Standard Committee), MPAI (Moving Picture, Audio and Data Coding by Artificial Intelligence), and others have been creating these standards with many contributions from academia and industry. While most of these visual coding standards have been very successfully deployed in many applications, there are many challenges currently, especially to accommodate the large volume of visual data in limited storage and limited bandwidth transmission links. Compression efficiency improvements are still needed, especially considering emerging data representation formats, from 8K/HDR (high dynamic range) image/video to rich plenoptic formats.

To improve compression efficiency, machine learning technologies, such as deep neural network-based technologies, have shown great potential for many types of visual data. Thus, new standardization activities that exploit this potential are ongoing, some more mature than others, such as learning-based image and video coding, learning-based point cloud coding, and learning-based light-field coding. These standardization efforts attracted significant attention in the aforementioned standardization organizations. The IEEE 1857.11 and JPEG AI group are preparing neural image coding standards in recent years. The MPAI end-to-end video project and enhanced video coding project are also trying to explore neural network-based video coding solutions. The JVET NNVC (neural network-based video coding) and AVS intelligent coding ad-hoc group have released reference models by integrating neural networks into the conventional hybrid framework. All of the above-mentioned standards are advancing neural network-based video coding for future use cases.

4 Our attempts at intelligent coding

LBC compresses the signal data into the compact latent representation containing the non-interpretable knowledge. Moreover, such a mechanism is not analysis-friendly enough to assist downstream machine analysis tasks. A novel LBC paradigm that incorporates more interpretable representation with powerful neural networks may achieve better coding performance, and the interpretable representation may also be beneficial for machine analysis. In this section, we introduce our attempts at such a paradigm, including conceptual image coding, generative video coding, and cross-modal coding.

Inspired by the human visual system (HVS) [177] which perceives visual contents by processing and integrating manifold information into abstract high-level concepts (e.g., structure, texture, and semantics) to form the basis of subsequent cognitive processes [178], conceptual compression has been an active research area in recent years [128, 179–182], following the insights of Marr [183] and Guo et al. [184]. Conceptual coding aims to encode images into compact, high-level interpretable representations for high visual quality reconstruction, allowing a more efficient and analysis-friendly compression architecture. At present, multi-layer decoded representations are integrated to synthesize target images in a deep generative fashion. Herein, the main challenges for conceptual coding include how to achieve efficient representation disentanglement, and how to devise effective generative models for high visual-quality reconstruction. Gregor et al. [128] introduced convolutional deep recurrent attentive writer (DRAW) [148], which extends VAE [147] by using RNNs as encoder and decoder, to transform an image into a series of increasingly detailed representations. However, the interpretability of the learned representations for the image is still insufficient and the models in [128] only worked on datasets of small resolutions. Neural video compression also suffers from similar constraints. Typical video compression methods [134] share the same VAE architecture with image compression methods [128] and transform the original sequence into a lower-dimensional representation. However, the interpretability of the learned representations for video still lacks exploration. Therefore, based on the conventional neural network-based image/video compression in Sect. 3, in this section, we introduce interpretable representations, such as structure information or high-level semantic information, into the compression process to enhance the interpretability of the representations for both images and videos.

4.1 Conceptual image coding

We propose encoding images into two complementary visual components [179, 180] as a milestone for conceptual coding of images. The structure and texture representations are disentangled, as demonstrated in Fig. 4 (b), where a typical texture modeling process is illustrated in Fig. 4 (c). The typical image synthesis process is depicted in Fig. 4 (d). A stylized illustration of disentangled structure and texture representations in domain spaces is proposed in our earlier study in Fig. 5. In our proposed dual-layered model of [179, 180], the structure layer is represented by edge maps, and the texture layer is extracted with the variational auto-encoder in the form of low-dimensional latent variables. To reconstruct the original image from the compressed layered features, our other attempt is to integrate the texture layer and structure layer with adaptive instance normalization adopted using a hierarchical fusion GAN method [180]. The benefits of the proposed conceptual compression framework in [180] have been demonstrated through extensive experiments with extremely low bitrates (<0.1 bpp) and high visual reconstruction quality, as well as content manipulation and analysis tasks through extensive experiments. Nevertheless, it is very challenging to model complex textures of the whole image using only a set of variables. In addition, how to build effective entropy models for visual representations has not been explored for joint rate-distortion optimization. In our recent study in [181], the semantic prior modeling for conceptual coding was proposed. Effective texture representation modeling and compression at semantic granularity are explored for high-quality image synthesis and promising coding efficiency. Moreover, we developed a cross-channel entropy model in [181] for joint texture representation compression and reconstruction optimization. Structural modeling was further introduced in our work [185], which proposed a consistency-contrast learning method to optimize the texture representation space by aligning the representation space with the source pixel space, resulting in higher compression performance. Our proposed models in [181, 185] have achieved superior visual reconstruction quality at ultra-low bitrate (<0.1 bpp) compared to the state-of-the-art VVC in the specific application domain.

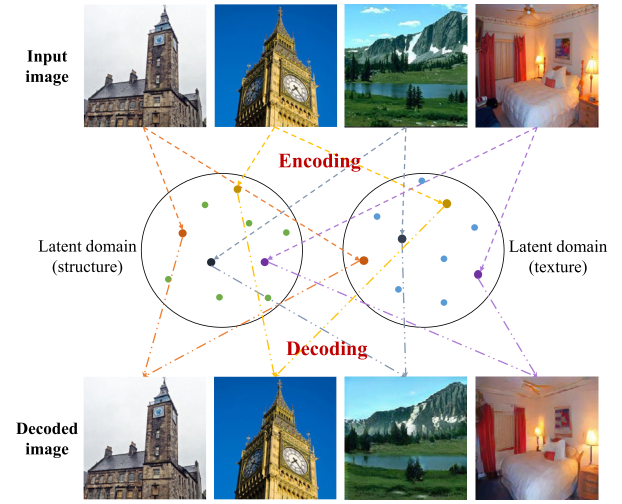
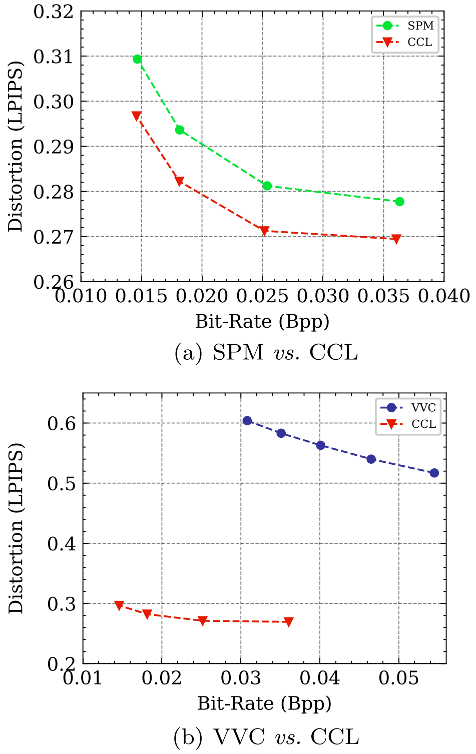


Figure 5 Stylized illustration of the typical conceptual coding

Since the conceptual coding methods pursue visually convincing reconstruction results with minimal bitrate consumption, the LPIPS metric [150] is usually selected as the quantitative perceptual distortion measure except for user study. In our previously established benchmark [186], this metric has been proven to be highly correlated with human visual perception instead of signal fidelity. For performance comparison, the rate-distortion performance in terms of LPIPS of VVC, the typical end-to-end learned image coding method [123] (E2E), our proposed typical conceptual coding methods LCIC [180] and SPM [181] at low bit-rate range over FFHQ [187] and ADE20K [188] outdoor testing sets are displayed in Table 3. The results demonstrate that conceptual coding methods are capable of achieving higher visual reconstruction results at specific domains compared to signal-based compression methods at extremely low bitrates. Moreover, as observed, LCIC behaves less effectively at the more challenging content of ADE20K compared to FFHQ, which consists of regular facial semantic regions. In contrast, SPM achieves remarkable improvements in reconstruction quality on challenging scenes with diverse semantic regions and textures, verifying the effectiveness of the proposed semantic prior modeling mechanism. Moreover, in terms of LPIPS over the ADE20K outdoor testing set, the rate-distortion curves of VVC, SPM [181] and the most recent work CCL [185]

Table 3 The quantitative results of VVC, E2E [123], and our proposed conceptual coding methods LCIC [180] and SPM [181] on the FFHQ, and ADE20K outdoor testing sets. The LPIPS is selected as the distortion metric

Metric	Dataset			
	FFHQ		ADE20K	
	Bitrate (bpp)	LPIPS ($10^{-2}\downarrow$)	Bitrate (bpp)	LPIPS ($10^{-2}\downarrow$)
VVC	0.045	36.9	0.035	58.3
	0.067	29.6	0.040	56.3
	0.075	27.4	0.047	54.0
	0.095	23.1	0.055	51.7
E2E [123]	0.039	33.4	0.016	62.8
	0.067	26.3	0.026	60.2
	0.071	25.6	0.035	53.3
	0.092	24.6	0.052	49.8
LCIC [180]	0.046	27.9	0.036	54.3
	0.055	26.8	0.046	52.0
	0.064	26.1	0.053	50.9
	0.074	25.9	0.061	50.3
SPM [181]	0.049	25.1	0.015	31.0
	0.063	24.1	0.018	29.4
	0.079	23.4	0.025	28.1
	0.110	23.1	0.036	27.8



(a) SPM *vs.* CCL

(b) VVC *vs.* CCL

Figure 6 The rate-distortion curves of SPM [181], CCL [185] and VVC. A lower LPIPS indicates better quality

are shown in Fig. 6. The comparison results verify the improvement in reconstruction quality brought by applying their proposed consistency-contrast learning method.

Compared to previous works, the proposed conceptual image coding demonstrates the superiority towards efficient visual representation learning, high-efficiency image compression (<0.1 bpp), better visual reconstruction quality, and intelligent visual applications (e.g., manipulation and analysis).

4.2 Generative video coding

Due to the powerful capability of deep generative models, many approaches [134] map the video sequences into latent representations and formulate the framework through generative networks to achieve low-bitrate compression. Based on the image animation model, such as FOMM [189], Konuko et al. [190] developed a generative compression framework for video conferencing. Wang et al. [138] also proposed a neural talking-head video synthesis model for video conference by adaptively extracting 3D keypoints from the input videos, achieving the same visual quality as the H.264/AVC [191] with only one-tenth of the bandwidth. Nevertheless, designing a video compression framework targeting high visual quality under extreme compression ratios (e.g., 1000 times) remains unsolved.

Motivated by recent attempts at layered conceptual image compression, we made the first attempt to utilize disentangled visual representations for extreme human body video compression, DHVC [139]. On the encoder side, the input video sequence is disentangled into structure and texture representations for further efficient compression. A pre-trained structure encoder is adopted to estimate the human pose keypoints of each frame. Similar to motion vectors in traditional video codecs, the displacements of each keypoint coordinate are computed as a feature to represent the motion information between two frames. For bitrate saving, only the structure code of the first frame and the motion codes of subsequent frames are transmitted during encoding. On the other hand, a texture encoder extracts the first frame into a semantic-level texture code that represents the texture information of the input video sequence. To ensure texture consistency across all frames, we introduce contrastive learning [192] for the alignment of texture representations. On the decoder side, the structure codes are reconstructed iteratively while the generator restores the video from texture codes and structure codes. Finally, entropy estimation of texture codes is introduced to establish rate-distortion optimization together with contrastive learning for end-to-end training of the framework, promoting bitrate saving and better reconstruction.

As depicted in Fig. 7, the main structure information of the human body can be efficiently represented by human pose keypoints. A pre-trained pose estimator [193] is employed as the structure encoder E_s to extract the structure information of each frame as the compact structure code. The texture encoder E_t aims to extract image frames into texture representations. To better capture the texture details of each frame, we adopt the decomposed component encoding (DCE) module [194] for semantic-aware texture code embedding.

To assure the texture consistency of all frames in the same video, contrastive learning [192] is introduced for training the texture encoder E_t. Instead of using augmentations for building positive samples, the frames in the same video are well-suited for constructing positive samples. Meanwhile, frames in different videos are regarded as negative samples. Moreover, the framework proposes contrastive learning at the semantic level and computes the semantic-wise infoNCE loss [195] with Eq. (1),

$$\mathcal{L}_{cst} = -\sum_{i=1}^{L} \log \frac{\exp(t_i \cdot t_i^{+}/\tau)}{\sum_{j=1}^{Q} \exp(t_i \cdot t_{ij}^{-}/\tau)}, \tag{1}$$

where t_i, t_i^{+}, t_i^{-}, τ, L, and Q denote semantic-wise texture parts of an input frame, another frame in the same video, other frames in different videos, a temperature parameter, the number of semantic regions of the image, and the length of negative sets, respectively. This technique enables the encoder to utilize both the similarity of the positive pair $(\mathbf{t}, \mathbf{t}^{+})$ and the dissimilarity of the negative pairs $(\mathbf{t}, \mathbf{t}^{-})$. Following MoCo [192], a queue is used for storing negative samples t_i^{-} of previous input frames. In this way, the module conducts contrastive learning efficiently with small batch sizes.

For compression comparisons, the average LPIPS and DISTS results of the Fashion and TaichiHD datasets are shown in Table 4. Noticeably, the bitrate of other compared methods is adjusted slightly above our method. Nevertheless, the proposed framework outperforms all other compression frameworks with the lowest LPIPS and DISTS scores at ultra-low bitrates. Moreover, the quantitative results in Table 4 further validate that integrating with con-

Table 4 Comparisons with state-of-the-art video compression methods. Lower scores represent better visual quality. "w/o c." denotes the proposed model without the proposed contrastive learning techniques

	Fashion [196]		Taichi [189]	
	LPIPS↓	DISTS↓	LPIPS↓	DISTS↓
VVC	0.2687	0.3009	0.3147	0.2709
ArtAni	0.1777	0.2273	0.3011	0.2491
Proposed (w/o c.)	0.1109	0.1687	0.2153	0.2206
Proposed	**0.1028**	**0.1604**	**0.2028**	**0.1987**

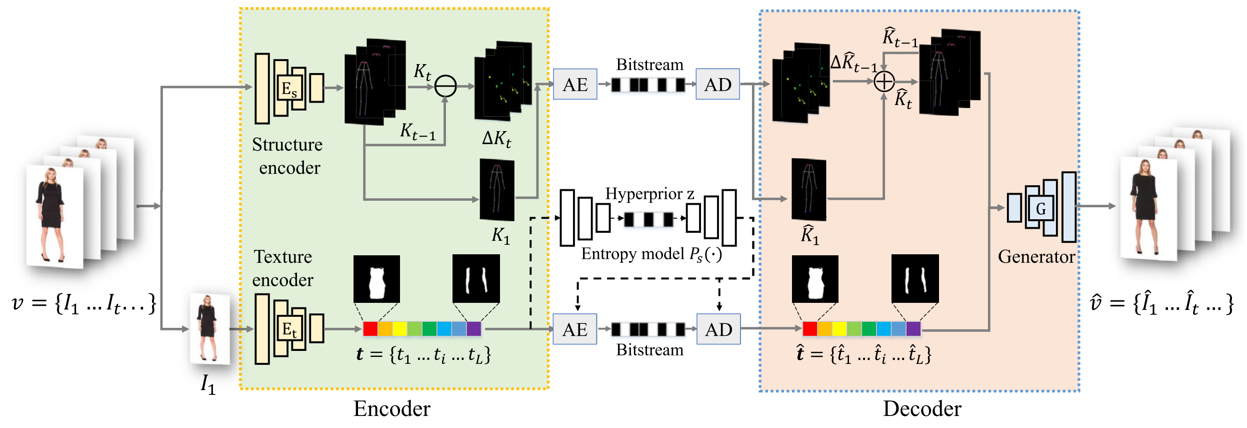

$v = \{I_1 \dots I_t \dots\}$
I_1
E_s
Structure encoder
K_t
K_{t-1}
ΔK_t
K_1
Texture encoder
E_t
$\boldsymbol{t} = \{t_1 \dots t_i \dots t_L\}$
AE
Bitstream
AD
Hyperprior z
Entropy model $P_s(\cdot)$
AE
Bitstream
AD
$\hat{K}_{t-1}$
$\Delta\hat{K}_{t-1}$
$\hat{K}_t$
$\hat{K}_1$
$\hat{\boldsymbol{t}} = \{\hat{t}_1 \dots \hat{t}_i \dots \hat{t}_L\}$
G
Generator
$\hat{v} = \{\hat{I}_1 \dots \hat{I}_t \dots\}$
Encoder
Decoder

Figure 7 The proposed pipeline using disentangled visual representation for video compression

trastive learning facilitates better visual qualities. In general, our method achieves superior visual quality compared to previous methods due to its disentangled texture and structure representations, resulting in sharper results with more details retained, such as facial features and intricate backgrounds.

4.3 Cross-modal coding

Conceptual compression frameworks encode images into representations, such as latent variables extracted from deep neural networks, which are not human-comprehensible. Human comprehensible representations, such as text, sketch, semantic map, and attributions, are significant for various applications, such as semantic monitoring and human-centered applications. Semantic monitoring aims to monitor the semantic information, such as identification, human traffic, or car traffic, rather than the raw signal or latent variables. Human-centered applications aim to directly convey the human-comprehensible information of visual data to human users. Therefore, we proposed cross-modal compression (CMC) [197] to take a step forward to transform the highly redundant visual data into a compact, human-comprehensible representation with ultra-high compression ratios.

We proposed a CMC framework, as illustrated in Fig. 8, which consists of four submodules: CMC encoder, CMC decoder, compression domain encoder, and compression domain decoder. The compressing procedure also consists of four steps. First, the CMC encoder compresses the raw signal into a compact and human-comprehensible representation. Second, the compression domain encoder encodes the representation to a bitstream in a lossless way. Third, the compression domain decoder reconstructs the representation from the bitstream in a lossless way. Finally, the CMC decoder reconstructs the signal from the representation with semantic consistency. The bitrate is optimized by finding a compact compression domain, while the distortion is optimized by preserving the semantics in the CMC encoder and decoder.

Under such a framework, we will further introduce a paradigm. With the recent advances of image captioning [198] and text-guided image generation [199], generating high-quality text from images and generating high-quality images from the text are more feasible. Therefore, we built an efficient image-text-image CMC paradigm, where the images are compressed into the text domain, which is compact, common, and human-comprehensible. Specifically, a classical CNN-RNN model [198] is adopted as the CMC encoder to compress the image to text, where the image feature is extracted from a CNN with the image as input, and fed to an RNN to generate the text in an autoregressive way. Huffman coding [1] can be used as the compression domain encoder/decoder to reduce the statistical redundancy of text in a lossless way. AttnGAN [199] is used as the CMC decoder to reconstruct images from the text due to its promising performance on text-to-image generation. The effectiveness of CMC is verified via various experiments on several datasets, and the model has achieved encouraging reconstructed results with an ultra-high compression ratio (4000-7000 times), showing better compression performance than the widely used JPEG baseline [200].

5 Open discussion

Considering the rapid growth of intelligent video coding, it is expected that a more advanced and insightful model will be developed in the near future, further facilitating the coding and representation efficiency of visual signals. Nevertheless, the field of intelligent video coding poses many new research challenges. Below are a few evolving and significant challenges that need to be addressed.

Domain and profiling There is considerable discussion in the video coding standards community regarding the

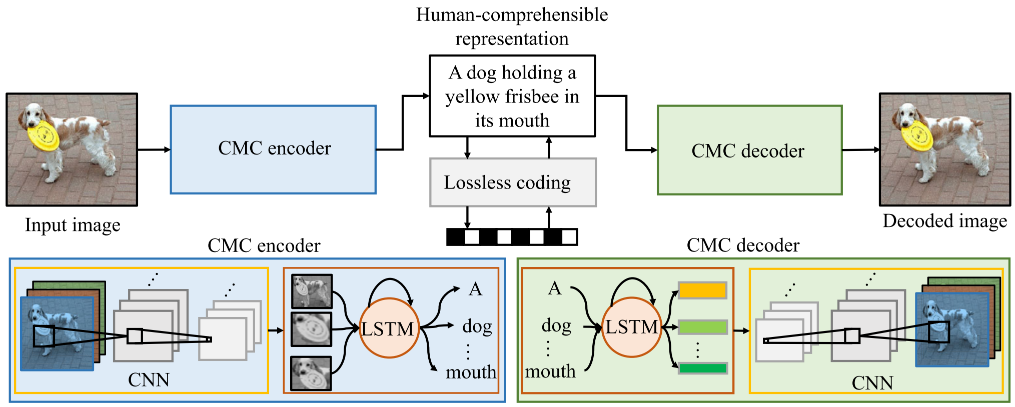


Figure 8 Illustration of the cross-modal compression (CMC) framework

definition of interoperability and conformance testing. To enable intelligent-video-coding-compliant terminals and systems to decode latent representations without ambiguity, it is necessary to standardize them by defining the appropriate rules and assigning them to syntax elements. At the system level, structural, semantic, and textual representations should be parsed correctly by compatible structural, semantic, or textural decoders. Meanwhile, intelligent-video-coding-compliant networks should be able to understand and process the meanings of the latent representations at the intelligent model level. However, visualizing or analyzing bitstreams of highly compact latent representations poses a considerable challenge in assessing the semantic conformance of existing intelligent video codecs. As such, the introduction of profiles may contribute to defining unambiguous conformance procedures and ensuring interoperability for intelligent video coding. Video coding standards have used profiles and levels to define tools with a restricted level of complexity suitable for specific applications. Similarly, intelligent video coding requires different subsets of latent representations for different applications. Some specialized applications may also need restrictions or extensions of the latents. In this regard, it is a critical issue as to how it should support extensions and specialization in specific domains while at the same time ensuring unambiguous conformance validation, requiring a nontrivial effort.

Data security In the context of intelligent video coding, latent representations derived from networks involving signal information can be used to reconstruct the entire video stream. Such representations, however, are not encrypted, and therefore pose the risk of sensitive information leakage. As such, trustworthy and robust coding network design plays a central role in real-world applications.

Representation interpretability To enhance the supporting ability for downstream tasks using the compressed data, it is important to develop latent representations that are highly interpretable. By using such representations, it becomes possible to apply interactive coding techniques, which can enable a range of novel applications such as content editing and immersive interaction. This opens up new opportunities for compression-based approaches to provide versatile features and functionalities beyond traditional video compression methods.

Generalization ability When standardized coding methods and technologies are ready for implementation and deployment, it becomes crucial to identify the path that intelligent video coding would follow to gain entry into practical application domains while satisfying the objectives that such codecs could satisfy versatile requirements. For example, some intelligent video codecs trained for outdoor scenes might not be an ideal choice for coding facial images. It is not practical to employ multiple models for scene adaptation. Furthermore, the active efforts to harmonize the intelligent video coding standard with other media data standards will facilitate and expedite its adoption in practical domains (e.g., short video on mobile devices and immersive media applications).

6 Conclusion

Intelligent video compression provides a comprehensive suite of compactly representing visual media with the capability of describing intrinsic semantics, which also has the potential to revolutionize current and future multimedia coding applications. In particular, such methods include latent codes for describing the structure, semantics, or motion of the visual data, which facilitate efficient editing, analysis, reconstruction of the decoded data, and access to the data. In addition, extracted latent codes can also describe content preferences and support on-the-fly manipulation and transfer of customized content and styles. In this review, the development roadmap for the history of intelligent video coding has been revisited, along with the methodology for describing the structure and semantics of video data. Furthermore, the paper presents three potential research directions in conceptual coding, cross-modality coding, and generative coding that could potentially provide promising solutions to future visual media coding utility and application scenarios. As a final point, a few evolving and significant challenges are discussed regarding future intelligent video coding deployment in practical real-world scenarios.

Acknowledgements
The High-Performance Computing platform of Peking University (PKU-HPC) is gratefully acknowledged.

Funding
This work was supported by the National Natural Science Foundation of China (Grant No. 62025101, 62088102, 62101007 and 61931014), and the Young Elite Scientist Sponsorship Program by the Beijing Association of Science and Technology (Grant No. BYSS2022019).

Abbreviations
IVC,intelligent video coding; MBC, model-based coding; LBC, learning-based coding; DCT, discrete cosine transform; DPCM, differential pulse-code modulation; MPEG, moving picture experts group; AVC, advanced video coding; AVS, audio and video coding standard in China; VVC, versatile video coding; PCM, pulse code modulation; 3DV, three-dimension video; PSNR, peak signal noise ratio; HVS, human visual system; CNN, convolution neural network; TAS, texture analysis and synthesis; TS, texture synthesis; RSQ, residual scalar quantization; BCIM, base colors and index map; SVM, support vector machine; CU, coding unit; BT, binary-tree; EQT, extended quad-tree; CTU, coding tree unit; SVD, singular value decomposition; ITAD, iteration-tuned and aligned dictionary; MLP, multilayer perception; NTA, nested training algorithm; VAE, variational autoencoders; GAN, generative adversarial network; LF, light field; LPIPS, learned perceptual image patch similarity; LSTM, long short term memory; GoP, group of pictures; VRCNN, variable-filter-size residue-learning convolution neural network; FRCNN, fractional pixel reference generation convolution neural network; IPCNN, intra-prediction convolutional neural

network; FRUC, frame rate up conversion; CABAC, context-adaptive binary arithmetic coding; STResNet, spatial-temporal residue network; ISO/IEC, international organization for standardization/international electrotechnical commission; JPEG, joint photographic experts group; ITU-T, international telecommunication union telecommunication standardization sector; VCEG, video coding experts group; JVET, joint video experts team; DCSC, data compression standard committee; MPAI, moving picture, audio and data coding by artificial intelligence; HDR, high dynamic range; NNVC, neural network-based video coding; DRAW, deep recurrent attentive writer; DCE, decomposed component encoding; CMC, cross modal compression.

Availability of data and materials

The datasets generated during and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Declarations

Competing interests

The authors declare no competing interests.

Author contributions

SM is the leader of the project and has constructed the entire framework for the review. All authors contributed to the study conception and design. Literature search and data analysis were performed by JG. Material preparation, data collection and analysis were performed by RW and JC. The first draft of the manuscript was written by JG, CJ and ZH. QM revised the manuscript and discuss the structure of the paper. All authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Author details

[1]National Engineering Research Center of Visual Technology, School of Computer Science, Peking University, Beijing, 100871, China. [2]State Key Laboratory of Media Convergence and Communication, Communication University of China, Beijing, 100024, China. [3]Wangxuan Institue of Computer Technology, Peking University, Beijing, 100871, China.

Received: 19 December 2022 Revised: 31 March 2023
Accepted: 15 June 2023 Published online: 02 August 2023

References

1. Huffman, D. A. (1952). A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, *40*(9), 1098–1101.
2. Golomb, S. (1966). Run-length encodings. *IEEE Transactions on Information Theory*, *12*(3), 399–401.
3. Andrews, H., & Pratt, W. (1968). Fourier transform coding of images. In *Hawaii international conference on system sciences* (pp. 677–679).
4. Pratt, W. K., Kane, J., & Andrews, H. C. (1969). Hadamard transform image coding. *Proceedings of the IEEE*, *57*(1), 58–68.
5. Ahmed, N., Natarajan, T., & Rao, K. R. (1974). Discrete cosine transform. *IEEE Transactions on Computers*, *100*(1), 90–93.
6. Harrison, C. W. (1952). Experiments with linear prediction in television. *The Bell System Technical Journal*, *31*(4), 764–783.
7. Seyler, A. J. (1962). The coding of visual signals to reduce channel-capacity requirements. *Proceedings of the IEE-Part C: Monographs*, *109*(16), 676–684.
8. Taki, Y., Hatori, M., & Tanaka, S. (1974). Inter frame coding that follows the motion. In *Proceedings of institute of electronics and communication engineers of Japan (IECE) annual convention* (pp. 1263). Tokyo: IEICE.
9. Netravali, A. N., & Stuller, J. A. (1979). Motion-compensated transform coding. *The Bell System Technical Journal*, *58*(7), 1703–1718.
10. Reader, C. History of video compression (draft). Retrieved March 30, 2023, from https://www.itu.int/wftp3/av-arch/jVt-site/2002_07_Klagenfurt/JVT-D068.doc.
11. Wiegand, T., Sullivan, G. J., Bjontegaard, G., & Luthra, A. (2003). Overview of the h. 264/avc video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, *13*(7), 560–576.
12. Fan, L., Ma, S., & Wu, F. (2004). Overview of AVS video standard. In *IEEE international conference on multimedia and expo* (Vol. 1, pp. 423–426). Los Alamitos: IEEE.
13. Ma, S., Zhang, L., Wang, S., Jia, C., Wang, S., Huang, T., et al. (2022). Evolution of AVS video coding standards: twenty years of innovation and development. *Science China. Information Sciences*, *65*(9), 1–24.
14. Zhang, J., Jia, C., Lei, M., Wang, S., Ma, S., & Gao, W. (2019). Recent development of AVS video coding standard: AVS3. In *IEEE picture coding symposium* (pp. 1–5). Los Alamitos: IEEE.
15. Ma, S., Huang, T., Reader, C., & Gao, W. (2015). AVS2? Making video coding smarter [standards in a nutshell]. *IEEE Signal Processing Magazine*, *32*(2), 172–183.
16. Sullivan, G. J., Ohm, J.-R., Han, W.-J., & Wiegand, T. (2012). Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on Circuits and Systems for Video Technology*, *22*(12), 1649–1668.
17. Bross, B., Wang, Y.-K., Ye, Y., Liu, S., Chen, J., Sullivan, G. J., & Ohm, J.-R. (2021). Overview of the versatile video coding (vvc) standard and its applications. *IEEE Transactions on Circuits and Systems for Video Technology*, *31*(10), 3736–3764.
18. Limb, J., Rubinstein, C., & Thompson, J. (1977). Digital coding of color video signals-a review. *IEEE Transactions on Communications*, *25*(11), 1349–1385.
19. Netravali, A. N., & Limb, J. O. (1980). Picture coding: a review. *Proceedings of the IEEE*, *68*(3), 366–406.
20. Jain, A. K. (1981). Image data compression: a review. *Proceedings of the IEEE*, *69*(3), 349–389.
21. Musmann, H. G., Pirsch, P., & Grallert, H.-J. (1985). Advances in picture coding. *Proceedings of the IEEE*, *73*(4), 523–548.
22. Sikora, T. (1997). MPEG digital video-coding standards. *IEEE Signal Processing Magazine*, *14*(5), 82–100.
23. Haskell, B. G., Howard, P. G., LeCun, Y. A., Puri, A., Ostermann, J., Civanlar, M. R., et al. (1998). Image and video coding-emerging standards and beyond. *IEEE Transactions on Circuits and Systems for Video Technology*, *8*(7), 814–837.
24. Sullivan, G. J., Topiwala, P. N., & Luthra, A. (2004). The h. 264/AVC advanced video coding standard: overview and introduction to the fidelity range extensions. In *SPIE Conference on applications of digital image processing XXVII* (Vol. 5558, pp. 454–474). Bellingham, Washington, SPIE.
25. Schwarz, H., Marpe, D., & Wiegand, T. (2007). Overview of the scalable video coding extension of the h. 264/AVC standard. *IEEE Transactions on Circuits and Systems for Video Technology*, *17*(9), 1103–1120.
26. Vetro, A., Wiegand, T., & Sullivan, G. J. (2011). Overview of the stereo and multiview video coding extensions of the h. 264/MPEG-4 AVC standard. *Proceedings of the IEEE*, *99*(4), 626–642.
27. Kunt, M., Ikonomopoulos, A., & Kocher, M. (1985). Second-generation image-coding techniques. *Proceedings of the IEEE*, *73*(4), 549–574.
28. Civanlar, M. R., Rajala, S. A., & Lee, W. M. (1986). Second generation hybrid image-coding techniques. In *SPIE visual communications and image processing* (Vol. 707, pp. 132–137). Bellingham, Washington, SPIE.
29. Torres, L., & Kunt, M. (2012). *Video coding: the second generation approach*. Berlin: Springer.
30. Reid, M. M., Millar, R. J., & Black, N. D. (1997). Second-generation image coding: an overview. *ACM Computing Surveys*, *29*(1), 3–29.
31. Musmann, H. G. (1995). A layered coding system for very low bit rate video coding. *Signal Processing. Image Communication*, *7*(4–6), 267–278.
32. Zhang, X., Huang, T., Tian, Y., & Gao, W. (2013). Background-modeling-based adaptive prediction for surveillance video coding. *IEEE Transactions on Image Processing*, *23*(2), 769–784.
33. Meng, X., Jia, C., Zhang, X., Wang, S., & Ma, S. (2021). Spatio-temporal correlation guided geometric partitioning for versatile video coding. *IEEE Transactions on Image Processing*, *31*, 30–42.
34. Wang, S., Jia, C., Zhang, X., Wang, S., Ma, S., & Gao, W. (2022). A pixel-level segmentation-synthesis framework for dynamic texture video compression. *IEEE Transactions on Circuits and Systems for Video Technology*, *32*(10), 7077–7091.
35. Harashima, H., Aizawa, K., & Saito, T. (1989). Model-based analysis synthesis coding of videotelephone images–conception and basic study of intelligent image coding. *IEICE Transactions (1976-1990)*, *72*(5), 452–459.
36. Cicconi, P., Reusens, E., Dufaux, F., Moccagatta, I., Rouchouze, B., Ebrahimi, T., & Kunt, M. (1994). New trends in image data compression. *Computerized Medical Imaging and Graphics*, *18*(2), 107–124.
37. Li, H., Lundmark, A., & Forchheimer, R. (1994). Image sequence coding at very low bit rates: a review. *IEEE Transactions on Image Processing*, *3*(5), 589–609.
38. Pearson, D. E. (1995). Developments in model-based video coding. *Proceedings of the IEEE*, *83*(6), 892–906.
39. Aizawa, K., & Huang, T. S. (1995). Model-based image coding advanced video coding techniques for very low bit-rate applications. *Proceedings of the IEEE*, *83*(2), 259–271.

40. Girod, B., Ben Younes, K., Bernstein, R., Eisert, P., Farber, N., Hartung, F., et al. (1996). Recent advances in video compression. In *IEEE international symposium on circuits and systems* (Vol. 2, pp. 580–583).
41. Pereira, F., Chang, S. f., Koenen, R., Puri, A., & Avaro, O. (1999). Introduction to the special issue on object-based video coding and description. *IEEE Transactions on Circuits and Systems for Video Technology*, *9*(8), 1144–1146.
42. Sikora, T. (2005). Trends and perspectives in image and video coding. *Proceedings of the IEEE*, *93*(1), 6–17.
43. Schreiber, W. F., Knapp, C. F., & Kay, N. D. (1959). Synthetic highs—an experimental TV bandwidth reduction system. *Journal of the Society of Motion Picture and Television Engineers*, *68*(8), 525–537.
44. Carbrey, R. L. (1960). Video transmission over telephone cable pairs by pulse code modulation. *Proceedings of the IRE*, *48*(9), 1546–1561.
45. Dinstein, I., Rose, K., & Heiman, A. (1990). Variable block-size transform image coder. *IEEE Transactions on Communications*, *38*(11), 2073–2078.
46. Wien, M., & Ohm, J.-R. (2001). Simplified adaptive block transforms. *Document VCEG-O30*. Retrieved March 30, 2023, from https://www.itu.int/wftp3/av-arch/jvt-site/2002_07_Klagenfurt/JVT-D068.doc.
47. Kim, I.-K., Min, J., Lee, T., Han, W.-J., & Park, J. (2012). Block partitioning structure in the HEVC standard. *IEEE Transactions on Circuits and Systems for Video Technology*, *22*(12), 1697–1706.
48. Huang, Y.-W., An, J., Huang, H., Li, X., Hsiang, S.-T., Zhang, K., et al. (2021). Block partitioning structure in the VVC standard. *IEEE Transactions on Circuits and Systems for Video Technology*, *31*(10), 3818–3833.
49. Seferidis, V. E., & Ghanbari, M. (1993). General approach to block-matching motion estimation. *Optical Engineering*, *32*(7), 1464–1474.
50. Sullivan, G. J., & Baker, R. L. (1991). Motion compensation for video compression using control grid interpolation. In *IEEE international conference on acoustics, speech, and signal processing* (pp. 2713–2714). Los Alamitos: IEEE.
51. Denis, K., & Guillemot, C. (2006). Mesh-based motion-compensated interpolation for side information extraction in distributed video coding. In *IEEE international conference on image processing* (pp. 261–264). Los Alamitos: IEEE.
52. Brusewitz, H. (1990). Motion compensation with triangles. [Paper presentation]. Proceedings of the 3rd international workshop on 64k bits/s coding of moving video, free session, Rotterdam, Netherlands.
53. Escoda, O. D., Yin, P., Dai, C., & Li, X. (2007). Geometry-adaptive block partitioning for video coding. In *IEEE international conference on acoustics, speech and signal processing* (pp. 653–657). Los Alamitos: IEEE.
54. Wang, Q., Ji, X., Sun, M.-T., Sullivan, G. J., Li, J., & Dai, Q. (2012). Complexity reduction and performance improvement for geometry partitioning in video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, *23*(2), 338–352.
55. Guo, L., Yin, P., & Francois, E. (2010). *Te:3 simplified geometry block partitioning*. Technical report JCTVC-B085. Geneva, Switzerland, 2nd meeting: joint collaborative team on video coding (JCT-VC) of ITU-T VCEG and ISO/IEC MPEG.
56. Graham, D. N. (1967). Image transmission by two-dimensional contour coding. *Proceedings of the IEEE*, *55*(3), 336–346.
57. Biggar, M. J., Morris, O. J., & Constantinides, A. G. (1988). Segmented-image coding: performance comparison with the discrete cosine transform. In *IEEE proceedings-f (communications, radar and signal processing)* (Vol. 135, pp. 121–132). Los Alamitos: IEEE.
58. Gilge, M., Engelhardt, T., & Mehlan, R. (1989). Coding of arbitrarily shaped image segments based on a generalized orthogonal transform. *Signal Processing. Image Communication*, *1*(2), 153–180.
59. Kassim, A. A., Lee, W. S., & Zonoobi, D. (2009). Hierarchical segmentation-based image coding using hybrid quad-binary trees. *IEEE Transactions on Image Processing*, *18*(6), 1284–1291.
60. Milani, S., & Calvagno, G. (2011). Segmentation-based motion compensation for enhanced video coding. In *IEEE international conference on image processing* (pp. 1649–1652). Los Alamitos: IEEE.
61. Chen, D., Chen, Q., & Zhu, F. (2019). Pixel-level texture segmentation based AV1 video compression. In *ICASSP 2019-2019 IEEE international conference on acoustics, speech and signal processing (ICASSP)* (pp. 1622–1626). Los Alamitos: IEEE.
62. Wang, Z., Wang, S., Zhang, X., Wang, S., & Ma, S. (2019). Three-zone segmentation-based motion compensation for video compression. *IEEE Transactions on Image Processing*, *28*(10), 5091–5104.
63. Marques, F., Vera, V., & Gasull, A. (1994). Hierarchical image sequence model for segmentation: application to region-based sequence coding. In *SPIE visual communications and image processing* (Vol. 2308, pp. 554–563). Bellingham, Washington, SPIE.
64. Gu, C., & Kunt, M. (1994). Very low bit-rate video coding using multi-criterion segmentation. In *IEEE international conference on image processing* (Vol. 2, pp. 418–422). Los Alamitos: IEEE.
65. Musmann, H. G., Hötter, M., & Ostermann, J. (1989). Object-oriented analysis-synthesis coding of moving images. *Signal Processing. Image Communication*, *1*(2), 117–138.
66. Salembier, P., Marqués, F., Pardas, M., Morros, J. R., Corset, I., Jeannin, S., et al. (1997). Segmentation-based video coding system allowing the manipulation of objects. *IEEE Transactions on Circuits and Systems for Video Technology*, *7*(1), 60–74.
67. Torres, L., García, D., & Mates, A. (1997). On the use of layers for video coding and object manipulation. In *Erlangen symposium, advances in digital image communication* (pp. 65–73). Erlangen: University of Erlangen-Nuremberg.
68. Vetro, A., Haga, T., Sumi, K., & Sun, H. (2003). Object-based coding for long-term archive of surveillance video. In *IEEE international conference on multimedia and expo* (Vol. 2, pp. II–417). Los Alamitos: IEEE.
69. De Silva, D. V. S. X., Fernando, W. A. C., & Yasakethu, S. L. P. (2009). Object based coding of the depth maps for 3d video coding. *IEEE Transactions on Consumer Electronics*, *55*(3), 1699–1706.
70. Asikuzzaman, M., Ahmmed, A., Pickering, M. R., & Sikora, T. (2020). Edge oriented hierarchical motion estimation for video coding. In *IEEE international conference on image processing* (pp. 1221–1225). Los Alamitos: IEEE.
71. Parke, F. I. (1982). Parameterized models for facial animation. *IEEE Computer Graphics and Applications*, *2*(9), 61–68.
72. Aizawa, K., Harashima, H., & Saito, T. (1989). Model-based analysis synthesis image coding (mbasic) system for a person's face. *Signal Processing. Image Communication*, *1*(2), 139–152.
73. Antoszczyszyn, P. M., Hannah, J. M., & Grant, P. M. (1998). Reliable tracking of facial features in semantic-based video coding. *IET Vision, Image and Signal Processing*, *145*(4), 257–263.
74. Hu, M., Worrall, S., Sadka, A. H., & Kondoz, A. M. (2004). Automatic scalable face model design for 2D model-based video coding. *Signal Processing. Image Communication*, *19*(5), 421–436.
75. Hou, J., Chau, L.-P., Zhang, M., Magnenat-Thalmann, N., & He, Y. (2014). A highly efficient compression framework for time-varying 3D facial expressions. *IEEE Transactions on Circuits and Systems for Video Technology*, *24*(9), 1541–1553.
76. Yu, J., Luo, C., Yu, L., Li, L., & Wang, Z. (2016). Facial video coding/decoding at ultra-low bit-rate: a 2D/3D model-based approach. *Multimedia Tools and Applications*, *75*(19), 12021–12041.
77. Safranek, R. J., & Johnston, J. D. (1989). A perceptually tuned sub-band image coder with image dependent quantization and post-quantization data compression. In *IEEE international conference on acoustics, speech, and signal processing* (pp. 1945–1948). Los Alamitos: IEEE.
78. Jayant, N., Johnston, J., & Safranek, R. (1993). Signal compression based on models of human perception. *Proceedings of the IEEE*, *81*(10), 1385–1422.
79. Zhang, Y., Ji, X., Zhao, D., & Gao, W. (2006). Video coding by texture analysis and synthesis using graph cut. In *Pacific-RIM conference on multimedia* (pp. 582–589). Berlin: Springer.
80. Ma, L., Ngan, K. N., Zhang, F., & Li, S. (2011). Adaptive block-size transform based just-noticeable difference model for images/videos. *Signal Processing: Image Communication*, *26*(3), 162–174.
81. Wang, S., Rehman, A., Wang, Z., Ma, S., & Gao, W. (2011). SSIM-inspired divisive normalization for perceptual video coding. In *IEEE international conference on image processing* (pp. 1657–1660). Los Alamitos: IEEE.
82. Wang, S., Rehman, A., Wang, Z., Ma, S., & Gao, W. (2012). Perceptual video coding based on SSIM-inspired divisive normalization. *IEEE Transactions on Image Processing*, *22*(4), 1418–1429.
83. Wang, S., Ma, S., Zhao, D., & Gao, W. (2014). Lagrange multiplier based perceptual optimization for high efficiency video coding. In *IEEE signal and information processing association annual summit and conference* (pp. 1–4). Los Alamitos: IEEE.
84. Luo, F., Wang, S., Zhang, N., Ma, S., & Gao, W. (2016). GPU based sample adaptive offset parameter decision and perceptual optimization for HEVC. In *IEEE international symposium on circuits and systems* (pp. 2687–2690). Los Alamitos: IEEE.

85. Zhang, X., Wang, S., Gu, K., Lin, W., Ma, S., & Gao, W. (2016). Just-noticeable difference-based perceptual optimization for jpeg compression. *IEEE Signal Processing Letters*, *24*(1), 96–100.
86. Cui, J., Xiong, R., Zhang, X., Wang, S., & Ma, S. (2019). Perceptual video coding based on visual saliency modulated just noticeable distortion. In *IEEE data compression conference* (pp. 565–565). Los Alamitos: IEEE.
87. Hall, C. F., & Andrews, H. C. (1978). Digital color image compression in a perceptual space. In *Digital image processing II* (Vol. 149, pp. 182–188). Bellingham: SPIE.
88. Ndjiki-Nya, P., Makai, B., Blattermann, G., Smolic, A., Schwarz, H., & Wiegand, T. (2003). Improved H.264/AVC coding using texture analysis and synthesis. In *IEEE international conference on image processing* (pp. 845–849). Los Alamitos: IEEE.
89. Stojanovic, A., Wien, M., & Ohm, J.-R. (2008). Dynamic texture synthesis for H.264/AVC inter coding. In *IEEE international conference on image processing* (pp. 1608–1611). Los Alamitos: IEEE.
90. Stojanovic, A., & Kosse, P. (2010). Extended dynamic texture prediction for H.264/AVC inter coding. In *IEEE international conference on image processing* (pp. 2045–2048). Los Alamitos: IEEE.
91. Zhu, C., Sun, X., Wu, F., & Li, H. (2007). Video coding with spatio-temporal texture synthesis. In *IEEE international conference on multimedia and expo* (pp. 112–115). Los Alamitos: IEEE.
92. Zhang, F., Bull, D. R., & Canagarajah, N. (2010). Region-based texture modelling for next generation video codecs. In *IEEE international conference on image processing* (pp. 2593–2596). Los Alamitos: IEEE.
93. ISO/IEC (1995). IS 144496-2: information technology - coding of audio-visual objects - part 2: visual (MPEG-4 video).
94. Chang, S.-F., Sikora, T., & Purl, A. (2001). Overview of the MPEG-7 standard. *IEEE Transactions on Circuits and Systems for Video Technology*, *11*(6), 688–695.
95. Lan, C., Xu, J., Wu, F., & Sullivan, G. J. (2010). Screen content coding. *Document JCTVC-B084, Geneva, Switzerland*. Retrieved March 30, 2023, from https://www.itu.int/wftp3/av-arch/jctvc-site/2010_07_B_Geneva/JCTVC-B200_r0.doc.
96. Dong, S., Zhao, L., Xing, P., & Zhang, X. (2013). Surveillance video coding platform for AVS2. [Paper presentation]. The 47th AVS meeting, Shenzhen, China.
97. Zhang, X., Liang, L., Huang, Q., Liu, Y., Huang, T., & Gao, W. (2010). An efficient coding scheme for surveillance videos captured by stationary cameras. In *Visual communications and image processing 2010* (Vol. 7744, pp. 729–738). Bellingham: SPIE.
98. Suthaharan, S. (2016). Support vector machine. In *Machine learning models and algorithms for big data classification: thinking with examples for effective learning* (pp. 207–235). Berlin: Springer.
99. Ma, W. J. (2019). Bayesian decision models: a primer. *Neuron*, *104*(1), 164–175.
100. Breiman, L. (2001). Random forests. *Machine Learning*, *45*, 5–32.
101. Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, *1*, 81–106.
102. Hastie, T., Rosset, S., Zhu, J., & Zou, H. (2009). Multi-class adaboost. *Statistics and Its Interface*, *2*(3), 349–360.
103. Zhang, Y., Kwong, S., Wang, X., Yuan, H., Pan, Z., & Xu, L. (2015). Machine learning-based coding unit depth decisions for flexible complexity allocation in high efficiency video coding. *IEEE Transactions on Image Processing*, *24*(7), 2225–2238.
104. Wang, M., Li, J., Zhang, L., Zhang, K., Liu, H., Wang, S., & Ma, S. (2019). Fast coding unit splitting decisions for the emergent AVS3 standard. In *2019 picture coding symposium (PCS)* (pp. 1–5). Los Alamitos: IEEE.
105. Amestoy, T., Mercat, A., Hamidouche, W., Menard, D., & Bergeron, C. (2019). Tunable VVC frame partitioning based on lightweight machine learning. *IEEE Transactions on Image Processing*, *29*, 1313–1328.
106. Yang, H., Shen, L., Dong, X., Ding, Q., An, P., & Jiang, G. (2019). Low-complexity CTU partition structure decision and fast intra mode decision for versatile video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, *30*(6), 1668–1682.
107. Zhang, J., Wang, M., Jia, C., Wang, S., Ma, S., & Gao, W. (2022). Scalable intra coding optimization for video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, *32*(10), 7092–7106.
108. Ori, B., & Elad, M. (2008). Compression of facial images using the K-SVD algorithm. *Journal of Visual Communication and Image Representation*, *19*(4), 270–282.
109. Sezer, O. G., Harmanci, O., & Guleryuz, O. G. (2008). Sparse orthonormal transforms for image compression. In *2008 15th IEEE international conference on image processing* (pp. 149–152). Los Alamitos: IEEE.
110. Zepeda, J., Guillemot, C., & Kijak, E. (2011). Image compression using sparse representations and the iteration-tuned and aligned dictionary. *IEEE Journal of Selected Topics in Signal Processing*, *5*(5), 1061–1073.
111. Salvatierra, J. Z. (2010). *New sparse representation methods; application to image compression and indexing*. PhD thesis, Université de Rennes 1.
112. Ma, S., Zhang, X., Jia, C., Zhao, Z., Wang, S., & Wang, S. (2020). Image and video compression with neural networks: a review. *IEEE Transactions on Circuits and Systems for Video Technology*, *30*(6), 1683–1698.
113. Ma, S., Zhang, X., Wang, S., Zhang, X., Jia, C., & Wang, S. (2018). Joint feature and texture coding: toward smart video representation via front-end intelligence. *IEEE Transactions on Circuits and Systems for Video Technology*, *29*(10), 3095–3105.
114. Huang, Z., Lin, K., Jia, C., Wang, S., & Ma, S. (2021). Beyond VVC: towards perceptual quality optimized video compression using multi-scale hybrid approaches. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 1866–1869). Los Alamitos: IEEE.
115. Chua, L. O., & Lin, T. (1988). A neural network approach to transform image coding. *International Journal of Circuit Theory and Applications*, *16*(3), 317–324.
116. Cottrell, G.W., Munro, P., & Zipser, D. (1989). Image compression by back propagation: an example of extensional programming. In N.E. Sharkey (Ed.), *Models of cognition: a review of cognitive science*. Hillsdale: Ablex.
117. Sonehara, N., Kawato, M., Miyake, S., & Nakane, K., (1989). Image data compression using a neural network model. In *International joint conference on neural networks* (Vol. 2, pp. 35–41). Los Alamitos: IEEE.
118. Toderici, G., Vincent, D., Johnston, N., Hwang, S. J., Minnen, D., Shor, J., & Covell, M. (2017). Full resolution image compression with recurrent neural networks. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 5306–5314). Los Alamitos: IEEE.
119. Minnen, D., Toderici, G., Covell, M., Chinen, T., Johnston, N., Shor, J., et al. (2017). Spatially adaptive image compression using a tiled deep network. In *IEEE international conference on image processing* (pp. 2796–2800). Los Alamitos: IEEE.
120. Ballé, J., Laparra, V., & Simoncelli, E. P. (2016). *End-to-end optimized image compression*. arXiv preprint. arXiv:1611.01704.
121. Ballé, J., Laparra, V., & Simoncelli, E. P. (2016). End-to-end optimization of nonlinear transform codes for perceptual quality. In *IEEE picture coding symposium* (pp. 1–5). Los Alamitos: IEEE.
122. Ballé, J., Minnen, D., Singh, S., Hwang, S. J., & Johnston, N. (2018). *Variational image compression with a scale hyperprior*. arXiv preprint. arXiv:1802.01436.
123. Minnen, D., Ballé, J., & Toderici, G. D. (2018). Joint autoregressive and hierarchical priors for learned image compression. In S. Bengio, H. M. Wallach, H. Larochelle, et al. (Eds.), *Advances in neural information processing systems* (Vol. 31, pp. 10771–10780). Red Hook: Curran Associates.
124. Zhou, L., Cai, C., Gao, Y., Su, S., & Wu, J. (2018). Variational autoencoder for low bit-rate image compression. In *IEEE/CVF conference on computer vision and pattern recognition workshops* (pp. 2617–2620). Los Alamitos: IEEE.
125. Cheng, Z., Sun, H., Takeuchi, M., & Katto, J. (2020). Learned image compression with discretized Gaussian mixture likelihoods and attention modules. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 7939–7948). Los Alamitos: IEEE.
126. Rippel, O., & Bourdev, L. (2017). Real-time adaptive image compression. In *International conference on machine learning* (pp. 2922–2930). PMLR.
127. Jia, C., Zhang, X., Wang, S., Wang, S., & Ma, S. (2018). Light field image compression using generative adversarial network-based view synthesis. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, *9*(1), 177–189.
128. Gregor, K., Besse, F., Rezende, D. J., Danihelka, I., & Wierstra, D. (2016). Towards conceptual compression. In D. D. Lee, M. Sugiyama, U. von Luxburg, et al. (Eds.), *Advances in neural information processing systems* (Vol. 29, pp. 3549–3557). Red Hook: Curran Associates.
129. Agustsson, E., Tschannen, M., Mentzer, F., Timofte, R., & van Gool, L. (2018). Extreme learned image compression with GANs. In *IEEE/CVF conference on computer vision and pattern recognition workshops* (pp. 2587–2590). Los Alamitos: IEEE.
130. Chen, T., Liu, H., Shen, Q., Yue, T., Cao, X., & Ma, Z. (2017). Deepcoder: a deep neural network based video compression. In *IEEE visual communications and image processing* (pp. 1–4). Los Alamitos: IEEE.

131. Chen, Z., He, T., Jin, X., & Wu, F. (2019). Learning for video compression. *IEEE Transactions on Circuits and Systems for Video Technology*, *30*(2), 566–576.
132. Srivastava, N., Mansimov, E., & Salakhudinov, R. (2015). Unsupervised learning of video representations using lstms. In *International conference on machine learning* (pp. 843–852). PMLR.
133. Habibian, A., van Rozendaal, T., Tomczak, J. M., & Cohen, T. S. (2019). Video compression with rate-distortion autoencoders. In *IEEE/CVF international conference on computer vision* (pp. 7033–7042). Los Alamitos: IEEE.
134. Lombardo, S., Han, J., Schroers, C., & Mandt, S. (2019). Deep generative video compression. In *Advances in neural information processing systems* (Vol. 32).
135. Liu, B., Chen, Y., Liu, S., & Kim, H.-S. (2021). Deep learning in latent space for video prediction and compression. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 701–710). Los Alamitos: IEEE.
136. Liu, H., Lu, M., Chen, Z., Cao, X., Ma, Z., & Wang, Y. (2022). End-to-end neural video coding using a compound spatiotemporal representation. *IEEE Transactions on Circuits and Systems for Video Technology*.
137. Veerabadran, V., Pourreza, R., Habibian, A., & Cohen, T. S. (2020). Adversarial distortion for learned video compression. In *IEEE/CVF conference on computer vision and pattern recognition workshops* (pp. 168–169). Los Alamitos: IEEE.
138. Wang, T.-C., Mallya, A., & Liu, M.-Y. (2021). One-shot free-view neural talking-head synthesis for video conferencing. In *IEEE/CVF conference on computer vision and pattern recognition*. Los Alamitos: IEEE.
139. Wang, R., Mao, Q., Wang, S., Jia, C., Wang, R., & Ma, S. (2022). Disentangled visual representations for extreme human body video compression. In *IEEE international conference on multimedia and expo* (pp. 1–6). Los Alamitos: IEEE.
140. Gardner, M. W., & Dorling, S. R. (1998). Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmospheric Environment*, *32*(14–15), 2627–2636.
141. Dianat, S. A., Nasrabadi, N. M., & Venkataraman, S. (1991). A non-linear predictor for differential pulse-code encoder (dpcm) using artificial neural networks. In *IEEE international conference on acoustics, speech, and signal processing* (pp. 2793–2794). Los Alamitos: IEEE.
142. Namphol, A., Chin, S. H., & Arozullah, M. (1996). Image compression with a hierarchical neural network. *IEEE Transactions on Aerospace and Electronic Systems*, *32*(1), 326–338.
143. Gelenbe, E. (1989). Random neural networks with negative and positive signals and product form solution. *Neural Computation*, *1*(4), 502–510.
144. Gelenbe, E., & Sungur, M. (1994). Random network learning and image compression. In *IEEE international conference on neural networks* (Vol. 6, pp. 3996–3999). Los Alamitos: IEEE.
145. Hai, F., Hussain, K. F., Gelenbe, E., & Guha, R. K. (2001). Video compression with wavelets and random neural network approximations. In *SPIE applications of artificial neural networks in image processing VI* (Vol. 4305, pp. 57–64).
146. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444.
147. Kingma, D. P., & Welling, M. (2013). *Auto-encoding variational bayes*. arXiv preprint. arXiv:1312.6114.
148. Gregor, K., Danihelka, I., Graves, A., Rezende, D., & Draw, D. W. (2015). A recurrent neural network for image generation. In *International conference on machine learning* (pp. 1462–1471). PMLR.
149. Agustsson, E., Tschannen, M., Mentzer, F., Timofte, R., & Van Gool, L. (2019). Generative adversarial networks for extreme learned image compression. In *IEEE/CVF international conference on computer vision* (pp. 221–231).
150. Zhang, R., Isola, P., Efros, A. A., Shechtman, E., & Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 586–595). Los Alamitos: IEEE.
151. Wu, C.-Y., Singhal, N., & Krahenbuhl, P. (2018). Video compression through image interpolation. In *European conference on computer vision* (pp. 416–431).
152. Ranzato, M., Szlam, A., Bruna, J., Mathieu, M., Collobert, R., & Chopra, S. (2014). *Video (language) modeling: a baseline for generative models of natural videos*. arXiv preprint. arXiv:1412.6604.
153. Agustsson, E., Minnen, D., Johnston, N., Balle, J., Hwang, S. J., & Toderici, G. (2020). Scale-space flow for end-to-end optimized video compression. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 8503–8512). Los Alamitos: IEEE.
154. Cui, W., Zhang, T., Zhang, S., Jiang, F., Zuo, W., Wan, Z., & Zhao, D. (2017). Convolutional neural networks based intra prediction for hevc. In *IEEE data compression conference* (pp. 436–436). Los Alamitos: IEEE.
155. Li, J., Li, B., Xu, J., Xiong, R., & Gao, W. (2018). Fully connected network-based intra prediction for image coding. *IEEE Transactions on Image Processing*, *27*(7), 3236–3247.
156. Li, Y., Liu, D., Li, H., Li, L., Wu, F., Zhang, H., & Yang, H. (2017). Convolutional neural network-based block up-sampling for intra frame coding. *IEEE Transactions on Circuits and Systems for Video Technology*, *28*(9), 2316–2330.
157. Feng, L., Zhang, X., Zhang, X., Wang, S., Wang, R., & Ma, S. (2018). A dual-network based super-resolution for compressed high definition video. In *Pacific rim conference on multimedia* (pp. 600–610). Berlin: Springer.
158. Dai, Y., Liu, D., & Wu, F. (2017). A convolutional neural network approach for post-processing in hevc intra coding. In *International conference on multimedia modeling* (pp. 28–39). Berlin: Springer.
159. Huo, S., Liu, D., Wu, F., & Li, H. (2018). Convolutional neural network-based motion compensation refinement for video coding. In *IEEE international symposium on circuits and systems* (pp. 1–4). Los Alamitos: IEEE.
160. Yan, N., Liu, D., Li, H., Li, B., Li, L., & Wu, F. (2018). Convolutional neural network-based fractional-pixel motion compensation. *IEEE Transactions on Circuits and Systems for Video Technology*, *29*(3), 840–853.
161. Zhao, L., Wang, S., Zhang, X., Wang, S., Ma, S., & Gao, W. (2018). Enhanced ctu-level inter prediction with deep frame rate up-conversion for high efficiency video coding. In *IEEE international conference on image processing* (pp. 206–210). Los Alamitos: IEEE.
162. Zhao, L., Wang, S., Zhang, X., Wang, S., Ma, S., & Gao, W. (2019). Enhanced motion-compensated video coding with deep virtual reference frame generation. *IEEE Transactions on Image Processing*, *28*(10), 4832–4844.
163. Niklaus, S., Mai, L., & Liu, F. (2017). Video frame interpolation via adaptive separable convolution. In *IEEE/CVF international conference on computer vision* (pp. 261–270). Los Alamitos: IEEE.
164. Zhao, Z., Wang, S., Wang, S., Zhang, X., Ma, S., & Yang, J. (2018). Cnn-based bi-directional motion compensation for high efficiency video coding. In *IEEE international symposium on circuits and systems* (pp. 1–4). Los Alamitos: IEEE.
165. Yang, R., Mentzer, F., van Gool, L., & Timofte, R. (2020). Learning for video compression with hierarchical quality and recurrent enhancement. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 6628–6637). Los Alamitos: IEEE.
166. Djelouah, A., Campos, J., Schaub-Meyer, S., & Schroers, C. (2019). Neural inter-frame compression for video coding. In *IEEE/CVF international conference on computer vision* (pp. 6421–6429). Los Alamitos: IEEE.
167. Zhao, L., Wang, S., Wang, S., Ye, Y., Ma, S., & Gao, W. (2022). Enhanced surveillance video compression with dual reference frames generation. *IEEE Transactions on Circuits and Systems for Video Technology*, *32*(3), 1592–1606.
168. Alam, M. M., Nguyen, T. D., Hagan, M. T., & Chandler, D. M. (2015). A perceptual quantization strategy for HEVC based on a convolutional neural network trained on natural images. In *Applications of digital image processing XXXVIII* (Vol. 9599, pp. 395–408). Bellingham: SPIE.
169. Song, R., Liu, D., Li, H., & Wu, F. (2017). Neural network-based arithmetic coding of intra prediction modes in HEVC. In *IEEE visual communications and image processing* (pp. 1–4). Los Alamitos: IEEE.
170. Zhang, Y., Shen, T., Ji, X., Zhang, Y., Xiong, R., & Dai, Q. (2018). Residual highway convolutional neural networks for in-loop filtering in HEVC. *IEEE Transactions on Image Processing*, *27*(8), 3827–3841.
171. Jia, C., Wang, S., Zhang, X., Wang, S., & Ma, S. (2017). Spatial-temporal residue network based in-loop filter for video coding. In *IEEE visual communications and image processing* (pp. 1–4). Los Alamitos: IEEE.
172. Jia, C., Wang, S., Zhang, X., Wang, S., Liu, J., Pu, S., & Ma, S. (2019). Content-aware convolutional neural network for in-loop filtering in high efficiency video coding. *IEEE Transactions on Image Processing*, *28*(7), 3343–3356.
173. Zhao, Y., Lin, K., Wang, S., & Ma, S. (2022). Joint luma and chroma multi-scale CNN in-loop filter for versatile video coding. In *IEEE international symposium on circuits and systems* (pp. 3205–3209). Los Alamitos: IEEE.
174. Lin, K., Jia, C., Zhang, X., Wang, S., Ma, S., & Gao, W. (2022). NR-CNN: nested-residual guided CNN in-loop filtering for video coding. *ACM Transactions on Multimedia Computing Communications and Applications*, *18*(4), 1–22.

175. Dong, C., Deng, Y., Loy, C. C., & Tang, X. (2015). Compression artifacts reduction by a deep convolutional network. In *IEEE/CVF international conference on computer vision* (pp. 576–584). Los Alamitos: IEEE.
176. Zhu, L., Zhang, Y., Wang, S., Yuan, H., Kwong, S., & Ip, H. H.-S. (2018). Convolutional neural network-based synthesized view quality enhancement for 3D video coding. *IEEE Transactions on Image Processing*, *27*(11), 5365–5377.
177. Kruger, N., Janssen, P., Kalkan, S., Lappe, M., Leonardis, A., Piater, J., et al. (2012). Deep hierarchies in the primate visual cortex: what can we learn for computer vision? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(8), 1847–1871.
178. Zhang, Y., Han, K., Worth, R., & Liu, Z. (2020). Connecting concepts in the brain by mapping cortical representations of semantic relations. *Nature Communications*, *11*(1), 1–13.
179. Chang, J., Mao, Q., Zhao, Z., Wang, S., Wang, S., Zhu, H., & Ma, S. (2019). Layered conceptual image compression via deep semantic synthesis. In *IEEE international conference on image processing* (pp. 694–698). Los Alamitos: IEEE.
180. Chang, J., Zhao, Z., Jia, C., Wang, S., Yang, L., Mao, Q., et al. (2022). Conceptual compression via deep structure and texture synthesis. *IEEE Transactions on Image Processing*, *31*, 2809–2823.
181. Chang, J., Zhao, Z., Yang, L., Jia, C., Zhang, J., & Ma, S. (2021). Thousand to one: semantic prior modeling for conceptual coding. In *IEEE international conference on multimedia and expo (ICME)* (pp. 1–6). Los Alamitos: IEEE.
182. Hu, Y., Yang, S., Yang, W., Duan, L.-Y., & Liu, J. (2020). Towards coding for human and machine vision: a scalable image coding approach. In *IEEE international conference on multimedia and expo* (pp. 1–6). Los Alamitos: IEEE.
183. Marr, D. (1982). *Vision: a computational investigation into the human representation and processing of visual information*. San Francisco: W. H. Freeman and Company.
184. Guo, C., Zhu, S.-C., & Wu, Y. N. (2007). Primal sketch: integrating structure and texture. *Computer Vision and Image Understanding*, *106*(1), 5–19.
185. Chang, J., Zhang, J., Xu, Y., Li, J., Ma, S., & Gao, W. (2022). Consistency-contrast learning for conceptual coding. In *ACM international conference on multimedia* (pp. 1–6). New York: ACM.
186. Li, Y., Wang, S., Zhang, X., Wang, S., Ma, S., & Wang, Y. (2021). Quality assessment of end-to-end learned image compression: the benchmark and objective measure. In *ACM international conference on multimedia* (pp. 4297–4305). New York: ACM.
187. Karras, T., Laine, S., & Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 4401–4410). Los Alamitos: IEEE.
188. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., & Torralba, A. (2017). Scene parsing through ADE20K dataset. In *IEEE/CVF conference on computer vision and pattern recognition*. Los Alamitos: IEEE.
189. Siarohin, A., Lathuilière, S., Tulyakov, S., Ricci, E., & Sebe, N. (2019). First order motion model for image animation. In H. M. Wallach, H. Larochelle, A. Beygelzimer, et al. (Eds.), *Advances in neural information processing systems* (Vol. 32, pp. 7135–7145). Red Hook: Curran Associates.
190. Konuko, G., Valenzise, G., & Lathuilière, S. (2021). Ultra-low bitrate video conferencing using deep image animation. In *IEEE international conference on acoustics, speech and signal processing* (pp. 4210–4214). Los Alamitos: IEEE.
191. Richardson, I. E. (2004). *H. 264 and MPEG-4 video compression: video coding for next-generation multimedia*. New York: Wiley.
192. He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *IEEE/CVF conference on computer vision and pattern recognition*. Los Alamitos: IEEE.
193. Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., & Sheikh, Y. (2021). Openpose: realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *43*(1), 172–186.
194. Men, Y., Mao, Y., Jiang, Y., Ma, W.-Y., & Lian, Z. (2020). Controllable person image synthesis with attribute-decomposed gan. In *IEEE/CVF conference on computer vision and pattern recognition*. Los Alamitos: IEEE.
195. van den Oord, A., Li, Y., & Vinyals, O. (2018). *Representation learning with contrastive predictive coding*. arXiv preprint. arXiv:1807.03748.
196. Zablotskaia, P., Siarohin, A., Zhao, B., & Dwnet, L. S. (2019). *Dense warp-based network for pose-guided human video generation*. arXiv preprint. arXiv:1910.09139.
197. Li, J., Jia, C., Zhang, X., Ma, S., & Gao, W. (2021). Cross modal compression: towards human-comprehensible semantic compression. In *ACM international conference on multimedia* (pp. 4230–4238). New York: ACM.
198. Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). Show and tell: a neural image caption generator. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 3156–3164). Los Alamitos: IEEE.
199. Xu, T., Zhang, P., Huang, Q., Zhang, H., Gan, Z., Huang, X., & He, X. (2018). AttnGAN: fine-grained text to image generation with attentional generative adversarial networks. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 1316–1324). Los Alamitos: IEEE.
200. Wallace, G. K. (1990). Overview of the JPEG (ISO/CCITT) still image compression standard. In *Image processing algorithms and techniques* (Vol. 1244, pp. 220–233). Bellingham: SPIE.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.