Semantic Communication Enabling Robust Edge Intelligence for Time-Critical IoT Applications

Andrea Cavagna, Nan Li, Alexandros Iosifidis and Qi Zhang

DIGIT, Department of Electrical and Computer Engineering, Aarhus University.
Email: andrea.cavagna.94@gmail.com, {linan, ai, qz}@ece.au.dk

Abstract—This paper aims to design robust Edge Intelligence using semantic communication for time-critical IoT applications. We systematically analyze the effect of image DCT coefficients on inference accuracy and propose the channel-agnostic effectiveness encoding for offloading by transmitting the most meaningful task data first. This scheme can well utilize all available communication resource and strike a balance between transmission latency and inference accuracy. Then, we design an effectiveness decoding by implementing a novel image augmentation process for convolutional neural network (CNN) training, through which an original CNN model is transformed into a Robust CNN model. We use the proposed training method to generate Robust MobileNet-v2 and Robust ResNet-50. The proposed Edge Intelligence framework consists of the proposed effectiveness encoding and effectiveness decoding. The experimental results show that the effectiveness decoding using the Robust CNN models perform consistently better under various image distortions caused by channel errors or limited communication resource. The proposed Edge Intelligence framework using semantic communication significantly outperforms the conventional approach under latency and data rate constraints, in particular, under ultra stringent deadlines and low data rate.

Index Terms—Semantic communication, Edge Intelligence, low latency, CNN inference, service reliability

I. INTRODUCTION

Artificial Intelligence based on convolutional neural networks (CNN) is widely applied for Internet of Things (IoT) applications. For time-critical IoT applications, it is indispensable to provide services not only of high reliability (e.g., high inference accuracy) but also within ultra-low latency. However, it is challenging for resource-constrained IoT devices to run computation-intensive inference tasks in real time locally [1].

Edge Computing is a promising approach to address this challenge; however, most of the existing works in Edge Computing focus on reliable transmission in offloading (i.e., accurately receiving the transmitted bits/packets) instead of service reliability [2]. The work [1] shows that it is more important to ensure service reliability from an application’s perspective, i.e., to provide good inference accuracy within the deadline. This is aligned with semantic communication [3], which is usually composed of semantic/effectiveness encoding and semantic/effectiveness decoding to achieve that the transmitted data can precisely convey the desired meaning and the received data enable the receiver to achieve a final goal using the received data, e.g., to perform inference or make the right decision, instead of simply focusing on reliable data transmission. Effectiveness encoding is the process to encode large raw data into more compact data representation. In the context of Edge Intelligence, effectiveness decoding basically includes receiving task data, decoding the data (e.g., image reconstruction), and performing inference at Edge server [4].

In time-critical IoT applications, on the one hand, it needs to decide how much data can be transmitted within limited communication time budget, furthermore, retransmission is often not feasible in offloading due to the excessive transmission latency it causes. On the other hand, as we know that slight distortion in input images will not cause significant loss of inference accuracy [5]. For example, the face recognition can still achieve acceptable inference accuracy as long as less than 13.3% of the image pixels are corrupted [6]. This shows that inference accuracy is not fully dependent on bit-level transmission reliability. Therefore, it is worth studying how to design an effectiveness encoding that uses less communication resource (e.g., transmission time, bandwidth) and how to design the corresponding effectiveness decoding that consistently achieves high inference accuracy.

In this paper, we aim to design an Edge Intelligence framework using semantic communication paradigm, which can enhance CNN-based inference at an edge server (ES) within strict time constraints, while ensuring decent inference accuracy even under severe channel errors without retransmission. In particular, the key research questions are in the following: (1) How can we design an effectiveness encoding to reduce transmission data size and shorten transmission latency while ensuring inference accuracy? (2) How can we design an effectiveness decoding to ensure inference accuracy without using ARQ-based retransmission nor FEC with low code rate?

To address these questions, we encode an image using its discrete cosine transform (DCT) representation [7] and the DCT coefficients are transmitted in sequence based on their importance. The effectiveness decoder can reconstruct an image from any subset of its original DCT representation. The decoder is designed in such a way that it can dynamically terminate reception if the communication time budget is over and start inference using all received DCT coefficients. To ensure inference accuracy with low-quality input image due to not using retransmission, we design the effectiveness decoder to cope with images of few DCT coefficients and images containing bit errors or packet losses, which is trained by a novel CNN training method. The main contributions are summarized as follows.

- We systematically analyze the effect of missing DCT
coefficients on image classification, and propose the channel-agnostic effectiveness encoding for offloading, which can make a flexible utilization of all available communication resource and strike a good balance between transmission latency and inference accuracy.

- We implement the effectiveness decoding through a novel image augmentation process for CNN training. It transformed original CNN models into new Robust CNN models that can provide high inference accuracy without all DCT coefficients. We use this training method to generate robust versions of MobileNet-v2 [8] and ResNet-50 [9]. The experimental results show that the Robust CNN models outperform their original models under missing DCT coefficients. For example, Robust ResNet-50 achieves a gain of inference accuracy up to 17.9% on flower dataset [10] and 25.3% on Caltech 256 [11].

- To demonstrate the feasibility of our designed effectiveness decoding using the Robust CNN models, we test their performance in an error-prone communication channel with different bit error probabilities and packet loss rates. The experimental results show that the Robust CNN models can perform better than their original models, in particular, when packet loss rate is high. For example, Robust MobileNet-V2 achieves a gain of average inference accuracy up to 20.4% on Caltech 256 when 5 packets in each image are lost.

- We further study the performance of our proposed Edge Intelligence framework using semantic communication under the latency and data rate constraints. The experimental results show that it outperforms the conventional framework, achieving a gain up to 40% for flower dataset using Robust ResNet-50. In particular, under extremely stringent deadlines and low data rates, the proposed framework can still achieve decent inference accuracy whereas the conventional approach will completely fail.

II. EDGE INTELLIGENCE USING SEMANTIC COMMUNICATION

This section describes the design of the proposed Edge Intelligence with semantic communication. To ensure inference accuracy within latency constraint, it is important to make use of the communication time budget to receive the best possible representation of task data, e.g., an image, regardless of wireless channel state. For image classification service, we design an effectiveness encoding which can perform fast image encoding for efficient computation offloading, and an effectiveness decoding which can dynamically terminate the reception and start image reconstruction and inference. Furthermore, we design and implement the effectiveness decoding in such a way that it still achieves high inference accuracy under missing DCT coefficients.

A. Channel-agnostic Effectiveness Encoding for Offloading

In conventional offloading scheme for applications with hard deadline, an IoT device first estimates the available time budget for data transmission based on the average task computation time and it calculates the amount of data that can be transmitted within the time budget based on channel estimation. Then, an IoT device can select proper compression parameters, for example using JPEG, to ensure completing inference at ES within the latency constraint. However, due to time-varying wireless channel, continuous channel estimation is required to do this. In addition, the actual data rate and channel bit error probability might not be accurately estimated. Furthermore, JPEG has limited compression options, and in some cases the minimal size of a compressed image may still be larger than the allowed data size, which can cause significant distortions in a JPEG image.

To address these issues, we design a channel-agnostic effectiveness encoding for offloading, as shown in Fig. 1. The basic idea is to let IoT device transmit as many DCT coefficients as possible without the knowledge of communication channel state and allow ES to decide when to start inference. This design is based on the fact that each DCT coefficient makes an additive contribution to the received image quality.

The rationale behind this encoding design is as follows. Each DCT coefficient encodes frequency information of an 8 × 8 block in an image. This means that every received DCT coefficient contributes information to the entire block. Using this type of compression will significantly reduce the size of a compressed image, especially in blocks with fairly uniform luminance or chrominance (i.e., most of the DCT coefficients are close to zero). Thus, the received image gradually gets refined after receiving more and more DCT coefficients within the allowed communication time. As IoT device and ES are synchronized, an ES can start to decode the received DCT coefficients and classify the reconstructed image as soon as the communication time budget is over. In this way, it is able to provide a representation of the input image by fully utilizing the available communication resource. Furthermore, it avoids the processing, such as channel estimation and selecting compression parameters, thereby reducing energy consumption at IoT device.

To better illustrate the importance of different DCT coefficients for image quality, we visually present the reconstructed images under different scenarios of missing DCT coefficients.
The DCT coefficients’ importance is ordered based on JPEG ZigZag matrix [12]. To achieve the best possible inference accuracy with limited communication time for offloading, it is better to transmit the most important DCT coefficients first. As shown in Fig. 2(b)-2(c), the loss of the 1st DCT coefficient results in severe image distortion. Note that for the flower dataset all DCT coefficients of an image are packetized into 36 packets and the first 2 packets contain all the 1st DCT coefficients. Fig. 2(d)-2(e) show the distorted images due to the 1st or 2nd packet loss. We can see that it has a great impact on a reconstructed image, as the packet contains the 1st DCT coefficients. Fig. 2(f)-2(g) show the examples of keeping the top-1 and top-5 coefficients of the 64 DCT coefficients. We can see that the image can be visually well recognized with merely the top-5 DCT coefficients. This implies that the loss of high frequency components would not significantly reduce inference accuracy, which is verified in Section III-A.

B. Robust CNN Training Method for Effectiveness Decoding

In the context of Edge Intelligence, the effectiveness decoding in semantic communication is to reconstruct image from the received DCT coefficients and then perform inference, e.g., image classification, using a CNN model. The performance metric of effectiveness decoding is inference accuracy. Images containing artifacts such as pixelation and color distortion, may cause inference accuracy loss. We have observed through experiments that a conventional CNN model trained on the original images often performs poorly, i.e. inference accuracy deteriorates, when an input image has significant distortion. To meet stringent deadline of time-critical IoT applications, IoT device often needs to compress image, furthermore, it is not viable to rely on retransmission or advanced FEC in edge computation offloading. Therefore, we aim at training a Robust CNN model for effectiveness decoding that can achieve high accuracy, regardless of the received image quality.

We also realized that the reason of inference accuracy loss under image distortion lies in that a conventional CNN model always expects an input image of the original training image quality, consequently, the filters and parameters learned in the training struggle to recognize the same pattern in an input image even though the image has different degrees of distortions. To address this issue, during the training phase a Robust CNN model should learn more filters for the same class and even for the same image, having different degrees of distortions. In this way, the new Robust CNN model will learn new filters associated with the same class increasing the probability of a correct classification. This is possible because neural network, and especially CNN, consists of millions of parameters and so it can learn multiple representations of the same pattern within a single CNN model. Based on these hypotheses, we implement a new training technique that involves the use of imperfect images, as shown in Fig. 3. To save the time needed to generate each image with high diversity in the data set, we combine our new training technique with some standard image augmentation techniques. At each epoch of the CNN parameters optimization process, we augment the training and validation sets as follows. (i) Each image is transformed to its DCT representation. (ii) Each DCT coefficient can get lost with a given probability. (iii) Images are reconstructed by using the corrupted DCT coefficients.

The image augmentation process generates a large number of reconstructed images. The generated images are further augmented using standard geometric transformation (rotation and scaling) and used to update the CNN model parameters. In each epoch of the training process, the data enhancement process is repeated to prevent the model from overfitting image distortion in the data generated by each epoch, and to improve the inference accuracy when severely distorted images appear.

III. PERFORMANCE EVALUATION

In this section we systematically studied the impact of distortion caused by missing DCT coefficients. Then we compare the inference accuracy of the proposed Edge Intelligence using semantic communication and the conventional approach (i.e., the conventional offloading using the original CNN model at Edge) under different image distortions, as well as under
Inference accuracy under lossy compression

To reduce the communication latency in computational offloading, reducing the image size by compression is a straightforward approach. In this way, it is possible to trade off inference accuracy and latency. The standard JPEG uses quality factor $Q$ to compress images by applying Quantization Matrix to the matrix of DCT coefficients. The quantization matrix follows the same principle that applying higher quantization for the less important DCT coefficients. Therefore, after quantization some of the DCT coefficients at high frequency will become zeros. The compressed image has similar data representation as our proposed compression method in Subsection II-A, the difference is that for JPEG the $Q$ factor needs to be selected before encoding starts whereas our offloading scheme allows encoding in the air.

Fig. 5 shows the inference accuracy of our proposed Robust CNN models and their original models on distorted images under varying quality factor $Q$. It can be seen that although our proposed Robust CNN models are trained on different types of images, they perform better than the original models especially when the images are highly compressed. For example, when $Q = 10$, Robust ResNet50 improve accuracy of 22.8% (i.e., $0.705 - 0.477$) for flower dataset and 21.3% (i.e., $0.388 - 0.175$) for Caltech 256. This shows that our proposed Robust CNN models not only performs better for images with few DCT coefficients but also for highly compressed images with severe quantization errors. Table II summarizes the average inference accuracy of Robust MobileNet-V2 and Robust ResNet-50, and their original models on distorted images under varying $Q$. We can see that Robust MobileNet-V2 and Robust ResNet-50 outperform their original models for both flower dataset and Caltech 256. For example, Robust ResNet50 improves average accuracy by 4.3% and 6.2% for Flower dataset and Caltech 256, respectively.

\[
\begin{array}{cccc|cccc}
\text{Flower} & \text{Caltech 256} \\
\hline
\text{Original} & \text{Robust} & \text{Diff.} & \text{Original} & \text{Robust} & \text{Diff.} \\
\hline
\text{MobileNet-v2} & 0.705 & 0.875 & +0.170 & 0.498 & 0.724 & +0.226 \\
\text{ResNet-50} & 0.717 & 0.896 & +0.179 & 0.503 & 0.756 & +0.253 \\
\end{array}
\]

Table I summarizes the average inference accuracy of Robust MobileNet-V2 and Robust ResNet-50, and their original models on distorted images (i.e. with top-$n$ DCT coefficients, $n$ varying from 1 to 64). We can see that under image distortion the effectiveness decoding using Robust MobileNet-V2 and Robust ResNet-50 achieves inference accuracy improvement of 17% and 17.9% on the flower dataset and 22.6% and 25.3% on the caltech 256, respectively. In general, the performances of Robust CNN models are much better when the images are heavily distorted (e.g. $n < 20$), and the inference accuracy difference becomes less significant when the images have little distortion. This study illustrates the rationale behind the design of channel-agnostic effectiveness encoding for offloading. It means under stringent latency and data rate constraints, even if an IoT device is only able to transmit the top-$n$ DCT coefficients ($n$ can be very small, causing severe image distortion), the Robust CNN models are still able to achieve decent inference accuracy. The detailed results will be presented in Subsection III-E.
achieves an average inference accuracy of 89.7% (i.e. using Robust CNN models. For example, Robust ResNet-50 to 5. We can see a similar performance improvement by MobileNet-V2 and ResNet-50 under packet losses from 0 to 5 packets. It can be seen that the effectiveness decoding using Robust CNN models, when 0 to 5 packets are lost. It is measured the inference accuracy when each image randomly loses 1 to 5 packets. Fig. 6 compares the inference accuracy of Robust CNN models and their original models, when 0 to 5 packets are lost. It can be seen that the effectiveness decoding using Robust CNN models achieves higher accuracy than the original models. Table III summarizes the average inference accuracy of the Robust CNN models and their original models for MobileNet-V2 and ResNet-50 under packet losses from 0 to 5. We can see a similar performance improvement by using Robust CNN models. For example, Robust ResNet-50 achieves an average inference accuracy of 89.7% (i.e. $(0.927 + 0.904 + 0.909 + 0.899 + 0.888 + 0.855)/6$) on flower dataset and 74.0% on Caltech 256 (i.e. $(0.817 + 0.776 + 0.755 + 0.722 + 0.704 + 0.669)/6$), which are 10.4% and 22.0% higher than that of original models.

C. Inference accuracy under packet losses

In this experiment, we study the impact of packet losses. One packet loss may cause one or more DCT coefficients losses, depending on the packet size and the amount of data of a DCT coefficient. Note that the higher order DCT coefficients have smaller values and hence are represented by fewer bits. Assuming that packet size is of 1024 bytes, which is approximately half of the data size of the 1st DCT coefficients in all blocks, namely, the 1st DCT coefficients of all blocks in an image can be transmitted in two packets. In the experiment, we measured the inference accuracy when each image randomly loses 1 to 5 packets. Fig. 6 compares the inference accuracy of Robust CNN models and their original models, when 0 to 5 packets are lost. It can be seen that the effectiveness decoding using Robust CNN models achieves higher accuracy than the original models. Table III summarizes the average inference accuracy of the Robust CNN models and their original models for MobileNet-V2 and ResNet-50 under packet losses from 0 to 5. We can see a similar performance improvement by using Robust CNN models. For example, Robust ResNet-50 achieves an average inference accuracy of 89.7% (i.e. $(0.927 + 0.904 + 0.909 + 0.899 + 0.888 + 0.855)/6$) on flower dataset and 74.0% on Caltech 256 (i.e. $(0.817 + 0.776 + 0.755 + 0.722 + 0.704 + 0.669)/6$), which are 10.4% and 22.0% higher than that of original models.

D. Inference accuracy under bit errors

During computation offloading, bit channel errors can occur. In this experiment, we study how bit errors and different effectiveness encoding strategies influence inference accuracy. First we tested Robust CNN models and the original models on images with uniformly distributed bit errors under different bit error probabilities. As shown in Fig. 7, the inference accuracy decreases as the bit error probability increases. The effectiveness decoding using Robust CNN models achieves higher inference accuracy (can also be seen in Table IV).

To improve inference accuracy under random bit errors, intuitively we can use FEC to protect the most important bits as one of the typical effectiveness encoding strategies. For binary representation, if the sign bit is flipped, it may cause significant change in values. In addition, as we know, the 1st DCT coefficients have the most important information about an image. The signs of the 1st DCT coefficients of an image are about less than 0.5% of total data size of an image. Therefore, we only need to add small amount of parity check bits, which increases transmission data size negligibly. As shown in Table IV, by protecting the signs of the 1st DCT coefficients, average inference accuracy of both the Robust models and their original models are improved by $2.6 - 7.5\%$ using the effectiveness encoding. For example, Original ResNet-50 improve the inference accuracy form 0.149 to 0.175 on Caltech 256. The effectiveness decoding Robust MobileNet-V2 and Robust ResNet-50 improve inference accuracy by $27\%$ and $28.1\%$ on flower dataset and $33.4\%$ and $34.1\%$ on Caltech 256, respectively. For example, Robust MobileNet-V2 achieves an average inference accuracy of 0.748 on flower dataset, while its original model only reaches 0.478.

To take a step further, since all the 1st DCT coefficients are about $4\%$ of the total image size, if protecting entire the 1st DCT coefficients using FEC, the increased transmission data size is still marginal; however, the inference accuracy is improved by $16.1 - 24.1\%$ on flower dataset and $33.8 - 40.7\%$ on Caltech 256 using our effectiveness encoding, as shown in Fig. 7 and Table IV. For example, the average inference accuracy of Robust MobileNet-V2 on flower dataset is improved from 0.698 to 0.861 and that of its original model is improved from 0.428 to 0.669. In general, the Robust CNN models are shown to be robust against bit errors and consistently outperform their original models.

E. Inference accuracy under latency and data rate constraints

This section studies how the proposed Edge Intelligence framework performs under latency and data rate constraints, in comparison with the conventional framework. The data rate
is assumed to be constant during task offloading. Note that the size of each image is not same even if using the same level of compression and same number of DCT coefficients. This is because the original image size depends on changes in image luminance and chrominance. We change the deadline and data rate in the range of $1 - 50$ ms and $1 - 50$ Mbps, respectively. As expected, the most critical conditions are those with low data rates and short deadlines. The simulations in this section are carried out using flower dataset in the following three scenarios.

1) No channel error: Fig. 8(a) shows the inference accuracy of the proposed Edge Intelligence framework using Robust ResNet-50 under latency and data rate constraints, when no channel error during offloading. It can be observed that the proposed framework can achieve high accuracy as long as the deadline is above $5$ ms and data rate is above $5$ Mbps. Fig. 8(b) shows that the difference in the inference accuracy of the proposed framework and that of the conventional framework is small, if an image can be completely transmitted under the given deadline and data rate; however, under the critical conditions, namely, under stringent latency ($< 15$ ms) and low data rate ($< 25$ Mbps), our proposed framework performs significantly better, achieving a gain of inference accuracy up to $40\%$ using Robust ResNet-50.

2) With channel errors: In this simulation, the bit error probability is set to $5\%$.

IV. Conclusion

In this paper, we design an robust Edge Inference framework for time-critical IoT applications using semantic communication paradigm. We propose the channel-agnostic effectiveness encoding for offloading and realize an effectiveness decoding by implementing a novel image augmentation process for CNN training which generates Robust MobileNet-v2 and Robust ResNet-50 to provide high inference accuracy for distorted images. The experimental results show that the effectiveness decoding using Robust CNN models perform consistently better than their original models under different kinds of image distortions. Thanks to the design of semantic communication, our proposed Edge Intelligence framework significantly outperforms the conventional framework under latency and data rate constraints, in particular, under extremely stringent deadlines and low data rates. In our future work, we will consider the deep learning-based image semantic coding.

References

[1] N. Li, A. Iosifidis, and Q. Zhang, “Collaborative edge computing for distributed cnn inference acceleration using receptive field-based segmentation,” Computer Networks, vol. 214, p. 109150, 2022.

[2] ——, “Graph reinforcement learning-based cnn inference offloading in dynamic edge computing,” in IEEE GLOBECOM 2022, pp. 982–987.

[3] Z. Weng and Z. Qin, “Semantic communication systems for speech transmission,” IEEE Journal on Selected Areas in Communications, vol. 39, no. 8, pp. 2434–2444, 2021.

[4] N. Li, A. Iosifidis, and Q. Zhang, “Attention-based feature compression for cnn inference offloading in edge computing,” arXiv, vol. abs/2211.13745, 2023.

[5] J. Xiao, L. Aggarwal, P. Banerjee, M. Aggarwal, and G. Medioni, “Identity preserving loss for learned image compression,” in IEEE/CVF CVPR Workshops, June 2022, pp. 517–526.

[6] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 2, pp. 210–227, 2009.

[7] X. Shen, J. Yang, C. Wei, B. Deng, J. Huang, X.-S. Hua, X. Cheng, and K. Liang, “Dct-mask: Discrete cosine transform mask representation for instance segmentation,” in IEEE/CVF CVPR, June 2021, pp. 8720–8729.

[8] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” in IEEE/CVF CVPR, 2018, pp. 4510–4520.

[9] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in IEEE/CVF CVPR, 2016, pp. 770–778.

[10] A. Mamaev, “Flowers dataset,” 2021. [Online]. Available: https://public.roboflow.com/classification/

[11] G. Griffin, A. Holub, and P. Perona, “Caltech 256,” 2022. [Online]. Available: https://data.caltech.edu/records/20087

[12] N. Reynders and W. Dehaene, “Jpeg encoder,” in Ultra-Low-Voltage Design of Energy-Efficient Digital Circuits. Springer, 2015, pp. 141–170.