International Journal of Advances in Scientific Research and Engineering (ijasre)

E-ISSN : 2454-8006

DOI: 10.31695/IJASRE.2024.1.4

Volume 10, Issue 1
January - 2024

Super-tokens Auto-encoders for image compression and reconstruction in IoT applications

Atiampo Kodjo Armand[1], Gokou Hervé Fabrice Diédié[2], N'Takpé Tchimou Euloge[3]

[1]Digital Research and Expertise Unit,
Distance Learning University of Ivory Coast,
Abidjan, Ivory Coast.

[2]Mathematics and Computer Science Laboratory,
Péléforo Gon Coulibaly University,
Korhogo, Ivory Coast.

[3]Mathematics and Computer Science Laboratory,
Université Nangui-Abrogoua University,
Abidjan, Ivory Coast.

ABSTRACT

New telecommunications networks are enabling powerful AI applications for smart cities and transport. These applications require real-time processing of large amounts of media data. Sending data to the cloud for processing is very difficult due to latency and energy constraints. Lossy compression can help, but traditional codecs may not provide enough quality or be efficient enough for resource-constrained devices. This paper proposes a new image compression and processing approach based on variational auto-encoders (VAEs). This VAE-based method aims to efficiently compress images while still allowing for high-quality reconstruction and object detection tasks. The encoder is designed to be lightweight and suitable for devices with limited computing power. The decoder is more complex and uses multi-level vector quantization to reconstruct high-resolution images. This approach allows for a simple encoder on edge devices and a powerful decoder on cloud servers. Key contributions include a low-complexity encoder, a new VAE model based on vector quantization, and a framework for using VAEs in IoT. The first experiments on reconstructed images on CelebA and ImageNet100 datasets show promising results in terms of MS-SSIM, PSNR, MSE and rFID compared to the literature and the ability of our approach to be used in IoT applications. Our approach presents results similar to complex algorithms like compression algorithms BPG in term of trade-off rate-distortion, and hierarchical auto-encoder (HQA) in terms of image reconstruction quality.

Key Words: IoT, Super-tokens, Variation Auto-Encoder, Vector Quantization.

1. INTRODUCTION

The appearance in recent years of very high-speed telecommunications networks, notably 5G [1], or even 6G [2], allows IoT devices to acquire, process, and transmit information in greater quantitie, more voluminous and of all types and in almost real time |3]. This allows the development of applications intended for the creation of smart cities and intelligent transport systems that combine technologies from artificial intelligence, in particular, deep learning and networks, to enable the development of learning-intensive applications. In this type of learning, media data (images or videos) from a source node are sent to remote cloud nodes for more complex analyses. In an IoT environment marked by energy consumption constraints and low storage and calculation resource capacities of the nodes, this type of service which authorizes the deployment of services near the devices must solve the problem of reducing latency, optimization of energy consumption, and use of peripheral memory [4]. Additionally, this type of network uses lossy compression algorithms that transmit information at low rates. The main constraint of a lossy compression algorithm at extremely low bit rates is low distortion and very low perceptual quality in the original data reconstruction processes [5-6]. These algorithms are generally analyzed through the flow-distortion compromise, initially proposed by Shannon [7] which aims to find a balance between the flow R and the distortion d by minimizing $d + \beta R$, where $\beta > 0$ balances the two competing objectives [8]. However, in modern lossy compression systems, high perceptual quality during data reconstruction is often more desirable than low distortions. At low throughput, it is desirable to communicate only high-level concepts and entrust a powerful decoder [6] with the ability to reconstruct the data taking into account hidden contextual information. Following this principle, neural networks present a promising avenue because they are flexible enough to learn the complex transformations needed both to capture high-level concepts and to reconstruct in a convincing manner that avoids artifacts [9].

In this paper, we propose a new image compression and processing approach based on variational auto-encoders [10]. The model aims to provide a native algorithm for data compromise and to reconstruct high-quality images as realistic as possible to enable classification or object detection tasks a posteriori. It provides an efficient data compression mechanism and an algorithm for reconstructing high-quality images similar to the original images usable for classification or object detection tasks. In a context where the encoder needs to be placed in low-power devices, needing to compress and transmit important features for complex tasks to dedicated edge/cloud servers, conventional compression codecs such as VVC [11] and HEVC [12] are most often unsuitable for this type of device because of their computational complexity. As for those based on the JPEG2000 protocol and these extensions, they may be suboptimal in terms of coding efficiency, since they only optimize part of the pipeline [13-14]. More recently, codecs based on vector quantization have emerged highlighting the importance of learning the distribution of codewords for efficient compression [15-17]. However, these encoders are complex and unsuitable for widespread deployment in IoT. Inspired by the work of [18] which used a variational auto-encoder to propose a lightweight encoder capable of efficiently compressing the images to be transmitted and a decoder that allows the images to be classified without reconstructing them, this paper aims to propose a framework that based on variational auto-encoders also provides an efficient encoder which compresses the data in the form of a dictionary and proposes a decoder which reconstructs the original images. The encoder uses residual approaches to capture information with significant semantic content with low computational power and extracts information with a spatial resolution reduced by a factor of 8 times the size of the original image. As for the decoder, much more complex, it uses a multi-level approach to vector quantization introduced by [15]. Unlike the work of [17] which uses a continuous representation of the latent space, we use a discrete representation of the latent space as in [6] at several resolution levels to obtain reconstructed images of high-resolution quality. The latent space is then composed of a dictionary learned from the data and the interactions between the different elements of the dictionary are modelled by self-attention mechanisms [19]. Indeed, in [20], the authors showed that the use of a continuous latent space considerably harmed the synthesis capabilities of the decoder in variational auto-encoders and led them to reconstruct relatively blurry images of poor spatial resolution quality. Furthermore, they show in this same study that the performances of a variational auto-encoder are mainly linked to those of the decoder that constitutes it. This allows us to have a relatively simple encoder that can be implemented in devices with low computing and memory resources and reserve a complex decoder on remote cloud servers. The main contributions of our approach are as follows:

- Development of an encoder with low computational complexity that can be used in low-cost and low-resource consumption systems. We also offer a decoder to synthesize high-resolution quality images that can be used as input to any classifier located on a cloud network
- Proposal of a new auto-encoder model based on vector quantization. This approach uses multi-level vector quantification of spatial resolution starting from the finest levels to the coarsest levels.
- Proposal of a framework for using discrete latent space variational auto-encoders in the context of IoT.

In the remainder of this article, section 2 presents the literature review to highlight the interest of our approach. Section 3 is devoted to a detailed study of our approach. In section 4, we present the experiments and discuss the results obtained compared to those of the best algorithms in the literature in section 5. We end this paper with a conclusion followed by perspectives.

2. LITERATURE SURVEY

The integration of artificial intelligence into IoT networks aims to provide devices with analysis and decision-making capabilities. In the context of learning-based IoT networks, this involves the development of codecs that can be embedded in low-cost equipment. Early solutions used existing codecs such as JPEG [21], JPEG2000[22], H.265/HEVC [12], AV1 [23], and the recent H.266/VVC [11]. These codecs, recognized for their excellent compression qualities, have been designed to minimize the distortions contained in terms of visual quality in the decoded signals and are thus adapted to the human vision system. The codecs for these algorithms transform the signal using orthogonal linear transforms such as discrete cosine transforms (DCT) or discrete wavelet transforms (TOD) to allow the encoder to quantize high spatial frequencies more severely. This is because the human visual system (HVS) is more sensitive to distortions present in high frequencies than in low frequencies. The SVH is also more sensitive to distortions on the luminance component than on the chrominance components. This means that these coding tools favor greater fidelity for the luminance component. Additionally, for a given image quality/distortion, these codecs can generate low-bitrate bitstreams for transmission over low-bandwidth networks. However, in the case of cloud-based learning, these codecs are unsuitable because machine task algorithms may not be accurate on compressed content based on the properties of the human visual system. For example, high spatial frequency features can be useful for object detection and chrominance components can have a significant impact when classifying elements in a scene. Furthermore, the authors of [24] showed that to be effective, cloud learning-based algorithms must follow the same encoding and decoding principles as traditional approaches but must do so in a non-linear manner. More recently, various algorithms [25-28] based on artificial neural networks for image compression have appeared with a bitrate-distortion trade-off equivalent to that of conventional codecs. In these algorithms, the encoder and decoder are constructed as deep neural networks (DNN) instead of linear orthogonal transformations. The auto-encoder transforms the input image into a descriptor vector which is quantized and whose entropy is then encoded in a differentiable manner by the probability distribution estimate. To improve compression efficiency, existing works have explored efficient neural network blocks, such as residual networks [29], self-attention [19], and transformers [30-32], as well as entropy-based models such as hierarchical [33] and auto-regressive models [34-36]. Regardless of their complexity, all these architectures rely on the variational auto-encoder model [37]. Indeed, variational auto-encoders are germinal models that seek to express the distribution $P(x;\ \theta)$ that the input data follows using neural networks. For this, they express the joint distribution of the input data and the latent data by (1)

$$P(x,z\,;\theta)\ =\ P(x\,\middle|\,z\,;\theta)P(z\,;\theta) \qquad (1)$$

where *P(z;θ)* represents the prior distribution of the latent data and *P(x|z;θ)* is the prior distribution of the input data according to the latent data we want as close as possible to $P(x;\theta)$. These distributions being in practice impossible to find, the auto-encoders achieve this by minimizing the following loss function according to (2).

$$L_{vae}\,(x;\theta,\phi,\lambda) = E_{x\sim P(.;\theta)z\sim Q(.;\phi)}\ \left[D_{KL}\big(Q(z|x,\phi)\big|\big|P(z;\ \theta)\big) + \lambda\, d(x,\hat{x})\right] \qquad (2)$$

in which $D_{KL}(.,.)$ *r*epresents the Kullback-Liebler divergence and Q $(z|x,\ \phi)$ represents the inference model (encoder) which generates the latent data z according to the observations x and λ a parameter of regularization. Recent studies [38] have shown that this equation Eq. (2) is an upper limit of the Lagrangian relaxation of the information rate-distortion function [39], showing the close link between the self-variational encoders (VAE) and lossy compression. However, VAEs generally produce images with characteristic and annoying blur because the average pixel variance in the generated images is significantly lower than that of the data in the training set [40]. To overcome this limitation, hierarchical approaches [41], [42], [43], [44] are among the most promising. In this case, the latent variable is partitioned as a group $z = z_{1:N}\ \ = \{z_1, z_2, \dots, z_N\ \}\ of$ latent variables grouped in an auto-regressive manner as shown in (3).

DOI: 10.31695/IJASRE.2024.1.4

$$P(z) = P(z_1, z_2, \ldots, z_N) = P(z_N | z_{<N}) \ldots . P(z_3 | z_2, z_1) P(z_2 | z_1) P(z_1) \qquad (3)$$

In this equation, $z_{<N}$ represents the set $\{z_1, z_2, \ldots, z_{N-1}\}$. The variable z_1 is of low dimensionality while the variable z_N is of high dimensionality. This not only increases the flexibility of the VAE but also allows the information contained in the images to be captured from the coarsest scales to the finest scales. However, these approaches determine latent variables in continuous space, making using a discrete encoder for lossy compression quite difficult for these auto-encoders. One solution approach is to implement relative entropy coding (REC) [45-48] in which samples of latent variables are stochastically encoded. VAE with average code length close to the KL term in equation Eq. (2). However, they either require a fairly long execution time or incur significant code length overhead. Another alternative is to construct VAEs with discrete latent variables. Discrete VAE (DVAE) [47], in which latent variables follow the Bernoulli distribution and vector quantized VAE (VQ-VAE) [5-6] assume categorical latent variables. Due to various statistical challenges, DVAEs and VQ-VAEs are still not capable of optimizing the end-to-end throughput-distortion (RD) trade-off in their current form, as they require on the one hand double encoding of data from the dictionary and the quantized vector and their simultaneous transmission to the decoder. To overcome these limitations and to create a unique model, the authors proposed adjustable compression algorithms depending on the quantified information of the vectors of the latent space descriptor [49], [50] only or by adding information depending on the spatial context. More recently in [51] the authors proposed a modular architecture according to the objectives set by the flow-distortion relationship. However, even if these algorithms are effective, they remain difficult to exploit in the context of IoT devices because of the complexity of their practical implementation. More recently, the authors of [5] proposed a hierarchical architecture of VQ-VAE, and HQA to achieve high compression rates. They show that the combination of latent structure in hierarchical form and a stochastic quantification approach facilitates image compression and allows the reconstruction of high perceptual quality images that retain semantically meaningful features. Likewise, in [15] the authors show that a theoretical reformulation of the ELBO in the form of cross entropy made it possible to construct a relatively lightweight auto-encoder usable in the context of IoT from a discrete quantification of the latent space and provide an output classifier. However, if these two approaches show that it is possible to successfully adapt auto-encoders to the context of IoT applications, the quality of the images reconstructed by [6] approaches remain relatively low resolution while in [51], encoding requires the presence of an adaptive type arithmetic encoder and is not particularly suited to image reconstruction tasks. The approach that we propose compensates for the inadequacies of the last two models, by offering an architecture that reconstructs images with a high-quality resolution while keeping a high compression rate and on the other hand which can be used as input to any classification architecture. The model proposed, as in [5], is based on an approach of decomposing the latent space hierarchically. Unlike the classic approach of discrete quantification, the data dictionary is constructed deterministically from the data, and each vector of the latent space is assigned a cluster by projection into the latent space by self-attention to take into account the spatial context and long-term dependencies.

3. Research Methodology

a. Vector quantization

i. Theoretical formulation

One of the main objectives is to provide a model capable of reconstructing images with high-resolution quality, we relied on the principle of vector quantization auto-encoders (VQVAE) proposed by [6]. Indeed, the goal of a variational auto-encoder is to train a generative model which is to estimate the set of parameters θ which gives the joint probability of the latent data z and the input data x like as shown in Eq. (1). $P(z\,;\,\theta)$ is a prior distribution over latent variables z and $P(x|z;\,\theta)$ is the likelihood function or decoder that generates data x given latent variables z. Since the analytical estimation of the posterior distribution $P(z|x\,;\,\theta)$ is in general intractable, the generative model is trained using an approximate posterior distribution or an encoder $Q(z|x,\phi)$ which is such that the KullBack-Liebler divergence between $Q(z|x,\phi)$ and $P(z|x\,;\,\theta)$ is close to 0 and is a lower bound of loglikehoogd as shown in (4).

DOI: 10.31695/IJASRE.2024.1.4

$$log(P(x;\,\theta\,)) - D_{KL}(Q(z|x,\phi\,)||P(z|x\,;\,\theta\,)) = E_{z\sim q}\,[log(P(x|z\,;\,\theta\,))] - D_{KL}\,(Q(z|x,\phi\,)||P(z\,;\,\theta\,))$$
$$\leq log\big(P(x;\,\theta\,)\big) \qquad (4)$$

In [16], the authors showed in (5) that the correspondence between the prior distribution

$$Q(z\,;\,\phi) = E[Q(z|x,\phi)] \approx P(z\,;\,\theta) \qquad (5)$$

is quite difficult to achieve and influences the poor performance in image reconstruction on the part of VAE. One solution to this problem is to move to a discrete latent space as in [20] instead of the continuous space. In this case, the prior distribution $Q(z|x,\phi\,)$ is deterministic, and the prior distribution $P\big(z\,;\,\theta\,\big)$ is considered uniform. This makes the KullBack-Liebler divergence becomes the entropy of the latent distribution $H(z)$ and the encoder can be adjusted to maintain the Shannon equilibrium [7] and guarantee the throughput-distortion trade-off. In the case of the VQ-VAE model, the assignment of vectors to their cluster is done according to (6).

$$z_q\,(x) = e_k, where\ k = argmin_j\,\|z_e(x) - e_j\,\|^2\,. \qquad (6)$$

In this equation $z_e(x)$ corresponds to the latent vector coming from the encoder, the center e_k of the cluster k assigned to the latent vector $z_e(x)$, the Euclidean distance $\|.\|^2$. The vectors $e_i \in \mathbb{R}^D,\ i \in 1,2,\ldots,K,$ and the distribution $Q(z\ = k|x,\phi)$ is therefore according to (7).

$$Q(z\ = k|x,\phi) = \begin{cases} 1\ if\ k\ = argmin_j\,\|z_e(x)\ - e_j\,\|^2 \\ \qquad q0 \end{cases} \qquad (7)$$

The loss function to be minimized becomes according to equation Eq. (8)

$$\mathcal{L}_t\,(\,\theta\,;\,\phi,x\,) = \mathcal{L}_r\,(\hat{\ }x,\widehat{x}) + \beta\|z_e\,(x) - sg(e_k\,)\|^2 \qquad (8)$$

where $sg(\cdot)$ is the gradient from the decoder transmitted to the encoder. The first term represents the reconstruction error of the auto-encoder. The second term is used to regularize the encoder so that the quantization error is minimized. The parameter is β generally taken equal to 0.25.

As in [41, 42], we propose a hierarchical approach to vector quantization to increase the reconstruction capabilities of our auto-encoder. To do this, we partition the latent variable z into L groups such that $z = z_{1:L}\ = \{z_1, z_2, \ldots, z_L\,\}$. The prior distribution of the latent variable z becomes according to (9).

$$P(z\,;\,\theta) = P(z_1\,;\,\theta)\prod_{l<L} P(z_l\,|z_{c<l}\,;\,\theta) \qquad (9)$$

and $Q(z|x\,;\,\phi)$ becomes according to (10) to

$$Q(z|x\,;\,\phi) = (Q(z|z_1\,,x\,;\,\phi)\prod_{l<L} (Q(z_l\,|z_{c<l}\,,x;\,\phi) \qquad (10)$$

The estimate of the ELBO represented in (4) then becomes (11).

$$E_{z\sim q}\,[log(P(x|z\,;\,\theta))] - D_{KL}\,(Q(z|x,\phi)||P(z\,;\,\theta)) = E_{z\sim q}[log(P(x|\,z\,;\,\theta))] - D_{KL}\,(Q(z_1\,|x,\phi)||P(z_1\,;\,\theta))$$
$$-\sum_{l<L} D_{KL}\,(Q(z_l\,|z_{c<l},x,\phi)||P(z|z_{c<l}\,;\,\theta))$$
$$\leq log\big(P(x;\,\theta\,)\big) \qquad (11)$$

Assuming conditional a priori law $P(z_l\,|z_{c<l};\,\theta)$ uniform and $Q(z_l\,|z_{c<l},x;\,\phi)$ deterministic, we have that the loss function becomes according to (12).

$$\mathcal{L}_{T,L}(\theta\,;\,\phi,x\,) = \mathcal{L}_r\,(x.\hat{x}) + \lambda\,\sum_{l<L} H(Q(z_l\,|z_{c<l},x\,)\,)\,) \quad + \beta\sum_{l<L}\|z_{\,e,l}\,(x) \,-\, sg(e_k\,)\|^2 \qquad (12)$$

In equation Eq. (12) H represents the entropy of the distribution Q.

Inspired by the work of [5], [20] which shows that such a hierarchy is similar to a Gaussian and Markovian hierarchical model, at each resolution level $z_{e,l}\,(x)$ is estimated by a residual approach in (13).

$$z_{e,l}\,(x) = \;G\big(x, z_{e,l-1}\,;\,\phi\big) = H\big(x, z_{e,l-1}\,;\,\phi\big) - x \qquad (13)$$

with the function $H(.\,;\phi)$ is the function allowing to extract features at the layer l.

Likewise, for the calculation of the entropy $H(Q(z_l\,|z_{c<l},x\,)$ the distribution Q is defined according to the equation Eq. (14)

$$Q\big(z_{\,e,l}\,(x) = k\big|x, z_{\,c<l}\,\big) = \propto exp\left(\frac{< z_{e,l}\;(x)|e_{k,l} >}{\sqrt{d_l}}\right). \qquad (14)$$

In this equation d_l is the dimensionality of a codebook tensor at layer l ; $<.\,;.>$ is the scalar dot product. To improve the quality and fidelity of the reconstructed images, the loss function of our auto-encoder is expressed by the relation $\mathcal{L}_{T,L}(\theta\,;\,\phi,x\,)$ given in (15)

$$\mathcal{L}_{T,L}(\theta\,;\,\phi,x\,) = (1-SSIM)\,\times\left(\frac{1}{2\gamma^2}\mathcal{L}_r\,(x.\hat{x}) + \lambda\,\sum_{l<L} H(Q(z_l\,|z_{c<l},x\,)\,)\,) \quad + \sum_{l<L}\|z_{\,e,l}\,(x) \,-\, sg(e_k\,)\|^2\right).\#(15)$$

In [20], the authors showed the mean square error (MSE) constitutes a source of limitation of the performance of VAEs because its estimation requires the calculation of an average over a set of pixels. This is partly responsible for the blurring effect observed in VAEs. To improve the quality of the images, we replace this MSE loss with the perceptual loss [52]. This type of loss compares the predicted images to those of a pre-trained model, generally, VGG-16 [53] to adjust the parameters which guarantee convergence. It is weighted by a coefficient that depends on the SSIM score [54] to take into account variations in luminance, contrast and distortions present between the reconstructed images and the original images. The parameter γ ensures the balance between good quality of data representation and good quality of image reconstruction [55] and decreases regularly towards 0 starting from an initial value of 1. The parameter λ is choosing to be close to 0 as possible, because quantization vector algorithm is the limit when entropy term goes to 0 [19]. In the next section, we describe the process of assigning a tensor from the latent space to its cluster.

ii. Estimation of the quantized vector

Papers can be written in English, French, Spanish or ArabicIn the classic approach, the assignment of a vector of the latent space is done by calculating its distance with all the vectors of the dictionary. This model, although effective, cannot be applied to low-cost devices because of its computational cost and applies well to small-sized images. To circumvent this limitation and allow the use of vector quantization at larger spatial scales, we propose a reformulation of the cluster assignment problem using self-attention mechanisms. Thus, if we consider $Z_e\,(x)$ the features descriptors tensor of the latent space coming from the encoder, $,Z_e\,(x) \in \mathbb{R}^{h\times w\times D}$. Thus, we can see $Z_e\,(x) \in \mathbb{R}^{N\times D}, N = h\times w$. The objective of the problem amounts to finding a dictionary $E\; = \{e_k\,|k = 1,2,\dots,K\}$ with ordered vectors $e_k\; \in \mathbb{R}^{1\times D}$ and $K\; << N$ which represents the centers of the clusters to which the vectors $z_{e,ij}\;(x) \;\in\; Z_e\,(x)$ belong. We therefore embed the space $\mathbb{R}^{N\times D}$ into the smaller space $\mathbb{R}^{K\times D}$ by the matrix Q of

DOI: 10.31695/IJASRE.2024.1.4

$\mathbb{R}^{K\times N}$ estimated iteratively. The initial dictionary $E^{(0)}$ is constructed by estimating the average of the closures on a regular grid of dimensions $p \times q$ such that $K = \frac{h}{p} \times \frac{w}{q}$

Thus, at iteration *t*, we have according to (16).

$$Q\ (t) = softmax\left(\frac{\left(E^{(t-1)}\ \right)Z^{T}}{\sqrt{D}}\right). \qquad (16)$$

The dictionary vectors are updated by column normalization of the matrix Q which we will denote $\tilde{Q}$ according to (17). So at iteration *t*:

$$\tilde{E}^{(t)} = \left(\tilde{Q}^{(t)}\right)Z \qquad (17)$$

The computational complexity is proportional to $vNKD$ operations where v is the number of iterations. The authors of [56] showed that the efficiency of the algorithm is proportional to the size of the dictionary. An increase in size would therefore make this algorithm unusable for our IoT systems. To do this as in [55], the scalar dot product is performed on a neighbourhood of size 3×3 taken around the dictionary vector. This reduces the computational complexity to $v9ND$. In our experiments, we take $v = 3$ iterations so finally we obtain an algorithm whose computational complexity is similar to $27ND$ and only depends on the dimensions of the latent space at the output of the encoder. To overcome the problem of the size of the dictionary, we will, as in [56], use multi-head self-attention [55] to capture the long-term dependencies between the elements of the dictionary and thus limit ourselves to relatively small dictionary sizes. The equations (18), (19) and (20) make it possible to estimate the new dictionary

$$\bar{E}^{(t)}\ = \tilde{E}^{(t)} \times Att\left(\tilde{E}^{(t)}\right) \qquad (18)$$

with

$$Att\left(\tilde{E}^{(t)}\right) = MHA\left(\tilde{E}^{(t)}\right) \qquad (19)$$

Here $MHA(.)$ represents multi-headed self-attention. Subsequently, the new dictionary is expressed by (20).

$$E\ (t) = \alpha\, E^{(t-1)}\ + (1-\alpha)\bar{E}^{(t)} \qquad (20)$$

In equation Eq. (20), it should be noted that the parameter α is learned during the training phase. Once the dictionary is constructed, we need to assign each vector to the center of the cluster e_k. To do this, the association between the latent space $Z_e\ (x)$ and the dictionary E is constructed by (21) which gives

$$\hat{Q}\left(z_{e,j}\ (x) = k\middle|x,\phi\ \right) = \begin{cases} 1\ if\ k\ = argmax_{(r|e_r\in \mathcal{V}_{3\times 3}\ de\ e_j)}\ (abs(Qrj) \\ 0 \end{cases}. \qquad (21)$$

In this equation, $\mathcal{Q}$ is calculated according to the equation Eq. (17). Subsequently, the quantized features descriptors $\hat{Z}_q$ of latent space features descriptors $Z_e\ (x)$ is estimated according to (22)

$$\hat{Z}_q = \hat{Q}^T E \qquad (22)$$

In which each element of the latent space is represented by the center of the cluster to which it belongs.

b. Structures of encoder and decoder

The proposed model follows the classical architecture of variational auto-encoders and includes an encoder followed by a decoder as shown in Fig. 1.

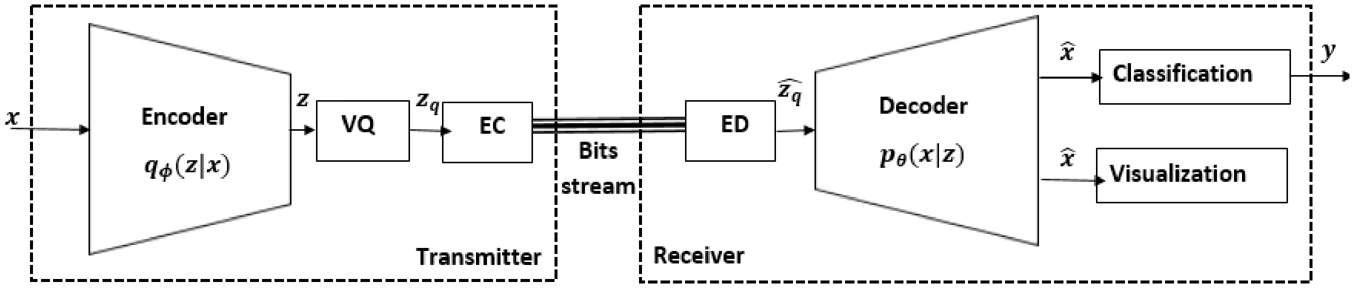


Fig. 1 Overall architecture of the variational auto-encoder

The encoder extracts feature from the coarsest spatial scale to the finest spatial scale, while the decoder reconstructs the image starting from the descriptors at the finest spatial scale until reconstructing the image whole. In its use in learning IoT networks, an arithmetic encoder, and decoder are placed respectively at the output of the encoder and the input of the decoder to protect the transmission of data from transmission channel errors. Our encoder, the structure of which is presented in Fig. 2, is composed of 2 residual blocks [56] to allow us to obtain characteristics with an important semantic context while avoiding gradient fading and avoiding a very deep architecture.

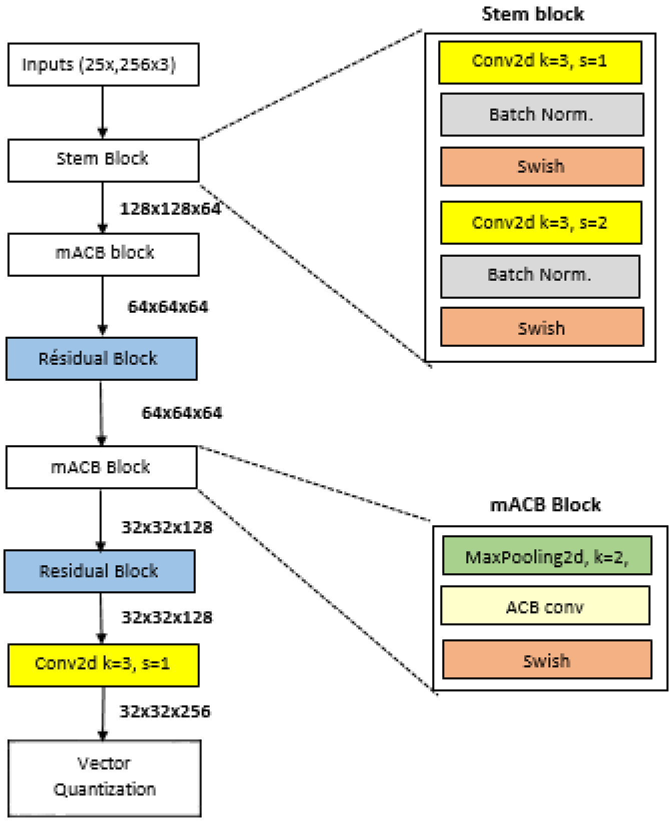


Fig. 2 Architecture of the encoder

The residual layers illustrated in Fig. 3 consist of a succession of two 2D convolution products with a 3 × 3 kernel followed by a Swish [57] type activation function. The Swish activation function helps improve network convergence during the training phase. In addition, each residual block is preceded by an asymmetric convolution of type ACB [58] which is composed of three 2D convolution products, one in the horizontal direction of kernel 1 × 3, the other in the vertical direction of kernel 3×1 and a convolution of square kernels of size 3×3. The ACB block makes it possible to increase the capacities of the reception field without having to increase the computational complexity. Thus, the encoder extracts the features with a local spatial context in a decreasing manner by decreasing each layer past its spatial resolution by half and increasing the dimensionality of the features by a factor of 2. This allows us to obtain a tensor of descriptors of size $\frac{H}{8} \times \frac{W}{8} \times 128$ at the output of the convolution blocks, with W, and H the spatial dimensions of the original images. The descriptor output from the encoder is then assigned to a cluster by vector quantization. Thus quantifying, the latent image representation space is discrete, and all of the semantic information contained in the image is thus represented by a set of K dictionary vectors of dimension D, with $K << HW$ where H, and W represent the width and height of the original image respectively.

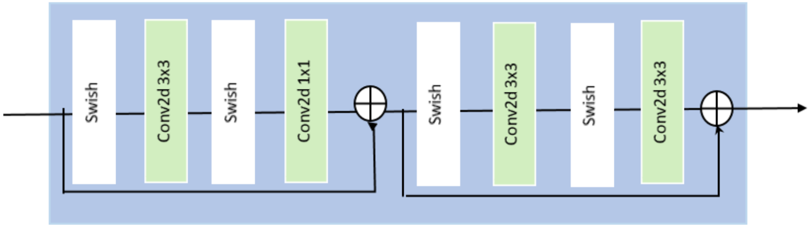


Fig. 3, Architecture of residual block used in our approach

The structure of the decoder is symmetrical to that of the encoder. Unlike the encoder, the transition from one layer to the next is done by a deconvolution product since we start from the deepest layers with a fine spatial resolution towards the highest spatial resolution layers while reducing the dimensionality of the descriptors. The size of the descriptors is multiplied by a factor of 2 with each passage from a lower layer to the next higher layer and the number of channels decreases by a factor of 2. The last layer of our model uses the activation function of type Elu [59] to maintain the values obtained in the interval [-1; 1] necessary for the estimation of the perceptual loss type loss function. In addition, our decoder uses the hierarchical vector quantization approach as indicated in the previous section. Figure 4 illustrate the working process of our proposed decoder.

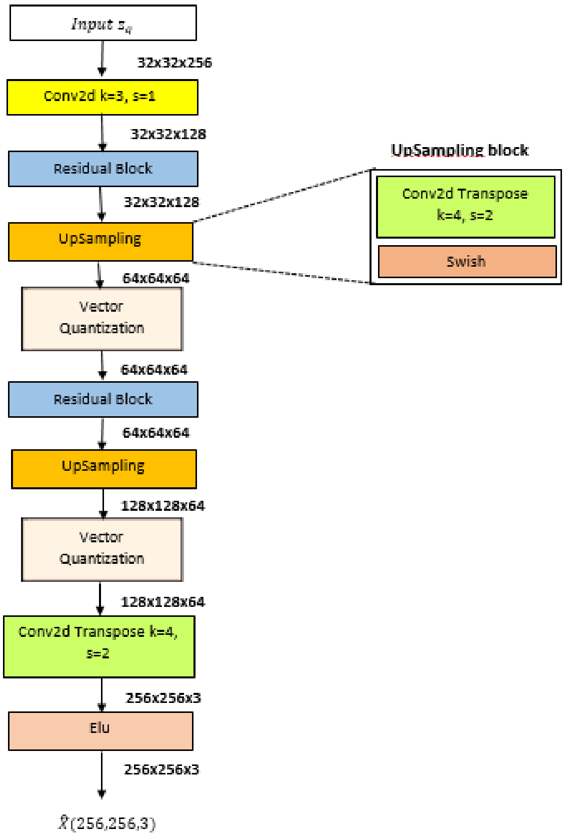


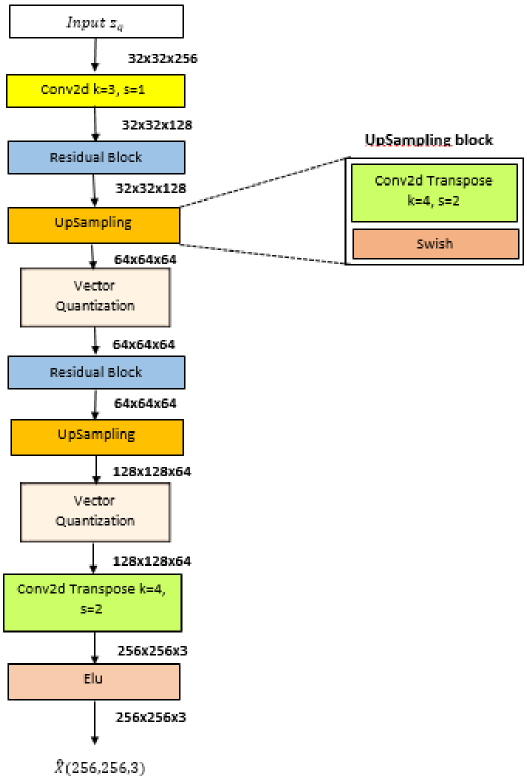
Fig. 4: Architecture of the decoder

4. Result and Discussion

4.1 Result

i. Experimental settings

The goal of our algorithm is to propose a framework capable of compressing and reconstructing images, we will carry out 2 series of experiments. The first concerns the compression capabilities of the encoder by analyzing the bitrate-distortion compromise. The second concerns the reconstruction and classification capabilities of the model. The experiments were conducted on a 16-core i7 processor PC with a NIVIDA RTX3070 GPU with 8GB of VRAM. During the training phase, the optimizer used is Adam type with an initial learning rate of 1×10^{-3}. The model input images were set to 256×256 pixels. Different types of datasets were used depending on the desired performance. The ImagNet100[60], celebA-HQ [61], Kodak [62] and Urban100 [63] datasets. The CelebA-HQ 256×256 dataset is a high-resolution version of the popular CelebA dataset, focused on celebrity faces for image generation and machine learning tasks. Consists of 30000 high-quality face images of celebrities. ImageNet100 is a smaller subset of the popular ImageNet dataset, designed for easier training and faster experimentation in image classification tasks. This dataset contains 100000 images across 100 object categories. Each category represents a distinct and easily recognizable object, like airplanes, dogs, flowers, or chairs. Images are of varying sizes and resolutions, reflecting real-world image diversity. The Urban100 dataset contains 100 images of urban scenes. It commonly used as a test set to evaluate the performance of super-resolution models. This images in this dataset are at different resolution from 256×256 pixels to 1024×1024 pixels. Kodak Dataset contains a collection of 24 high-quality, full-color images with 768×512 pixels of resolution. It iscommonly use for evaluating performance of compression algorithms and testing image quality metrics. Kaodak and Urban100 datasets were used in the first set of experiments as they are considered benchmarks reference when studying the flow-distortion trade-off. The compression qualities of our model will be studied according to the rate-distortion compromise. This trade-off is characterized by the evolution of the bitrate and MS-SSIM [64] or mean square error (MSE). Bitrate is the ratio between the number of bits necessary to encode the image on the size of the image. MS-SSIM and MSE represent the distortions caused by the compression of this data. The image generation capabilities of our algorithm will be evaluated based on the celebA-HQ and ImageNet100 datasets. The interest in these datasets comes from the fact that facial recognition is a particularly difficult task for generative models because of the local semantic content that varies greatly from one image to another.



DOI: 10.31695/IJASRE.2024.1.4

ii. Presentation of main results

To study the bitrate-distortion trade-off, we will simulate different compression rates of the Kodak and Urban100 datasets ranging in the corresponding interval Q = [0 ;100] varying from 0.1 to 0.99 to simulate low bitrate (bpp). This is justified by the fact that in practice, our devices will operate with very low bitrates (bpp). The results of the curves in Fig. 5 shows the evolution of the MS-SSIM and the MSE according to the different compression rates.

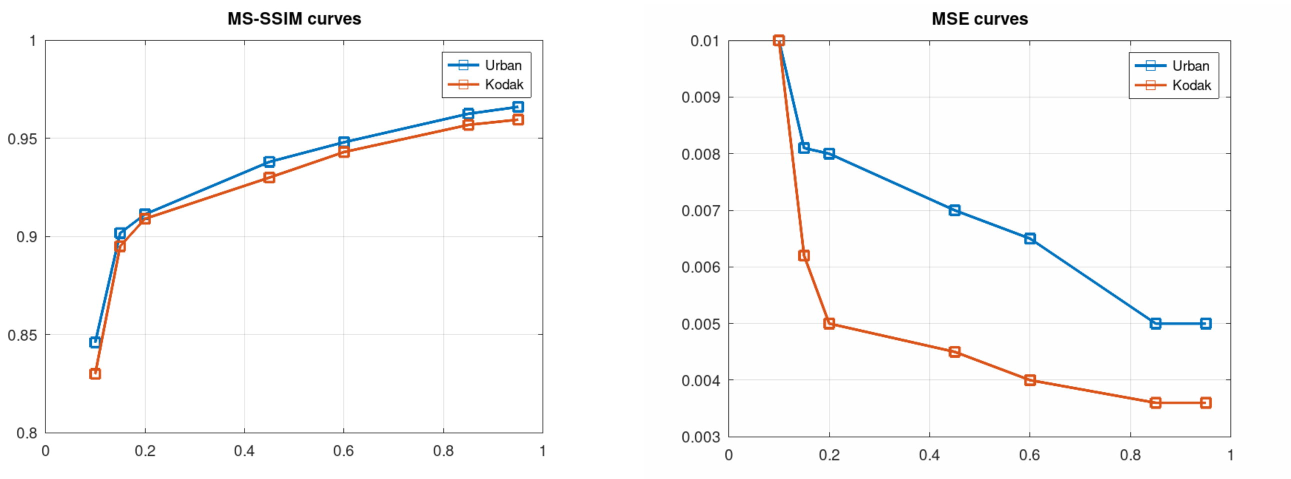


Fig. 5: MS-SSIM and MSE curves for Kodak and Ureban100 datasets

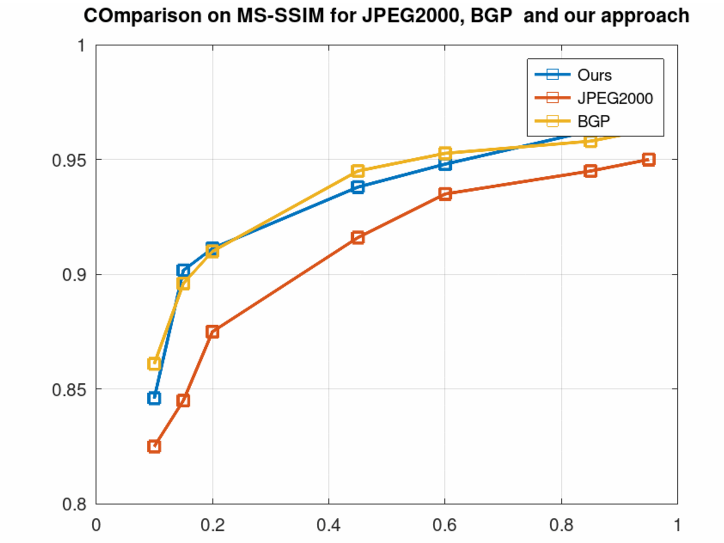


Fig. 6: Comparison of MS-SSIM between different compression algorithms: our model, JPEG2000 and BPG

In Fig. 6, we show the compression qualities of our algorithm by comparing the evolution of the MS-SSIM of our approach to those of classic compression algorithms such as JPEG 2000 and it's extension BPG. Fig. 7 and Fig. 8 show an example of image reconstruction from the celebA-HQ and ImageNet datasets for different compression rates. For each image, we give the PNSR, the MS-SSIM, and the rFID score [65]. The MS-SSIM describes the distortions present during the reconstruction of the images, the rFID which is an application of FID to the test data allows us to see the quality of resemblance of the reconstructed images with those of the original images. In the following figures, we can see the results of reconstructions for our approach for different low bitrates between 0 and 1. The choice of low bitrates is crucial in the case of IoT applications particularly when the environment is degraded. In this case, the bitrate from the encoder to the decoder becomes very low. To simulate this case, we compress each image in ImageNet and CelebA-HQ dataset with JPEG 2000 protocols to obtain different bitrates and we submit these compresses datasets to our model. The following figure shows the results after 10000 epochs. For the simulations, images are resized to 256×256 pixels in RGB color space

DOI: 10.31695/IJASRE.2024.1.4

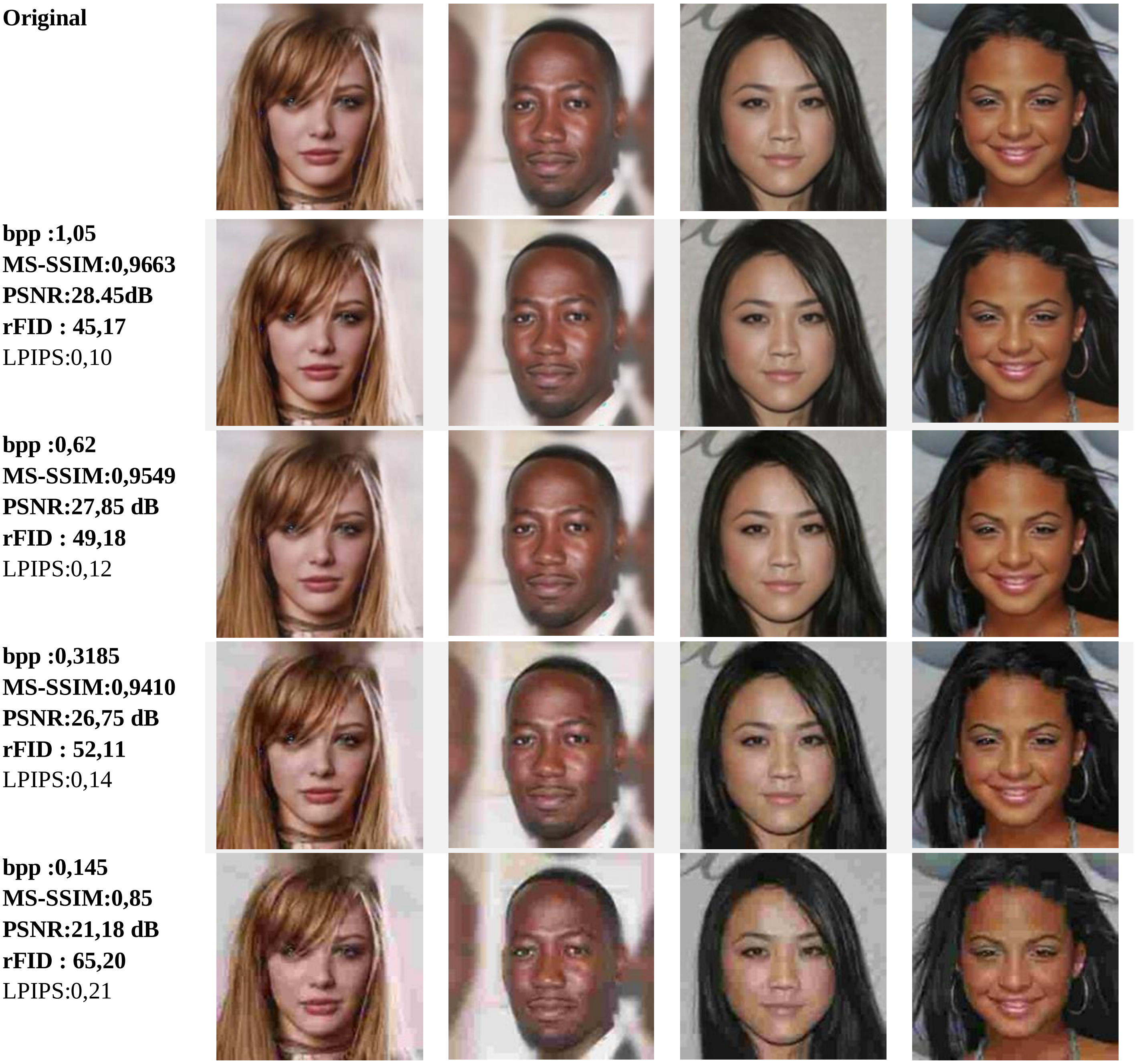
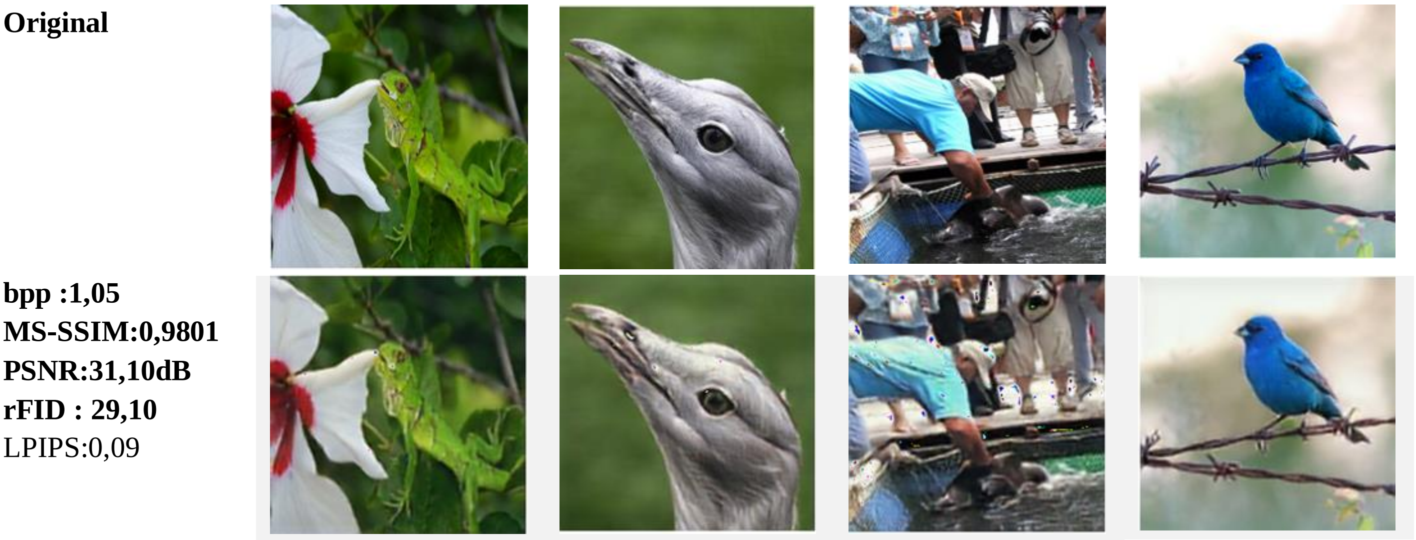


Fig. 7: Images reconstructions for different bitrates on CelebA-HQ 256 x256 dataset



DOI: 10.31695/IJASRE.2024.1.4

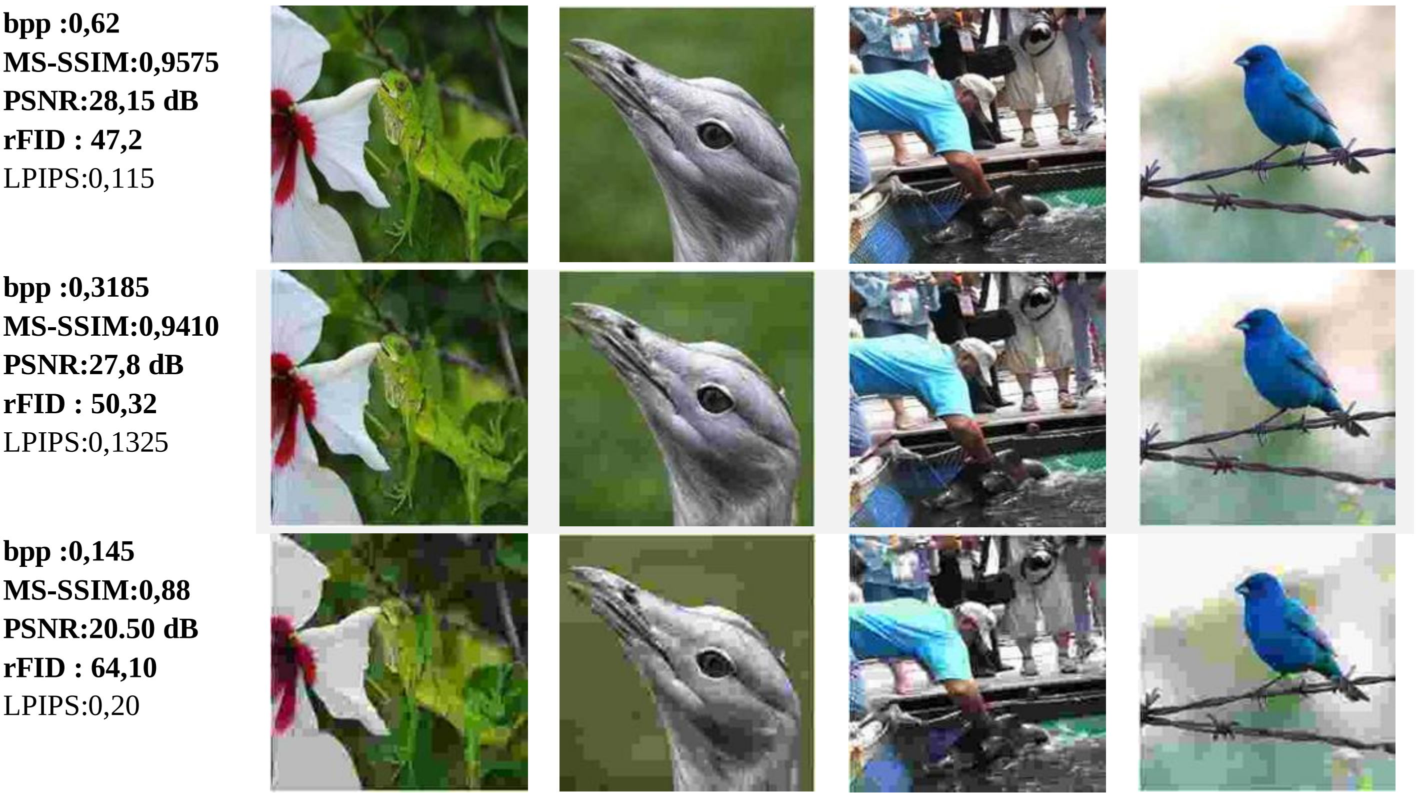


Fig. 8: Image reconstruction for Imagenet100 datasets for difeérents bitrate(bpp)

As we can see, our decoder preserves most information in the images despite the low bitrate. In most case PSNR > 20 dB. The metrics like PNSR, and MS-SSIM grows with the augmentation of the bitrate and shows that, when the bitrate increase, the distortions decrease. It   is due to complex structure of the decoder that propagate information from the finest lower layers to coarse higher layer using residual block at each step and vector quantization. We can notice in this figure that, rFID and LPIPS decrease when the bitrate increases. This behavior shows that, when the bitrate increases, our model is able to build images that are similar to the original images unlike the bit rate is still slow.

b.   Discussion

i.   Analysis of trade-odd debit-distortion

The curves in Figure 5 shows the evolution of the MS-SSIM, the P-SNR (dB), and the MSE as a function of the different compression rates and therefore the bitare (bpp). In distortion analysis, MS-SSIM analyzes the structural similarity between the compressed image and the original image at several levels of spatial resolution and translates the quality of preservation of local structures. The MS-SSIM evolution curve shows that for our Kodak and Urban100 image sets, it varies between 0.87 and 0.96 in an increasing manner depending on the evolution of the bitrate. This evolution shows that an increase in the compression quality increases the MS-SSIM and therefore consequently reduces the distortions in the images reconstructed at the decoder output. Moreover, the MS-SSIM score obtained in the worst cases is relatively good around 0.87 in Kodak and urban100. This reflects the fact that despite significant degradation, most of the local structures are preserved by our model. As for the curve reflecting the evolution of the MSE as a function of the bitrate, it decreases as a function of the evolution of the bitrate and therefore the quality of the compression. This curve reflects the overall reconstruction error of the compressed image compared to the original image. We see that despite the significant distortions introduced by the low compression qualities, the reconstruction error remains relatively low ranging from 0.01 for a bitrate of **0.1** bpp to **0.003** for a bitrate of 1. In addition, it should be noted that in the curves of Figure 5, the results in terms of MS-SSIM and MSE are better for the Kodak dataset than that of Urban100. This is explained by the complexity of the structures present in the Urban100 images which generally represent an urban environment with greater variability of objects in terms of size and local context than that of the Kodak dataset. In Figure 6, we compare the results obtained with those of other compression algorithms such as JPEG 2000 and BPG. The results show that our algorithm performs better than JPEG 2000 and presents similar results

DOI: 10.31695/IJASRE.2024.1.4

to that of BPG. Indeed, whatever the bitrate, the curve reflecting the evolution of the MS-SSIM of our approach is always above that of the JPEG2000 algorithm. Indeed, JPEG2000 suffers from implementation complexity accompanied by latencies in data coding and uses larger encoders to represent the information. Which limits its performance in coding with low compression quality, therefore requiring low bitrates. As for the BGP algorithm, the evolution of MS-SSIM is relatively similar for all compression qualities except in compression qualities close to 100 in which our algorithm becomes superior by +0.1% in terms of MS-SSIM. For low bitrates, BPG and our approach give fairly close results with MS-SSIM between 0.87 and 095 for our approach and 0.88 and 0.95 for BPG, for bitrates between 0.15 and 0.6. this reflects the fact that our algorithm has, just like BPG, the ability to encode most of the visual information into small-sized vectors.

ii. Analysis of image reconstruction quality

Figures 7 and 8 showed that the image reconstruction quality of our model increases with the quality of the compression of the input images. the distortions characterized by the MS-SSIM, and the PNSR decrease while the visual quality and resemblance to the original image increases by the rFID and LPIPS cores. for these 2 metrics, the more they decrease, the better the quality of the reconstruction. In order to evaluate the generation quality of our decoder, we will compare it to some of the vector quantization based variational auto-encoder.

Table 1. Comparative results of rFID score for differents algorithms on Celeb A-Hq dataset

Dataset	Methods	rFID↓
CelebA-HQ 256 x 256	**VQ-VAE**	85.9
	VQ- VAE+ HMA	45.6
	HQA	22.8
	Our approach	23.4

The previous table summarizes the rFID score of our approach and different auto-encoder algorithms based on vector quantization for the celebA-HQ. Our approach is compared to those of the HQA and VQ algorithms. VQ-VAE and VQ-VAE +HMA. The previous table shows that the VQ-VAE model presents the highest rFID score compared to other hierarchical approaches. Our approach in terms of rFiD outperforms the original VQ-VAE model by a reduction of 62.2 in the rFID score and a reduction of 22.2 compared to its extension by the hierarchical approach. This is explained by the fact that our approach judiciously combines the information from the lower layers with that from the higher layers while in this VQ-VAE + HMA approach, the output of a vector quantifier layer serves as input to the next layer without the combination of information. Our model provides an rFID score slightly lower than that of the HQA model despite less completeness for our model while the HQA approach based on an auto-regressive approach is relatively difficult to implement in IoT application situations.

5. CONCLUSION

In this paper, we proposed a variable auto-encoder architecture adaptable to IoT networks. These types of networks use devices with low memory and calculation capacity and low input consumption and deport the majority of the processing to the edge of the network. These networks therefore have the constraints of a low throughput at the output of the input device, which requires highly compressing the data and a large capacity for reconstruction of the peripheral equipment. Our algorithm based on vector quantization proposes a quantification approach based on the calculation of cosine similarities between the vectors of the latent space coming from the encoder and the data representation dictionary. It also uses a hierarchical vector quantization approach in the decoder. This made it possible to obtain an encoder capable of strongly compressing the data and offered the decoder reconstruction capabilities which surpass the most efficient algorithms in the literature with less computational complexity.

Future research will aim to introduce a classifier at the decoder level so that it can combine in a single architecture visualization and reconstruction of data in addition to having the capacity of their labelling.

DOI: 10.31695/IJASRE.2024.1.4

REFERENCES

[1] J. Kaur, M. A. Khan, M. Iftikhar, M. Imran, and Q. E. U. Haq, “Machine learning techniques for 5g and beyond,” IEEE Access, vol. 9, pp. 23472– 23488, 2021.

[2] B. Ji, Y. Wang, K. Song, C. Li, H. Wen, V. G. Menon, and S. Mumtaz, “A survey of computational intelligence for 6g: Key technologies, applications and trends,” IEEE Transactions on Industrial Informatics, vol. 17, no. 10, pp. 7145–7154, 2021.

[3] M. H. Miraz, M. Ali, P. S. Excell, and R. Picking, “A review on internet of things (iot), internet of everything (ioe) and internet of nano things (iont),” 2015 Internet Technologies and Applications (ITA), pp. 219–224, 2015.

[4] C. Gong, F. Lin, X. Gong, and Y. Lu, “Intelligent cooperative edge computing in internet of things,” IEEE Internet of Things Journal, vol. 7, no. 10, pp. 9372–9382, 2020.

[5] W. Williams, S. Ringer, T. Ash, D. MacLeod, J. Dougherty, and J. Hughes, “Hierarchical quantized auto-encoders,” Advances in Neural Information Processing Systems, vol. 33, pp. 4524–4535, 2020.

[6] A. Van Den Oord, O. Vinyals et al., “Neural discrete representation learning,” Advances in neural information processing systems, vol. 30, 2017.

[7] C. E. Shannon et al., “Coding theorems for a discrete source with a fidelity criterion,” IRE Nat. Conv. Rec, vol. 4, no. 142-163, p. 1, 1959.

[8] F. Mentzer, E. Agustsson, M. Tschannen, R. Timofte, and L. Van Gool, “Conditional probability models for deep image compression,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4394–4402.

[9] N. Johnston, E. Eban, A. Gordon, and J. Ballé, “Computationally efficient neural image compression,” arXiv preprint arXiv:1912.08771, 2019.

[10] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” arXiv preprint arXiv:1312.6114, 2013.

[11] J. Chen, Y. Ye, and S. Kim, “Jvet-n1002: Algorithm description for versatile video coding and test model 5 (vtm 5),” in Proceedings of the Joint Video Experts Team (JVET), 14th Meeting, Geneva, Switzerland, 2019, pp. 19–27.

[12] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, “Overview of the high efficiency video coding (hevc) standard,” IEEE Transactions on circuits and systems for video technology, vol. 22, no. 12, pp. 1649–1668, 2012.

[13] L. D. Chamain and Z. Ding, “Faster and accurate classification for jpeg2000 compressed images in networked applications,” arXiv preprint arXiv:1909.05638, 2019.

[14] “Improving deep learning classification of jpeg2000 images over bandlimited networks,” in ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020, pp. 4062–4066.

[15] L. D. Chamain, F. Racapé, J. Bégaint, A. Pushparaja, and S. Feltman, “End-to-end optimized image compression for machines, a study,” in 2021 Data Compression Conference (DCC). IEEE, 2021, pp. 163–172.

[16] E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, and L. V. Gool, “Soft-to-hard vector quantization for end-toend learning compressible representations,” Advances in neural information processing systems, vol. 30, 2017.

[17] J. Ball´e, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, “Variational image compression with a scale hyperprior,” arXiv preprint arXiv:1802.01436, 2018.

[18] L. D. Chamain, S. Qi, and Z. Ding, “End-to-end image classification and compression with variational auto-encoders,” IEEE Internet of Things Journal, vol. 9, no. 21, pp. 21916–21931, 2022.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” Advances in neural information processing systems, vol. 30, 2017.

[20] A. Asperti, D. Evangelista, and E. Loli Piccolomini, “A survey on variational auto-encoders from a green ai perspective,” SN Computer Science, vol. 2, no. 4, p. 301, 2021.

[21] I. E. Commission et al., “Information technology: digital compression and coding of continuous-tone still images.”

[22] D. S. Taubman, M. W. Marcellin, and M. Rabbani, “Jpeg2000: Image compression fundamentals, standards and practice,” Journal of Electronic Imaging, vol. 11, no. 2, pp. 286–287, 2002.

[23] B. Bross, J. Chen, S. Liu, and Y.-K. Wang, “Jvet-s2001 versatile video coding (draft 10),” Joint Video Exploration Team (JVET) of ITU-T SG, vol. 16, 2020.

[24] J. Ballé, P. A. Chou, D. Minnen, S. Singh, N. Johnston, E. Agustsson, S. J. Hwang, and G. Toderici, “Nonlinear transform coding,” IEEE Journal of Selected Topics in Signal Processing, vol. 15, no. 2, pp. 339–353, 2020.

[25] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, “Learned image compression with discretized gaussian mixture likelihoods and attention modules,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 7939–7948.

[26] D. Minnen and S. Singh, “Channel-wise autoregressive entropy models for learned image compression,” in 2020 IEEE International Conference on Image Processing (ICIP). IEEE, 2020, pp. 3339–3343.

[27] F. Mentzer, G. D. Toderici, M. Tschannen, and E. Agustsson, “High-fidelity generative image compression,” Advances in Neural Information Processing Systems, vol. 33, pp. 11913–11924, 2020.

[28] T. Chen, H. Liu, Z. Ma, Q. Shen, X. Cao, and Y. Wang, “End-to-end learnt image compression via non-local attention optimization and improved context modeling,” IEEE Transactions on Image Processing, vol. 30, pp. 3179– 3191, 2021.

[29] M. Lu, P. Guo, H. Shi, C. Cao, and Z. Ma, “Transformer-based image compression,” arXiv preprint arXiv:2111.06707, 2021.

[30] R. Zou, C. Song, and Z. Zhang, “The devil is in the details: Windowbased attention for image compression,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 17492– 17501.

[31] M. Lu, F. Chen, S. Pu, and Z. Ma, “High-efficiency lossy image coding through adaptive neighborhood information aggregation,” arXiv preprint arXiv:2204.11448, 2022.

[32] Y. Hu, W. Yang, Z. Ma, and J. Liu, “Learning end-to-end lossy image compression: A benchmark,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 8, pp. 4194–4211, 2021.

[33] D. Minnen, J. Ball´e, and G. D. Toderici, “Joint autoregressive and hierarchical priors for learned image compression,” Advances in neural information processing systems, vol. 31, 2018.

[34] D. He, Y. Zheng, B. Sun, Y. Wang, and H. Qin, “Checkerboard context model for efficient learned image compression,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 14771–14780.

DOI: 10.31695/IJASRE.2024.1.4

[35] H. Ma, D. Liu, N. Yan, H. Li, and F. Wu, "End-to-end optimized versatile image compression with wavelet-like transform," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 3, pp. 1247–1263, 2020.

[36] Y. Yang and S. Mandt, "Towards empirical sandwich bounds on the rate-distortion function," arXiv preprint arXiv:2111.12166, 2021.

[37] T. M. Cover, Elements of information theory. John Wiley & Sons, 1999.

[38] A. Asperti, "Variance loss in variational auto-encoders," in Machine Learning, Optimization, and Data Science: 6th International Conference, LOD 2020, Siena, Italy, July 19–23, 2020, Revised Selected Papers, Part I 6. Springer, 2020, pp. 297–308.

[39] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," in Advances in Neural Information Processing Systems, vol. 29, 2016.

[40] R. Child, "Very deep vaes generalize autoregressive models and can outperform them on images," in International Conference on Learning Representations, 2021.

[41] A. Vahdat and J. Kautz, "Nvae: A deep hierarchical variational auto-encoder," Advances in neural information processing systems, vol. 33, pp. 19667–19679, 2020.

[42] Z. Duan, M. Lu, J. Ma, Y. Huang, Z. Ma, and F. Zhu, "Qarv: Quantizationaware resnet vae for lossy image compression," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023.

[43] G. Flamich, M. Havasi, and J. M. Hernandez-Lobato, "Compressing images by encoding their latent representations with relative entropy coding," Advances in Neural Information Processing Systems, vol. 33, pp. 16131– 16141, 2020.

[44] E. Agustsson and L. Theis, "Universally quantized neural compression," Advances in neural information processing systems, vol. 33, pp. 12367– 12376, 2020.

[45] L. Theis and N. Y. Ahmed, "Algorithms for the communication of samples," in International Conference on Machine Learning. PMLR, 2022, pp. 21308–21328.

[46] G. Flamich, S. Markou, and J. M. Hernandez-Lobato, "Fast relative entropy coding with a* coding," in International Conference on Machine Learning. PMLR, 2022, pp. 6548–6577.

[47] A. Vahdat, E. Andriyash, and W. Macready, "Dvae#: Discrete variational auto-encoders with relaxed boltzmann priors," Advances in Neural Information Processing Systems, vol. 31, 2018.

[48] Y. Choi, M. El-Khamy, and J. Lee, "Variable rate deep image compression with a conditional auto-encoder," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 3146–3154.

[49] F. Yang, L. Herranz, Y. Cheng, and M. G. Mozerov, "Slimmable compressive auto-encoders for practical neural image compression," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 4998–5007.

[50] M. Song, J. Choi, and B. Han, "Variable-rate deep image compression through spatially-adaptive feature transform," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 2380–2389.

[51] C. Gao, T. Xu, D. He, Y. Wang, and H. Qin, "Flexible neural image compression via code editing," Advances in Neural Information Processing Systems, vol. 35, pp. 12184–12196, 2022.

[52] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," IEEE transactions on image processing, vol. 13, no. 4, pp. 600–612, 2004.

[53] A. Asperti and M. Trentin, “Balancing reconstruction error and kullbackleibler divergence in variational autoencoders,” IEEE Access, vol. 8, pp. 199440–199448, 2020.

[54] P. Esser, R. Rombach, and B. Ommer, “Taming transformers for highresolution image synthesis,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 12873–12883.

[55] V. Jampani, D. Sun, M.-Y. Liu, M.-H. Yang, and J. Kautz, “Superpixel sampling networks,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 352–368.

[56] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[57] P. Ramachandran, B. Zoph, and Q. V. Le, “Searching for activation functions,” arXiv preprint arXiv:1710.05941, 2017.

[58] X. Ding, Y. Guo, G. Ding, and J. Han, “Acnet: Strengthening the kernel skeletons for powerful cnn via asymmetric convolution blocks,” in Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 1911–1920.

[59] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, “Fast and accurate deep network learning by exponential linear units (elus),” arXiv preprint arXiv:1511.07289, 2015.

[60] J. Deng, “A large-scale hierarchical image database,” Proc. of IEEE Computer Vision and Pattern Recognition, 2009, 2009.

[61] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive growing of gans for improved quality, stability, and variation,” arXiv preprint arXiv:1710.10196, 2017.

[62] A. Yanagawa, A. C. Loui, J. Luo, S.-F. Chang, D. Ellis, W. Jiang, L. Kennedy, and K. Lee, “Kodak consumer video benchmark data set: concept definition and annotation,” Columbia University ADVENT Technical Report, pp. 246–2008, 2008.

[63] J.-B. Huang, A. Singh, and N. Ahuja, “Single image super-resolution from transformed self-exemplars,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 5197–5206.

[64] D. Severo, L. Theis, and J. Ballé, “The unreasonable effectiveness of linear prediction as a perceptual metric,” arXiv preprint arXiv:2310.05986, 2023.

[65] U. Albalawi, S. P. Mohanty and E. Kougianos, "A Hardware Architecture for Better Portable Graphics (BPG) Compression Encoder," 2015 IEEE International Symposium on Nanoelectronic and Information Systems, Indore, India, 2015, pp. 291-296, doi: 10.1109/iNIS.2015.12.

[66] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” Advances in neural information processing systems, vol. 30, 2017.