Robust End-to-End Image Transmission with Residual Learning

Cenk M. Yetis
Lund Research Center, Huawei Technologies
cenkmyetis@ieee.org

***Abstract*—Recently, deep learning (DL) based image transmission at the physical layer (PL) has become a rising trend due to its ability to significantly outperform conventional separation-based digital transmissions. However, implementing solutions at the PL requires a major shift in established standards, such as those in cellular communications. Application layer (AL) solutions present a more feasible and standards-compliant alternative. In this work, we propose a layered image transmission scheme at the AL that is robust to end-to-end (E2E) channel errors. The base layer transmits a coarse image, while the enhancement layer transmits the residual between the original and coarse images. By mapping the residual image into a latent representation that aligns with the structure of the E2E channel, our proposed solution demonstrates high robustness to E2E channel errors.**

***Index Terms*—Semantic communication, residual image, channel error robustness.**

I. Introduction

6G aims to meet extreme requirements across all aspects of wireless communications, necessitating a departure from traditional approaches and the exploration of new dimensions. Semantic communication (SC) is considered a crucial step towards achieving human-like intelligence and is expected to significantly transform all aspects of next-generation networks. By focusing on conveying the meaning or intent of the data rather than just the bits, SC can drastically reduce the amount of data transmitted, thereby improving efficiency and performance,

One of the early successes in this domain is semantic-based deep joint source and channel coding (SB-DJSCC), which has demonstrated the potential to outperform traditional separate source and channel coding methods [1]. While some works align better with existing standards through the use of bitstreaming [2], their implementation still requires significant modifications to current communication standards. For a more immediate standards-compliant SC solution, an attractive alternative is to implement SC functionalities within the application layer (AL) [3].

In parallel, neural network (NN) based image compression algorithms utilizing residual coding have gained attention [4]–[7]. Residual coding involves transmitting a residual image r, representing the difference between the original x and a coarse base layer image x', as an enhancement layer, i.e., $r = x - x'$. This approach helps in progressively improving the image quality [8]. However, existing NN solutions with residual coding often incorporate conventional compression techniques such as BPG and JPEG [4], [5], [7] or directly transmit the residual without specific channel considerations [6].

979-8-3315-6655-5/25/$31.00 ©2025 IEEE

In this work, we propose deep semantics source channel coding (DS^2C^2) embedded in the AL that is robust to end-to-end (E2E) channel errors. Unlike previous works, DS^2C^2 is a complete end-to-end NN solution for layered image transmission with residual coding, specifically designed for robustness in error-prone channels. The robustness is achieved by mapping source data into a latent representation which has a similar structure with the E2E channel characteristics. To the best of our knowledge, this is the first work to discover this technique, and numerical results show that our method reduces the image quality degradation from around 20% to just 3% compared to a key competitive NN solution.

The key contributions of this paper are summarized as follows:

- We propose DS^2C^2, a novel deep learning-based semantic-empowered multi-layer image transmission scheme operating entirely at the application layer, offering a standards-compliant approach for next-generation networks.
- We design a system that demonstrates high robustness to end-to-end channel errors by strategically mapping the residual image into a latent representation whose structure aligns well with the channel characteristics.
- We introduce a disjoint architecture comprising recursive generative adversarial network (RGAN) and block residual network (BResNet) (+SumNet at the receiver), providing flexibility and adaptability to changing network conditions.
- We provide numerical and visual results demonstrating the effectiveness and robustness of DS^2C^2 under various channel error conditions and show its capability for efficient updating with a low number of training images.

DS^2C^2 differs from [4]–[7] in several key aspects:

1) DS^2C^2 is a complete NN solution, where NN is used for encoding the residual image, referred to as BResNet.
2) Two consecutive generative adversarial networks (GANs) are implemented, referred to as RGAN.
3) NN is used to improve the summation accuracy at the receiver, referred to as SumNet.
4) DS^2C^2 employs a disjoint design of two main NN blocks: RGAN and BResNet (+ SumNet at the receiver). These two NN blocks are trained separately, offering modularity.
5) It is assumed that the base layer image x' is received

without error, allowing the system to focus on protecting the residual information.

The reasons behind these design choices are intertwined.

Items 1, 4 and 5: The 1st NN block (RGAN) generates highly compressed image data (the base layer image) that can be strongly protected, ensuring it is received error-free. This means that when there are changes in the network affecting the residual image, only the 2nd NN block (BResNet+SumNet) needs to be retrained or updated. However, if an error-free assumption for the base layer is not feasible, both NN blocks would require retraining. Consequently, the disjoint design provides flexibility in different deployment scenarios.

Items 2 and 4: Since the 1st and 2nd NN blocks are trained independently, the number of RGANs in the first block can be adjusted based on channel conditions. For instance, if the output of the BResNet (2nd NN block) encounters harsher channel conditions, the number of RGANs should potentially be increased, allowing the residual to carry less critical information as x' is made closer to the original image x.

Item 3: As detailed later, a simple summation operation at the receiver is insufficient to handle the complexities during transitions between tensor and image domains, particularly concerning most significant bits (MSB) and least significant bits (LSB). Therefore, we incorporate SumNet to improve the summation accuracy at the receiver.

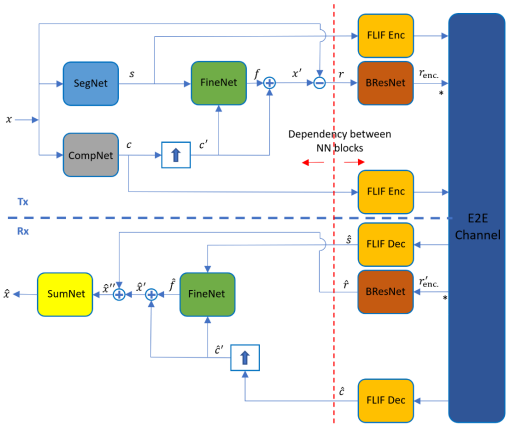
II. System Model and Related Works

In this section, we introduce the DS^{2}C^2 architecture and compare it to the existing NN architectures using residual coding. We also apply quantization of latents since it is effective to achieve highly compressed outputs while keeping the NN architecture relatively simple. Finally, we interleave the quantized latents before the transmission to achieve robustness against block errors in the channel.

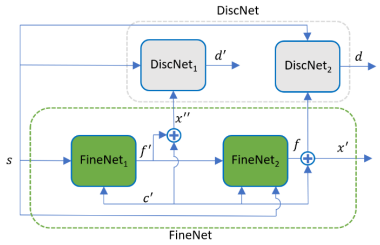
A. DS^{2}C^2 architecture

DS^{2}C^2 architecture and the inherent RGAN architecture are shown in Fig. 1. In Fig. 1, segmentation network (SegNet) generates a semantic image, s, such as labels and boundary maps. Compression network (CompNet) generates a downsampled version of the original image, c, which is later upsampled to c'. Then s and c' are fed into FineNet. The fine details in the FineNet output f are summed with c' to get a better estimate, x', coined as synthesized image. The residual image r is obtained by a subtraction operation and then encoded by block residual network (BResNet). At the receiver, in addition to the complementary operations to the transmitter, a summation network (SumNet) is included to improve the summation accuracy. Also, quantization and interleaving operations are applied after BResNet. This part is marked by star in Fig. 1 and further details are deferred to the next section.

During the training phase, RGAN comprises two networks: the generator (FineNet) and the discriminator (DiscNet) networks [5], as shown in Fig. 1. However, during the test phase, DiscNet is not used. Only a 2-step RGAN is shown in Fig. 1.



(a) DS^{2}C^2 architecture.



(b) RGAN architecture.

Fig. 1: DS^{2}C^2 and the inherent RGAN architectures.

If more steps are taken, the estimation quality of synthesized image x' increases at the cost of increased computational complexity. DiscNet$_i$ is a 3-staged multi-scale discriminator network as detailed in [9]. The same architectures as in [5] are used in SegNet, CompNet and FineNet$_i$.

DS^{2}C^2 offers a complete NN solution, whereas [4], [5], [7] use BPG or JPEG2000 in their solutions. In [6], r is directly sent to the channel.

B. Quantization and interleaver

In Fig. 2, the details of quantization and interleaver in the DS^{2}C^2 architecture are provided, expanding on the star marks in Fig.1. Quantization followed by entropy coding effectively improves the rate-distortion performance. Rate-distortion optimization has been addressed through two approaches so far: 1) Joint optimization of rate and distortion [4], and 2) Minimization of distortion given a fixed rate [8]. In this work, we use the latter approach by leveraging statistical binarizer [8] that allows nearly accurate predetermination of bits per pixel (BPP) before both training and inference phases. The binarizer facilitates a direct $1/8$ BPP compression, and further compression is achieved by adjusting the latent dimensions. By fixing the BPP values, we can more definitively analyze the channel effects for the proposed DS^{2}C^2 architecture.

The binary symmetric channel (BSC) and binary erasure channel (BEC) are well-known channel models. BSC randomly flips the bits, while the BEC has ternary outputs despite the binary inputs. In this work, we assume BEC model with a bit probability of flipping from 1 to 0 (or equivalently from 0 to 1). Furthmore, we assume block BEC (BBEC) in our

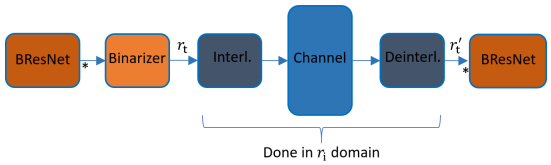


Fig. 2: Binarization, interleaving, channel error modelling, and deinterleaving operations used in DS^{2}C^2.

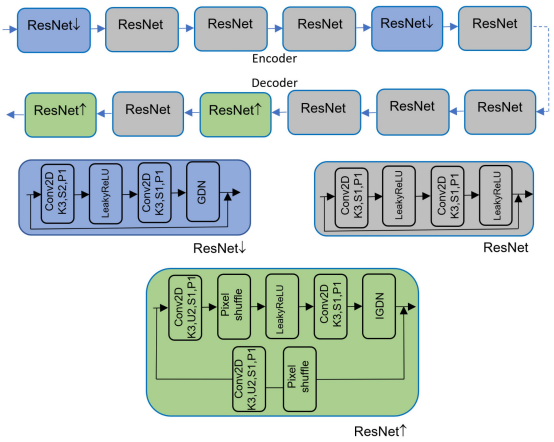


Fig. 3: The details of BResNet encoder and decoder in the DS^{2}C^2 architecture.

work because, according to 5G NR specifications, a code block can be a maximum of 1056 bytes. In the context of our simulations, BBEC model is implemented as follows: the transmitted binary latent representation is divided into blocks of a fixed size (e.g., corresponding to the 1056 bytes from 5G NR specifications). For each block, with a certain probability, the entire block is erased, meaning all bits within that block are marked as lost or corrupted. If a block is not erased, all bits within it are received correctly. This model simulates scenarios where entire data blocks are lost or severely corrupted during transmission, which is relevant in packet-based communication systems. To mitigate the severity of block losses or block errors, a pixel-level interleaver is used before the transmission.

C. BResNet architecture

In this section, the BResNet encoder and decoder blocks used in DS^{2}C^2 are detailed. In Fig. 3, each ResNet sub-block refers to a 2D convolutional NN with a residual connection. For the normalization of the layers, generalized divisive normalisation (GDN) and inverse GDN (IGDN) are used. Also, a pixel shuffle in the upsampling process is adopted instead of transpose convolution. Finaly, K, S, P, and U refer to kernel, stride, and padding sizes, and upsampling factor, respectively. For SumNet, 3 consecutive ResNet (gray box) sub-blocks are used.

III. Robust Semantic Image Transmission with Residual Coding

In this section, the proposed DS^{2}C^2 architecture is further detailed. The loss function of RGAN is given as:

$$\mathcal{L}_{\text{RGAN}} = \mathcal{L}_{\text{G}} + \mathcal{L}_{\text{D}}, \text{ where} \tag{1}$$

$$\mathcal{L}_{\text{G}} = -\sum_{j=1}^{2}\sum_{i=1}^{3} \log \text{D}_{ji}(s, c', x_j') + \mathcal{L}_{\text{d}} \tag{2}$$

$$\mathcal{L}_{\text{D}} = -\sum_{i=1}^{3} \log \text{D}_{i}(s, c', x) - \sum_{j=1}^{2}\sum_{i=1}^{3} \log(1 - \text{D}_{ji}(s, c', x_j')) \tag{3}$$

$$\mathcal{L}_{\text{d}} = \mathcal{L}_1 + \mathcal{L}_{\text{SSIM}} + \mathcal{L}_{\text{VGG}}. \tag{4}$$

Here, $\mathcal{L}_{\text{d}}$ is the distance measure between the original and estimated images, where $\mathcal{L}_1, \mathcal{L}_{\text{SSIM}}$, and $\mathcal{L}_{\text{VGG}}$ are the well-known L1-norm, structural similarity index measure (SSIM), and pre-trained visual geometry group (VGG) network loss evaluations [5], respectively.

For BResNet and SumNet, the $\mathcal{L}_{\text{d}}$ loss metric defined above is used as well. The training of DS^{2}C^2 is achieved in two steps. First, RGAN is trained by minimizing $\mathcal{L}_{\text{RGAN}}$. Then, the pretrained RGAN is loaded for the joint training of BResNet and SumNet by minimizing $\mathcal{L}_{\text{BResNet}} + \mathcal{L}_{\text{SumNet}}$.

It is well-known that statistical dependence in the latent representation leads to suboptimal compression performance. In [10], standard deviations of latents are predicted and used to obtain independent latents, thereby improving rate-distortion performance. However, in this work, we demonstrate that structured latent representations can enhance robustness against channel errors.

In Fig. 4, the encoded images before the interleaver r_{i} and after the deinterleaver r_{i}' are shown for Cityscapes and Kodak datasets. During the training phase, probability of error pe_{train} is set to 0%, while during the test phase, probability of error pe_{test} is set to 16%, e.g., 16% of 1s in an image are corrupted by the channel and flipped to 0s. r_{i} already exhibits a noisy structure. Consequently, even after experiencing channel errors, r_{i}' maintains a structure similar to r_{i}. Contrary to expectations, this indicates that channel errors do not significantly impact r_{i}, as the structure of r_{i} remains largely unchanged after the E2E channel errors. In other words, the residual image is mapped into a latent representation that aligns well with the E2E channel characteristics. Hence, robustness against channel errors is achieved.

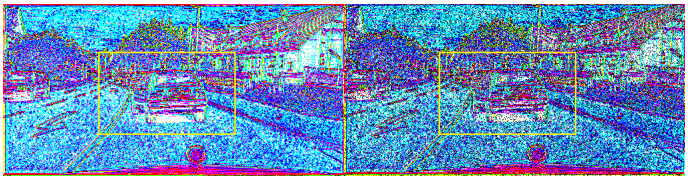
IV. Numerical Results

In this section, we provide numerical and visual results for varying BPP, pe_{train}, and pe_{test} parameters. In addition, we benchmark DS^{2}C^2 with a competitive NN solution in [11] coined here as LIC-GMM-AM. Finally, we demonstrate that DS^{2}C^2 can be effectively trained even with a small number of training images.

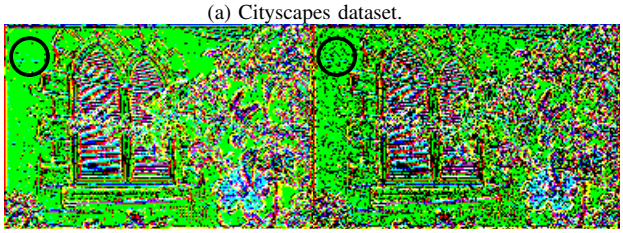
The experiments are conducted using the Cityscapes, ADE20K, and Kodak datasets. In the Cityscapes and ADE20K tests, 50 images are used for each experiment, which are excluded from their respective training datasets. To demonstrate the generalization capability of DS^{2}C^2, the pre-trained DS^{2}C^2 model on the ADE20K dataset is applied to the classical Kodak dataset, which includes 24 test images.

A. Benchmarking to existing NN architecture

Similar to conventional compression techniques, LIC-GMM-AM has no channel error correction capability un-



(a) Cityscapes dataset.



(b) Kodak dataset.

Fig. 4: Similar structures observed in the latent representations of r_{i} (left) and r_{i}' (right) for Cityscapes and Kodak datasets.

TABLE I: DS^{2}C^2 vs. LIC-GMM-AM for Kodak dataset at 0.26 BPP. Benchmark case: $\mathrm{pe}_{\mathrm{train}}= 0$, $\mathrm{pe}_{\mathrm{test}}= 0$. Trained without channel errors: $\mathrm{pe}_{\mathrm{train}}= 0$, $\mathrm{pe}_{\mathrm{test}}= 16$. Trained with channel errors: $\mathrm{pe}_{\mathrm{train}}= 8$, $\mathrm{pe}_{\mathrm{test}}= 16$.

PSNR (Loss %)	Benchmark	Trained without channel errors	Trained with channel errors
DS^{2}C^2	28.25	25.03 (**11.4%**)	27.32 (**3.3%**)
LIC-GMM-AM	30.61	-1.18 (**104%**)	24.14 (**21.14%**)

less it is trained against channel errors. For $\mathrm{pe}_{\mathrm{train}} = 0$ and $\mathrm{pe}_{\mathrm{test}} = 16$, peak signal-to-noise ratio (PSNR) of LIC-GMM-AM drops to a negative PSNR value of -1.18 from 30.61. When channel errors are introduced during training, for $\mathrm{pe}_{\mathrm{train}} = 8$ and $\mathrm{pe}_{\mathrm{test}} = 16$, it can achieve only 24.14 PSNR. On the other hand, DS^{2}C^2 achieves PSNR values of 25.03 and 27.32 for $(\mathrm{pe}_{\mathrm{train}} = 0, \mathrm{pe}_{\mathrm{test}} = 16)$ and $(\mathrm{pe}_{\mathrm{train}} = 8, \mathrm{pe}_{\mathrm{test}} = 16)$, respectively. The results and loss percentages relative to benchmarks are summarized in Table I.

B. Low number of training images

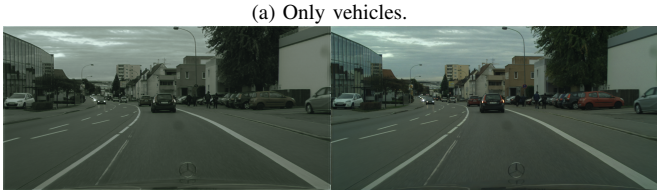
Swift update of NNs is a critical issue when an immediate update is needed due to changing conditions or insufficient training for particular conditions. For re-training, updating NNs with much less NN inputs compared to the initial training is very important for a quick response to avoid losing quality of experience (QoE). Low number of NN inputs for updating is also advantageous when collecting new training inputs which can be challenging and time consuming.

In this section, we demonstrate that DS^{2}C^2 can be quickly updated with a small number of training images. As seen in Fig. 5, the number of training images required for an efficient update can remain quite low, depending on the application. This makes DS^{2}C^2 particularly promising in cases where fine details are less critical, as illustrated in Fig. 5.

V. Conclusion

In this work, we present a robust deep learning (DL) based semantic-empowered multi-layer image transmission scheme

(a) Only vehicles.



(b) Vehicles and pedestrians.

Fig. 5: Estimated images of DS^{2}C^2 for Cityscapes dataset at 0.54 BPP for varying number of training images. Left: 16. Right: 2968.

that is resilient to end-to-end (E2E) channel errors. With moderate changes to existing standards, the proposed deep semantics source channel coding (DS^{2}C^2) solution can be viably implemented at the application layer (AL). We believe that flexibility is as important as achieving high estimation quality at high compression region. Thus, the proposed disjoint design of layered compressed image transmission offers adaptability across various scenarios under channel errors.

References

[1] D. Gunduz, Z. Qin, I. E. Aguerri, H. S. Dhillon, Z. Yang, A. Yener, K. K. Wong, and C. B. Chae, "Beyond transmitting bits: Context, semantics, and task-oriented communications," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 1, pp. 5–41, 2023.

[2] D. Huang, F. Gao, X. Tao, Q. Du, and J. Lu, "Toward semantic communications: Deep learning-based image semantic coding," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 1, pp. 55–71, 2023.

[3] Q. Lan, D. Wen, Z. Zhang, Q. Zeng, X. Chen, P. Popovski, and K. Huang, "What is semantic communication? A view on conveying meaning in the era of machine intelligence," *J. of Commun. and Inf. Netw.*, vol. 6, no. 4, pp. 336–371, 2021.

[4] W. C. Lee, D. Alexandre, C. P. Chang, W. H. Peng, C. Y. Yang, and H. M. Hang, "Learned image compression with residual coding," in *IEEE Conf. on Comp. Vision and Pattern Recog. (CVPR)*, Jun. 2019, pp. 1–5.

[5] M. Akbari, J. Liang, and J. Han, "DSSLIC: Deep semantic segmentation-based layered image compression," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Process. (ICASSP)*, May 2019, pp. 2042–2046.

[6] D. Huang, X. Tao, F. Gao, and J. Lu, "Deep learning-based image semantic coding for semantic communications," in *IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2021, pp. 1–6.

[7] C. Dong, H. Liang, X. Xu, S. Han, B. Wang, and P. Zhang, "Semantic communication system based on semantic slice models propagation," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 1, pp. 202–213, 2023.

[8] G. Toderici, S. M. O'Malley, S. J. Hwang, D. Vincent, D. Minnen, S. Baluja, M. Covell, and R. Sukthankar, "Variable rate image compression with recurrent neural networks," in *Int. Conf. on Learning Representations (ICLR)*, May 2016.

[9] T. C. Wang, M. Y. Liu, J. Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional GANs," in *IEEE Conf. on Comp. Vision and Pattern Recog. (CVPR)*, Jun. 2018, pp. 8798–8807.

[10] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," in *Int. Conf. on Learning Representations (ICLR)*, May 2018.

[11] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized Gaussian mixture likelihoods and attention modules," in *IEEE Conf. on Comp. Vision and Pattern Recog. (CVPR)*, Jun. 2020, pp. 7936–7945.