Efficient Screen Content Image Compression via Superpixel-based Content Aggregation and Dynamic Feature Fusion

Sheng Shen , **Huanjing Yue*** , **Jingyu Yang***
Tianjin University
{codyshens, huanjing.yue, yjy}@tju.edu.cn,

Abstract

This paper addresses the challenge of efficiently compressing screen content images (SCIs) – computer generated images with unique attributes such as large uniform regions, sharp edges, and limited color palettes, which pose difficulties for conventional compression algorithms. We propose a Superpixel-based Content Aggregation Block (SCAB) to aggregate local pixels into one superpixel and aggregate non-local information via super-pixel transformer. Such aggregation enables the dynamic assimilation of non-local information while maintaining manageable complexity. Furthermore, we enhance our channel-wise context entropy model with a Dynamic Feature Fusion (DFF) mechanism. This mechanism integrates decoded slices and side information dynamically based on their global correlation, allowing the network to dynamically learn the optimal weights for global information usage. Extensive experiments on three SCI datasets (SCID, CCT, and SIQAD) show our method's superior RD performance and inference time, making it the first network comparable with the advanced VVC-SCC standard.

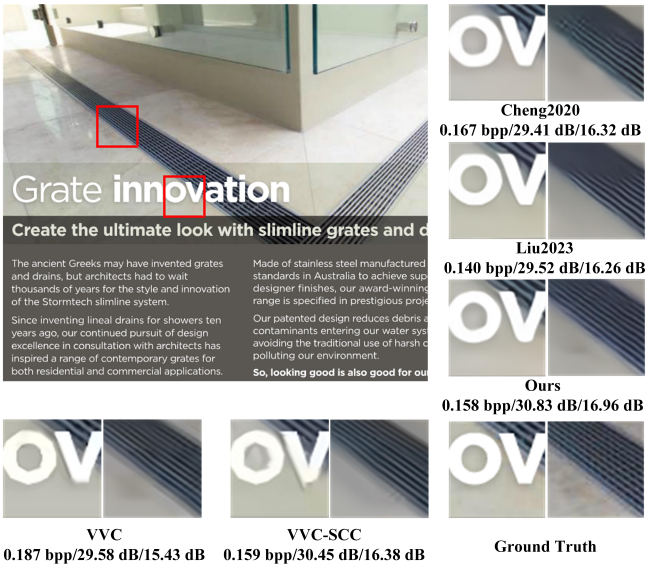

Figure 1: Decompressed images with various coding methods for the image 'siqad17' in SIQAD Dataset. We highlight the decoded patches in the red box area generated by different coding methods. For each method, its corresponding bit rate, Peak Signal-to-Noise Ratio (PSNR), and Multi-Scale Structural Similarity Index (MS-SSIM, in terms of dB) for the whole image is listed below the corresponding patch.

1 Introduction

With the surge in screen-content applications like online conferences and webcasting, efficient SCI compression is increasingly needed. Unlike natural images, SCIs contain large uniform regions, repetitive patterns, and limited color palettes, challenging traditional compression algorithms such as JPEG [Wallace, 1992] and JPEG2000 [Rabbani and Joshi, 2002]. The HEVC [Sullivan *et al.*, 2012] extension for screen content coding (HEVC-SCC)[Liu *et al.*, 2015] introduced innovative coding tools to address these challenges, later incorporated into the VVC standard [Bross *et al.*, 2021]. Meanwhile, The advent of deep neural networks has revolutionized visual signal compression and processing. End-to-end networks have made significant progress in natural image compression [Ballé *et al.*, 2016; Ballé *et al.*, 2018; Minnen *et al.*, 2018; Minnen and Singh, 2020; Cheng *et al.*, 2020; Zou *et al.*, 2022; Liu *et al.*, 2023], even surpassing the latest VVC standards. However, when applied to SCIs, the rate-distortion (RD) performance tends to deteriorate, as evidenced by our comparative experiments illustrated in Figs. 1, 7, and 9. Therefore, in this work, we focus on LIC based SCI compression.

To cope with the unique characteristics of SCIs, content dependent redundancy exploration and energy compaction methods are required. There are some end-to-end natural image coding networks exploring this approach by using window-based self-attention mechanisms. However, they can only capture the correlations among a small local region, which degrades their capabilities in using the non-local similarities. Meanwhile, the SCIs usually have many repetitive patterns and the pixels in a local region may be very similar due to the piece-wise constant properties. Therefore, effi-

*Corresponding author.

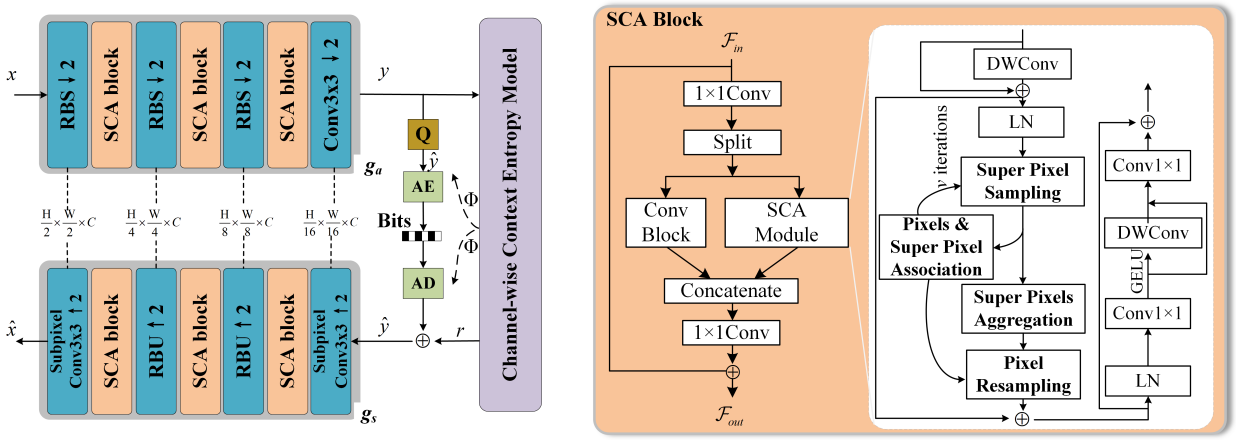

Figure 2: The overall framework of our method. RBS refers to the residual block that uses a convolutional layer with a stride of 2, while RBU denotes the residual block that employs a deconvolutional layer, also with a stride of 2.

cient non-local and local redundancy exploration is essential for SCIs. A straightforward solution for exploring non-local correlations is to employ global self-attention. However, this approach introduces heavy computational costs.

Based on the observations above, we introduce a superpixel based content aggregation strategy, motivated by the HEVC-SCC intra-block copy technique. Unlike the hard copy mode of intra-blocks, our method aggregates repetitive and local information through a soft copy approach (pixel aggregation). Similar pixels in a local region are first grouped into a superpixel to represent regional information, which significantly reduces the number of image primitives. Correspondingly, we propose a superpixel based feature aggregation mechanism. In this way, pixels within the superpixel can utilize global correlations. We then apply the proposed algorithm to enhance both the feature extraction network and the entropy model. The contributions of this paper can be summarized as follows:

1. We propose a Learned Image Compression (LIC) framework for efficiently compressing SCIs. A Superpixel-based Content Aggregation Block (SCAB) is proposed to aggregate regional information and explore non-local correlations via super-pixel based transformer. This mechanism enables our network to dynamically utilize non-local redundancy with small computing costs.
2. We introduce a Dynamic Feature Fusion (DFF) module into the channel-wise context entropy model. The DFF module dynamically combines decoded slices with side information based on their global correlation, which is realized by gating coefficients derived from super-pixel aggregation result. In this way, our entropy model can adaptively adjust the probability distribution of undecoded channels based on the dynamic fusion result.
3. Experiments demonstrate that our method outperforms six benchmarking LIC methods on three SCI datasets (i.e., SCID, CCT, and SIQAD) with different resolutions. In addition, our method is the first network that can be comparable with the advanced VVC-SCC standard.

2 Related Work

2.1 Traditional Screen Content Image Compression

Popular image/video coding standards like JPEG [Wallace, 1992], JPEG2000 [Rabbani and Joshi, 2002], HEVC [Sullivan *et al.*, 2012], and VVC [Bross *et al.*, 2021] are designed mainly for camera-captured images and videos. To address the unique features of screen content, research has built upon these standards for screen content coding. The HEVC-SCC [Liu *et al.*, 2015] extension, for instance, includes tools like intra block copy (IBC) [Xu *et al.*, 2016], adaptive color transform, and palette mode, enhancing the codec's adaptability to SCIs and improving compression efficiency. These tools, with modifications, are integrated into the VVC standard.

2.2 Learning-Based Image Compression

LIC has seen significant advancements. Ballé et al. [Ballé *et al.*, 2016] introduced an end-to-end CNN-based model, later enhanced with a VAE architecture and hyper-priors [Ballé *et al.*, 2018]. They also improved the entropy model using a local context model [Minnen *et al.*, 2018]. Guo et al. [Guo *et al.*, 2021] proposed a causal context model, while He et al. [He *et al.*, 2021] and Minnen et al. [Minnen and Singh, 2020] designed models for parallel computation. Various convolutional neural networks have been explored to boost image compression, such as the residual network by Cheng et al. [Cheng *et al.*, 2020] and the INNs by Xie et al. [Xie *et al.*, 2021]. However, these models struggle with SCIs due to the local receptive fields of convolutions. Recent LIC solutions have explored the Transformer architecture to capture long-range dependencies. Attention mechanisms have been integrated to enhance the transformation and entropy models of learned LICs. Several studies [Zou *et al.*, 2022; Zhu *et al.*, 2022; Liu *et al.*, 2023] have attempted to establish a Swin-Transformer-based LIC model.

LIC for Screen Content

LIC's application for SCIs is under-explored. Mitrica et al. [Mitrica *et al.*, 2019] proposed a semantic compression

scheme for airplane cockpit screen content. Tang et al. [Tang *et al.*, 2022] extended this with a text semantic-aware scheme (TSA-SCC) for ultra low bitrate settings. A learned image codec integrating transform skip was also proposed to enhance SCI compression [Wang *et al.*, 2022], but lacks comparison with VVC and public dataset validation. A multi-task learning architecture was presented for SCI encoding performance improvement [Zamanshoar Heris and Bajić, 2023], but it increases model complexity. This paper addresses these challenges by leveraging repetitive patterns in SCIs. We propose a super-pixel attention-based end-to-end encoding compression scheme, tailored for SCIs, outperforming the conventional encoding scheme VVC on multiple public datasets.

3 Method

3.1 Problem Formulation

Figure 2 presents our proposed network architecture. We adopt the VAE structure, a classic architecture in the deep image compression framework. It forms the foundation of both our primary and hyper encoder-decoder pairs. The primary encoder-decoder pair processes the raw input image x and the reconstructed image $\hat{x}$[1], while the hyper encoder-decoder pair models the spatial dependencies among the elements of the latent representation y. The entire framework can be summarized as follows:

$$\begin{aligned} y &= g_a(x;\phi) \\ \hat{y} &= Q(y-\mu)+\mu \\ \hat{x} &= g_s(\hat{y};\theta), \end{aligned} \tag{1}$$

where the analysis function g_a with parameters ϕ transforms x into y, which is then quantized by Q into $\hat{y}$. We adopt a channel-wise entropy model [Minnen and Singh, 2020; Zou *et al.*, 2022; Liu *et al.*, 2023] that computes $\Phi = (\mu, \sigma)$ for each channel of y, as shown in Equation 2 and Figure 2. The entropy model partitions y into s evenly slices $y_0, y_1, ..., y_{s-1}$ and enhances the encoding efficiency by exploiting the dependencies among the slices. This can be expressed as:

$$\begin{aligned} z &= h_a(y;\phi_h) \\ \hat{y} &= Q(z) \\ F_{\text{mean}}, F_{\text{scale}} &= h_s(\hat{z},\theta_h) \\ r_i, \Phi_i &= C_i(F_{\text{mean}}, F_{\text{scale}}, \overline{y}_{<i}, y_i), 0 \le i < s \\ \overline{y}_i &= r_i + \hat{y}_i, \end{aligned} \tag{2}$$

where the hyper-prior analysis h_a with parameters ϕ_h obtains side information z to capture the spatial dependencies among the elements of y. A factorized density model ψ is used to encode quantized $\hat{z}$ as $p_{\hat{z}|\psi}(\hat{z}|\psi)$ follow the prior work [Ballé *et al.*, 2018; Minnen *et al.*, 2018], as shown in Equation 3. Then, the hyper-prior synthesizer h_s, with parameters θ_h, decodes $\hat{z}$ to produce two latent features, F_{mean} and F_{scale}, which are subsequently fed into each slice network C_i. Each slice y_i is processed sequentially to generate $\hat{y}$. During this operation, the decoded slices $y_{<i} = \{y_0, y_1, ..., y_{i-2}, y_{i-1}\}$ and

[1]The encoder is commonly referred to as the analysis transform, playing the same role as prediction and transform in the traditional coding process.

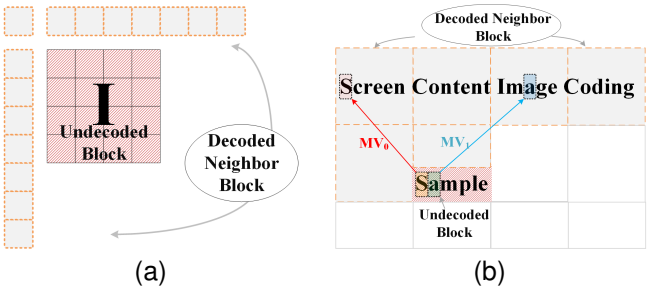

Figure 3: Illustration of prediction and transform in traditional compression process. (a) The hybrid intra prediction and transform in diverse Intra Profiles, such as HEVC, VVC, etc., where I signifies the block to be compressed. Various prediction modes facilitate leveraging reconstructed neighbors to form prediction I_m^p (see Equation 4 for details). (b) The most common prediction mode utilized in SCIs, *i.e,* the intra block copy (IBC) mode, exemplifying how I can be decomposed and reconstructed using this method.

the current slice y_i are input to the slice network C_i to estimate the distribution parameters $\Phi_i = (\mu_i, \sigma_i)$, which are then used to generate bit-streams. As such, we can posit $p_{\hat{y}|\hat{z}}(\hat{y}|\hat{z}) \sim N(\mu, \sigma^2)$. Furthermore, the residual r is employed to mitigate the quantization errors $(y - \hat{y})$ introduced by quantization. Finally, the synthesizer g_s with parameters θ reconstructs $\hat{x}$ from $\hat{y}$. Figure 5 provides a clear illustration of the detailed process of this channel-wise entropy model. The loss function is defined as follows:

$$\begin{aligned} \mathcal{L} &= R + \lambda D \\ &= R(\hat{y}) + R(\hat{z}) + \lambda D(x,\hat{x}) \\ &= E\left[-\log_2(p_{\hat{y}|\hat{z}}(\hat{y}|\hat{z}))\right] + E\left[-\log_2(p_{\hat{z}|\psi}(\hat{z}|\psi))\right] \\ &\quad + \lambda D(x,\hat{x}), \end{aligned} \tag{3}$$

where the trade-off between the rate and distortion in our compression approach is modulated by a Lagrange multiplier λ. The average number of bits needed to encode the input data is represented by R, and distortion, denoted by D, can be quantified using metrics such as MSE or MS-SSIM. Our objective is to minimize the loss function defined as Equation 3.

3.2 Content-Adaptive Transform via Stacked SCABs

We implement the content adaptive analysis transform by stacking Superpixel-Context Adaptive Blocks (SCABs), as depicted in Figure 2. Each SCAB unit comprises a superpixel transformer layer and a parallel convolutional residual layer.

Motivation

Traditional coding algorithms such as HEVC and VVC exhibit outstanding performance. These algorithms combine intra prediction and transform coding, effectively leveraging reconstructed neighboring regions to adaptively capture the dynamic characteristics of the content, thereby efficiently handling the complexity and variability of the image. The process can be simply formulated as:

$$
\begin{aligned}
I_m^p &= W_m N_{\text{rec}} \\
\hat{r}_m &= Q(g_a(I - I_m^p)); g_a = DCT(\cdot) \\
\overline{r}_m &= g_s(\hat{r}_m); g_s = IDCT(\cdot) \\
\hat{I} &= I_m^p + \overline{r}_m,
\end{aligned} \tag{4}
$$

where g_a and g_s denote the DCT [Ahmed *et al.*, 1974] and its inverse IDCT, respectively. N_{rec} represents the reconstructed neighbors. Each mode m has a specific W_m to weight the reconstructed N_{rec} to form the prediction I_m^p. A special case occurs when the IBC mode (specifically designed for screen content in HEVC-SCC and VVC-SCC) is applied: W_m becomes a one-hot mask, copying the same block as shown in Figure 3b. The content-adaptive weighting of reconstructed neighbors allows us to exploit screen content redundancy. Many CNN-based compression networks [Ballé *et al.*, 2016; Ballé *et al.*, 2018; Ballé *et al.*, 2020] may not support this feature due to fixed weights after training. Some studies [Cheng *et al.*, 2020; Zou *et al.*, 2022; Liu *et al.*, 2023] use window attention mechanisms for more dynamic and adaptive learning of neighborhood content, but these mechanisms have a limited dynamic range and integrate local features based on each pixel. A better method is to use larger pixel sets to perceive features, as shown in Figure 3b, where IBC performs content-awareness based on blocks. The symbol "s" finds its most similar block through block matching. Therefore, if the pixels representing "s" can form a superpixel, we can better perceive the surrounding information. Based on this, we propose a superpixel-based content aggregation for screen content compression.

Superpixel Based Aggregation Module

As shown in Figure 2, the Superpixel-based Content Aggregation (SCA) module consists of three processes: superpixel sampling, content aggregation in the superpixel space, and pixel resampling. Specifically, we first aggregate pixels into superpixels by using a superpixel sampling algorithm, then perform attention models in the superpixel space to handle global dependencies, and finally map the super pixels back to visual pixels by using a pixel resampling algorithm.

Superpixel Sampling: In the Superpixel Sampling process, we use the soft k-means based superpixel algorithm from SSN [Jampani *et al.*, 2018] to cluster the features $F \in \mathbb{R}^{B\times N\times C}$ (where $N = H \times W$ is the number of pixels) into m super pixels $S \in \mathbb{R}^{B\times m\times C}$. Each pixel feature $F_i \in \mathbb{R}^{B\times 1\times C}$ is assigned to one super pixel, and the assignment map is denoted by $\mathcal{M} \in \mathbb{R}^{N\times m}$. Formally, superpixel aggregation follows an Expectation-Maximization-like process, consisting of a total of T iterations. We initialize the super pixels S_0 by averaging the features in regular grid regions. The grid size is $h \times w$, and the number of super pixels is $m = \frac{H}{h} \times \frac{W}{w}$. Then we iteratively update the super pixels and the association with the following two steps:

1. Pixel-superpixel association: The pixel-superpixel association at iteration t is computed as:

$$
\mathcal{M}_{ij}^t = \text{Softmax}\left(\frac{F_i S_j^{t-1\text{T}}}{\sqrt{d}}\right), \tag{5}
$$

where i denotes the pixels, j denotes the superpixels. d is the channel number. It is noteworthy that the superpixel aggregation calculates the association between each pixel and only its 9 surrounding superpixels (as shown in Figure 4), ensuring the locality of superpixels, which also renders the computation and memory more efficient [Huang *et al.*, 2022].

Figure 4: Visualization of super-pixels. Left is the initialized super-pixels and right is the final learned super-pixels (for example, the number '9' is grouped into one super-pixel). For each pixel in the green box, we get its corresponding super-pixel association by only considering the surrounding super pixels within the red box.

2. Superpixels cluster centers updates: The superpixels are updated as the weighted sum of pixel features, defined as:

$$
S_j^t = \frac{1}{Z_j^t}\sum_i (\mathcal{M}_{ij}^t)^{\text{T}} F_i, \tag{6}
$$

where $Z_j^t = \sum_i \mathcal{M}_{ij}^t$ is the normalization factor. After T iterations, the final superpixel clusters S^T and the associated mapping $\mathcal{M}^T$ are obtained.

Content Aggregation in the Superpixel Space: Given that superpixels serve as compact representations of visual content, we employ attention mechanisms to emphasize their significance, enabling us to prioritize global contextual dependencies over local features. Specifically, we apply the standard self-attention technique to the sampled superpixels $S \in \mathbb{R}^{m\times C}$, which are defined as follows:

$$
\text{Attn}(S) = \text{Softmax}\left(\frac{q(S)k^{\text{T}}(S)}{\sqrt{d}}\right) v(S), \tag{7}
$$

where $q(\cdot), k(\cdot), v(\cdot)$ are linear functions, respectively. We omit the multi-head mechanism for clarity.

Pixel Resampling: After aggregating information in the superpixel space, the pixels composing the superpixels also receive global information flow. Subsequently, we map them back to pixels using the association map $\mathcal{M}_t$ and incorporate them into the original input F.

$$
\text{Resampling}(Attn(S)) = \mathcal{M}^t Attn(S). \tag{8}
$$

3.3 Superpixel Based Aggregation Block

We introduce an innovative transformer-style block that significantly enhances the screen image encoding process. This block, inspired by but diverging from the design in [Huang *et al.*, 2022], consists of three core components: Convolutional Position Embedding (CPE), Superpixel based Content Aggregation (SCA), and Convolutional Feed-Forward-Network (ConvFFN). These components are intricately designed to optimize the transformation of model features, F_{trans}, in the following manner:

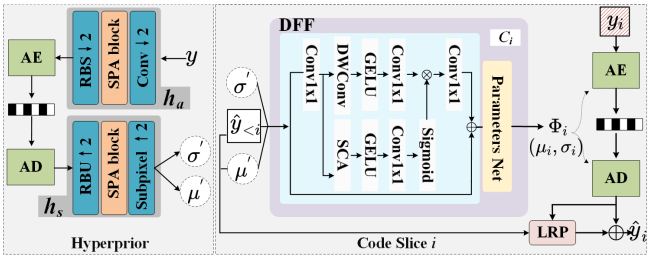

Figure 5: The proposed channel-wise context entropy model. The first slice (y_1) is compressed using a Gaussian entropy model conditioned solely on the hyperprior, while the entropy model for the rest slice y_i is conditioned on both the hyperprior and the decoded symbols in the former slices $\hat{y}_{<i}$. The structures of the Latent Residual Prediction Layer (LRP) and the Parameters Network are identical, consisting of "Conv3x3-GELU-Conv3x3-GELU-Conv3x3".

$$
\begin{aligned}
F_{trans} &= \text{CPE}\left(F_{trans}\right) + F_{trans} \\
F_{trans} &= \text{SCA}(\text{LN}(F_{trans})) + F_{trans} \\
F_{trans} &= \text{ConvFFN}(\text{BN}(F_{trans})) + F_{trans},
\end{aligned} \tag{9}
$$

where the CPE utilizes a 3×3 depth-wise convolution to provide positional information. The SCA is tailored to screen images, leveraging their characteristic structures to improve global context representation by efficiently capturing long-range dependencies. The ConvFFN, comprising two 1×1 convolutions, one 3×3 depth-wise convolution, and a GELU non-linear function, is specifically optimized for strengthening local feature representations in screen images.

Recognizing the importance of both global and local information in screen image encoding, we complement our transformer with a parallel convolution residual layer, uniquely designed for capturing fine-grained local details often present in screen content. This dual approach allows our model to effectively balance between global and local feature extraction. The entire process, incorporating this local learning aspect, is formulated as:

$$
\begin{aligned}
F_{\text{cnn}}\,, F_{\text{trans}} &= \text{Split}(\text{Conv1} \times 1(F_{in})) \\
F_{\text{cnn}}\,, F_{\text{trans}} &= \text{Res}\left(F_{\text{cnn}}\right), \text{Trans}\left(F_{\text{trans}}\right) \\
F_{\text{out}} &= F_{in} + \text{Conv1} \times 1\left(\text{Cat}\left(F_{\text{cnn}}\,, F_{\text{trans}}\right)\right).
\end{aligned} \tag{10}
$$

The complete architecture of our Screen Content Aggregation Block (SCAB) is depicted in Figure 2, illustrating how it uniquely addresses the challenges of screen image compression.

3.4 Advanced Entropy Coding with Channel-wise Context Modeling and Dynamic Feature Fusion

The most popular context modeling is proposed by Minnen [Minnen *et al.*, 2018], but it comes with increased decoding complexity. To address this, [Minnen and Singh, 2020] proposes a context model along the channel dimension, improving decoding efficiency. The latent tensor y is divided into N slices along the channel dimension with entropy parameters for each slice conditioned on the previously decoded ones. However, this approach lacks dynamic integration of side information from each slice and its decoded counterpart for entropy model parameter estimation, an aspect that could be further improved for more efficient screen content image encoding.

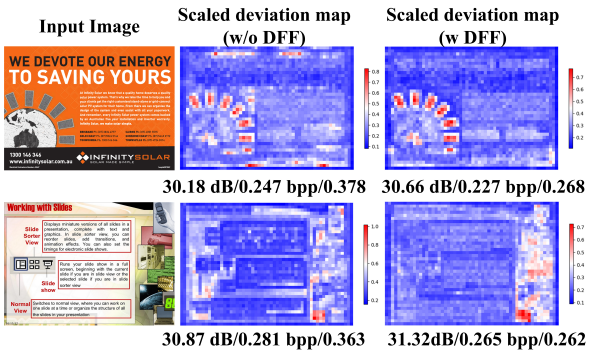

Figure 6: The deviation maps scaled by the model without (left) and with (right) DFF. For image 'siqad16' (top row), the model without DFF has a PSNR of 30.18 dB, a bitrate of 0.247 bpp, and a scaled deviation of $\tilde{\epsilon} = 0.378$; with DFF, it has a PSNR of 30.66 dB, a bitrate of 0.227 bpp, and a scaled deviation of $\tilde{\epsilon} = 0.268$. For the image 'siqad20' (bottom row), without DFF, the model has a PSNR of 30.87 dB, a bitrate of 0.281 bpp, and a scaled deviation of $\tilde{\epsilon} = 0.363$; with DFF, it has a PSNR of 31.32 dB, a bitrate of 0.265 bpp, and a scaled deviation of $\tilde{\epsilon} = 0.262$.

To address this issue, we introduce a dynamic feature fusion mechanism (DFF) combined with superpixel aggregation. As illustrated in Figure 5, we first compress the concatenated features using a 1x1 convolution. A gating mechanism then dynamically determines weights to enhance the entropy parameter estimation process by adapting to the decoded slices and side information. Simultaneously, the embedded Superpixel Context Aggregation (SCA) module allows the learning process to incorporate global information. Finally, the learned weights are multiplied back into the input branch, followed by another convolution to map the channels back to their original size.

To validate that our proposed DFF introduces minimal deviation, we follow the approach by Xie *et al.* [Xie *et al.*, 2021]. This involves analyzing pixel-wise differences between the compressed $\hat{y}$ and the original latent y, and measuring the information loss during compression by evaluating the deviation between $\hat{y}$ and y. The mean absolute pixel deviation ϵ is defined as:

$$
\epsilon = \frac{1}{H \times W} \sum_{h=1}^{H} \sum_{w=1}^{W} \left| y_{c,h,w} - \hat{y}_{c,h,w} \right|. \tag{11}
$$

We further introduce a scaling factor γ to normalize the deviation, accounting for different pixel value ranges in different models:

$$
\tilde{\epsilon} = \frac{\epsilon}{\gamma} = \frac{\frac{1}{H \times W} \sum_{h=1}^{H} \sum_{w=1}^{W} \left| y_{c,h,w} - \hat{y}_{c,h,w} \right|}{\frac{1}{H \times W} \sum_{h=1}^{H} \sum_{w=1}^{W} \left| y_{c,h,w} \right|}. \tag{12}
$$

Figure 6 illustrates the scaled deviation map of y for 'siqad16' and 'siqad20' from the SIQAD dataset, using models both with and without DFF. The result indicates that the model with DFF incurs less information loss, thus producing higher-quality decompressed images.

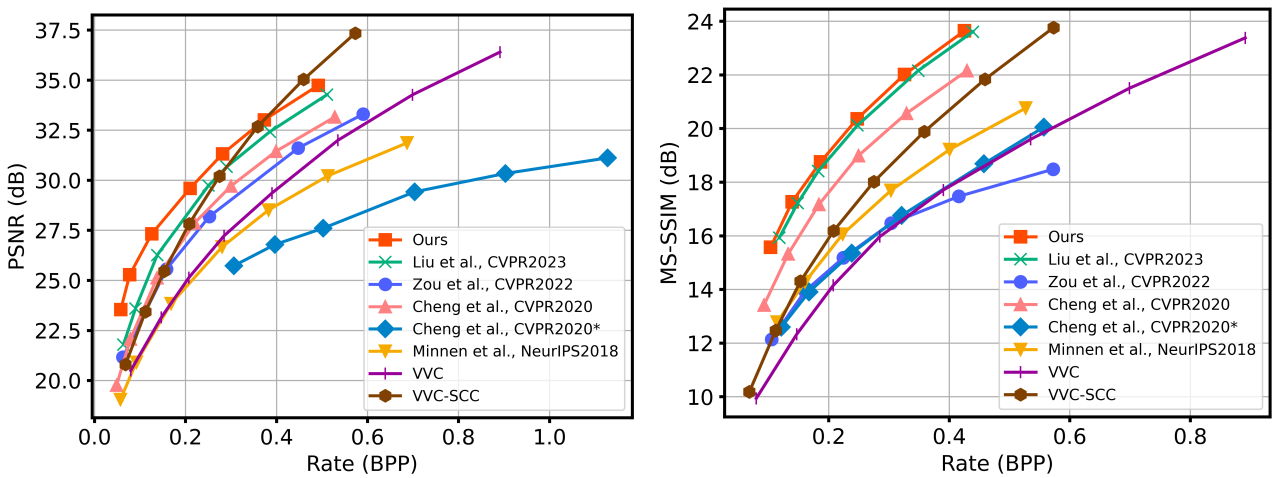

Figure 7: Performance evaluation on the SIQAD dataset. Comparative results of various models. Note that "cheng et al. CVPR 2020*" denotes the use of models pre-trained on natural images, while all other models represent the results of fine-tuning on the SCI2K dataset.

4 Experimental Evaluation

4.1 Experimental Setup

Training Settings

We select the SCI2K [Shen *et al.*, 2022] as our training dataset, where image samples are randomly cropped into $256 \times 256 \times 3$ patches. The network is optimized with the Adam optimizer, a batch size of 8, and trained on a single RTX 3090 GPU for $5M$ steps. The initial learning rate is 10^{-4}, reduced to 10^{-5} after $4.5M$ steps.

Model Settings

Our network is implemented using the open-source CompressAI PyTorch library [Bégaint *et al.*, 2020]. Six models are trained from scratch to match different bitrates by adjusting λ. The number of channels, M, for the latent variable y is set to 320, and for z it's 192. These settings balance computing cost and model performance. The number of slices s of the entropy model is set to 5. Additional hyperparameters within the entropy model align with the settings described in [Minnen and Singh, 2020]. For the distortion measurement using MSE loss, λ^2 is chosen from the set $\{0.00013, 0.00025, 0.00067, 0.0018, 0.0035, 0.0067, 0.013\}$. They were chosen based on preliminary experiments that yield good balance between compression rate and quality. For the MS-SSIM loss, λ is chosen from the set $\{2.4, 4.58, 8.73, 16.64, 31.73, 60.5\}$.

Testing

During testing, we use three common datasets for SCI quality estimation: CCT[3], SCID[4], and SIQAD[5]. These representative SCI datasets are also used in the SCI coding quality assessment research community[Min *et al.*, 2021]. The CCT dataset contains samples with a resolution of 915×1627, the SCID dataset provides images with a resolution of 720×1280, and the SIQAD dataset contains 24 screen images with 2k spatial resolution approximately. We use both PSNR and MS-SSIM to quantify the quality of decoded images, and bpp to measure the compressed bitrate.

[2]In the implementation, the actual loss function is given by $\lambda \times 255^2 \times D + R$, which is specifically tailored for optimization in the context of Mean Squared Error (MSE). When optimizing with the Multi-Scale Structural Similarity Measure (MS-SSIM), as suggested in [Bégaint *et al.*, 2020], the actual loss function is modified to $\lambda \times (1 - D) + R$.

[3]https://sites.google.com/site/minxiongkuo/uca

[4]http://smartviplab.org/pubilcations/SCID.html

[5]http://smartviplab.org/pubilcations/SCI_GSS.html

4.2 Evaluation on Rate-Distortion Performance

	CCT	SCID	SIQAD	Average
VVC	0.00	0.00	0.00	0.00
VVC-SCC	-43.01	-34.12	-33.57	-36.90
Minnen et al.	94.26	33.71	8.21	45.39
Cheng et al.	40.38	-7.02	-28.73	1.54
Zou et al.	48.07	-2.46	-24.25	7.12
Liu et al.	16.83	-22.04	-39.36	-14.86
Ours	**-17.00**	**-35.94**	**-51.04**	**-34.66**

Table 1: Averaged BD-rate saving against the VVC anchor on different datasets. Lower values are better.

Anchor and Comparative Methods

We compare our model with several notable LIC solutions, such as Minnen et al. (2018) [Minnen *et al.*, 2018], Cheng et al. (2020) [Cheng *et al.*, 2020], Zou et al. (2022) [Zou *et al.*, 2022], and Liu et al. (2023) [Liu *et al.*, 2023]. We use VVC Intra, with its reference software VTM-20.0, as the anchor for calculating BD-rate gains. We also present results of VVC-SCC Intra with screen content coding options. R-D curves are plotted in Figures 7 and BD-rate gains against the VVC anchor are in Table 1. For fairness, all compared methods are finetuned on the SCI2K dataset for 300 epochs based on their best pretrained models.

Quantitative Performance

As for the results on SIQAD dataset, our proposed method outperforms all other LIC solutions for the distortion measured by both PSNR and MS-SSIM, as shown in Figure 7.

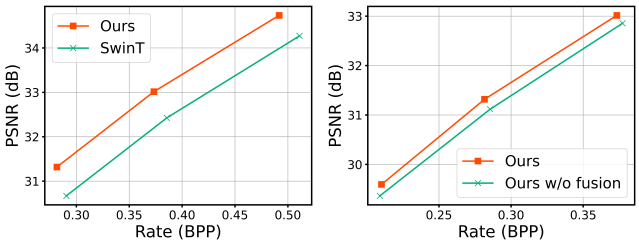

Figure 8: Ablation results on SIQAD. (a) Backbone ablation by comparing with Swin Transformer Blocks. (b) The ablation studies on DFF module ("w/o fusion" means removing DFF in the channel-wise context entropy model).

We convert MS-SSIM to $-10\log_{10}(1 - \text{MS-SSIM})$ to ease the comparison. Remarkably, our proposed method surpasses the state-of-the-art H.266 screen content coding tool, VVC-SCC, especially at lower bit rates (below 0.2 bpp). Our model achieves this by effectively prioritizing the most visually significant information for compression, thereby retaining higher image quality even at lower bitrates. For a quantitative evaluation, we employ BD-rate derived from PSNR-BPP curves as the metric. Remarkably, our method exceeds VVC (VTM-20.0) by 17.00%, 35.94%, and 51.04% in BD-rate on the CCT, SCID, and SIQAD datasets, respectively. These results are summarized in Table 1. *More comparisons are reported in supplementary.*

Qualitative Visualization

We perform a meticulous qualitative analysis by visually comparing the results generated by our proposed model with those generated by CNN-based models [Minnen *et al.*, 2018; Cheng *et al.*, 2020] and Transformer-based models [Zou *et al.*, 2022; Liu *et al.*, 2023], using images from the SIQAD dataset, specifically the 'siqad17'. Figure 1 illustrates the results of this comparative analysis. We focus on two distinct local regions to assess detail reconstruction. Our method outperformed others in the upper local region of 'siqad17', generating fewer artifacts, especially in the reconstruction of the characters "O" and "V", and enhancing the clarity of background elements like the drainage pipe. *For more visual results, see the supplementary material.*

4.3 Ablation Studies

SCAB Module

We demonstrate the effectiveness of our proposed SCAB through an ablation study, comparing our model with an alternative using Swin Transformer blocks [Liu *et al.*, 2021]. Unlike Swin Transformer blocks that focus on local attention, SCAB calculates global attention over superpixels for more holistic information capture. The study results, presented in Figure 8, show that our model with SCAB blocks significantly improves efficiency and performance over the "SwinT" model that employs Swin Transformer blocks.

DFF Module

To highlight the efficacy of the DFF Module within our proposed entropy model, we conducted ablation studies and present the results in Figure 6 and Figure 8(b). After removing the DFF module, the RD performance is degraded, as shown in Figure 8(b). Furthermore, the visualization of residual errors in Figure 6 also shows that the probabilities estimated using the DFF module are more accurate, resulting in smaller latent feature errors. These results confirm the crucial role of the DFF module in improving the performance of the entropy model.

4.4 Complexity and Efficiency Analysis

We evaluate the computational complexity and decoding quality of various image compression methods [Minnen *et al.*, 2018; Cheng *et al.*, 2020; Zou *et al.*, 2022; Liu *et al.*, 2023], including our proposed method, based on the SIQAD dataset. Figure 9 visualizes the trade-off between the model complexity (parameters and inference time) and the compression performance (BD-rate) for various models. Our proposed method achieves an impressive balance by maintaining faster inference times and a relatively smaller model size while also reaching lower BD-rates. *For detailed information, see the supplementary material.*

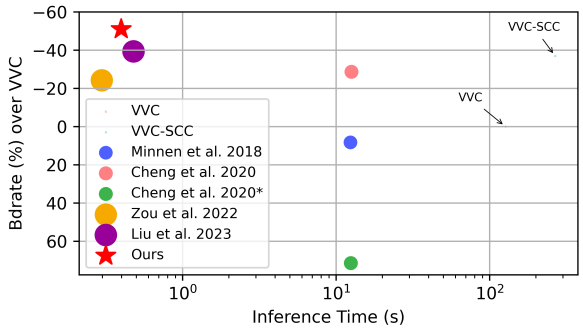

Figure 9: Compression performance vs. computing complexity. The figure depicts the trade-off between compression performance, measured by the average BD-rate relative to VVC, and computing complexity, indicated by inference time (encoding+decoding) and amount of parameter numbers, for various compression methods. The BD-rate is averaged over all test images in the SIQAD dataset. Notably, "Cheng et al. 2020*" represents the model pre-trained on natural images. For representation purposes, VVC and VVC-SCC are assigned negligible parameter sizes as they do not possess parameters.

5 Conclusion

In this work, we propose a new end-to-end network for screen content image compression. Specifically, a superpixel-based content aggregation block is proposed to aggregate local regional information through superpixel grouping and the global redundancy is explored via super-pixel based transformer. Additionally, we also introduce a dynamic feature fusion mechanism to enhance the channel-wise context entropy model. The DFF module adaptively adjusts the probability distribution of undecoded channels, optimizing information utilization during decoding. Comprehensive experiments on three datasets validate the superiority of our method in terms of both RD performance and efficiency.

Acknowledgements

This work was supported in part by National Natural Science Foundation of China under Grant 62231018 and Grant 62072331.

References

[Ahmed *et al.*, 1974] Nasir Ahmed, T_ Natarajan, and Kamisetty R Rao. Discrete cosine transform. *IEEE transactions on Computers*, 100(1):90–93, 1974.

[Ballé *et al.*, 2016] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. End-to-end optimized image compression. *arXiv preprint arXiv:1611.01704*, 2016.

[Ballé *et al.*, 2018] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. *arXiv preprint arXiv:1802.01436*, 2018.

[Ballé *et al.*, 2020] Johannes Ballé, Philip A Chou, David Minnen, Saurabh Singh, Nick Johnston, Eirikur Agustsson, Sung Jin Hwang, and George Toderici. Nonlinear transform coding. *IEEE Journal of Selected Topics in Signal Processing*, 15(2):339–353, 2020.

[Bégaint *et al.*, 2020] Jean Bégaint, Fabien Racapé, Simon Feltman, and Akshay Pushparaja. Compressai: a pytorch library and evaluation platform for end-to-end compression research. *arXiv preprint arXiv:2011.03029*, 2020.

[Bross *et al.*, 2021] Benjamin Bross, Ye-Kui Wang, Yan Ye, Shan Liu, Jianle Chen, Gary J Sullivan, and Jens-Rainer Ohm. Overview of the versatile video coding (vvc) standard and its applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(10):3736–3764, 2021.

[Cheng *et al.*, 2020] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro 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*, pages 7939–7948, 2020.

[Guo *et al.*, 2021] Zongyu Guo, Zhizheng Zhang, Runsen Feng, and Zhibo Chen. Causal contextual prediction for learned image compression. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(4):2329–2341, 2021.

[He *et al.*, 2021] Dailan He, Yaoyan Zheng, Baocheng Sun, Yan Wang, and Hongwei Qin. Checkerboard context model for efficient learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14771–14780, 2021.

[Huang *et al.*, 2022] Huaibo Huang, Xiaoqiang Zhou, Jie Cao, Ran He, and Tieniu Tan. Vision transformer with super token sampling. *arXiv:2211.11167*, 2022.

[Jampani *et al.*, 2018] Varun Jampani, Deqing Sun, Ming-Yu Liu, Ming-Hsuan Yang, and Jan Kautz. Superpixel sampling networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 352–368, 2018.

[Liu *et al.*, 2015] Shan Liu, Xiaozhong Xu, Shawmin Lei, and Kevin Jou. Overview of hevc extensions on screen content coding. *APSIPA Transactions on Signal and Information Processing*, 4:e10, 2015.

[Liu *et al.*, 2021] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[Liu *et al.*, 2023] Jinming Liu, Heming Sun, and Jiro Katto. Learned image compression with mixed transformer-cnn architectures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1–10, 2023.

[Min *et al.*, 2021] Xiongkuo Min, Ke Gu, Guangtao Zhai, Xiaokang Yang, Wenjun Zhang, Patrick Le Callet, and Chang Wen Chen. Screen content quality assessment: overview, benchmark, and beyond. *ACM Computing Surveys (CSUR)*, 54(9):1–36, 2021.

[Minnen and Singh, 2020] David Minnen and Saurabh Singh. Channel-wise autoregressive entropy models for learned image compression. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 3339–3343. IEEE, 2020.

[Minnen *et al.*, 2018] David Minnen, Johannes Ballé, and George D Toderici. Joint autoregressive and hierarchical priors for learned image compression. *Advances in neural information processing systems*, 31, 2018.

[Mitrica *et al.*, 2019] Iulia Mitrica, Eric Mercier, Christophe Ruellan, Attilio Fiandrotti, Marco Cagnazzo, and Béatrice Pesquet-Popescu. Very low bitrate semantic compression of airplane cockpit screen content. *IEEE Transactions on Multimedia*, 21(9):2157–2170, 2019.

[Rabbani and Joshi, 2002] Majid Rabbani and Rajan Joshi. An overview of the jpeg 2000 still image compression standard. *Signal processing: Image communication*, 17(1):3–48, 2002.

[Shen *et al.*, 2022] Sheng Shen, Huanjing Yue, Jingyu Yang, and Kun Li. Itsrn++: Stronger and better implicit transformer network for continuous screen content image super-resolution. *arXiv preprint arXiv:2210.08812*, 2022.

[Sullivan *et al.*, 2012] Gary J Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand. Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on circuits and systems for video technology*, 22(12):1649–1668, 2012.

[Tang *et al.*, 2022] Tong Tang, Ling Li, Xiaoyu Wu, Ruizhi Chen, Haochen Li, Guo Lu, and Limin Cheng. Tsa-scc: text semantic-aware screen content coding with ultra low bitrate. *IEEE Transactions on Image Processing*, 31:2463–2477, 2022.

[Wallace, 1992] Gregory K Wallace. The jpeg still picture compression standard. *IEEE transactions on consumer electronics*, 38(1):xviii–xxxiv, 1992.

[Wang *et al.*, 2022] Meng Wang, Kai Zhang, Li Zhang, Yaojun Wu, Yue Li, Junru Li, and Shiqi Wang. Transform skip inspired end-to-end compression for screen content image. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 3848–3852. IEEE, 2022.

[Xie *et al.*, 2021] Yueqi Xie, Ka Leong Cheng, and Qifeng Chen. Enhanced invertible encoding for learned image compression. In *Proceedings of the 29th ACM international conference on multimedia*, pages 162–170, 2021.

[Xu *et al.*, 2016] Xiaozhong Xu, Shan Liu, Tzu-Der Chuang, Yu-Wen Huang, Shaw-Min Lei, Krishnakanth Rapaka, Chao Pang, Vadim Seregin, Ye-Kui Wang, and Marta Karczewicz. Intra block copy in hevc screen content coding extensions. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 6(4):409–419, 2016.

[Zamanshoar Heris and Bajić, 2023] Rashid Zamanshoar Heris and Ivan V Bajić. Multi-task learning for screen content image coding. *arXiv e-prints*, pages arXiv–2302, 2023.

[Zhu *et al.*, 2022] Yinhao Zhu, Yang Yang, and Taco Cohen. Transformer-based transform coding. In *International Conference on Learning Representations*, 2022.

[Zou *et al.*, 2022] Renjie Zou, Chunfeng Song, and Zhaoxiang Zhang. The devil is in the details: Window-based attention for image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17492–17501, 2022.