SELECTIVE SENSING: A DATA-DRIVEN NONUNIFORM SUBSAMPLING APPROACH FOR COMPUTATION-FREE COMPRESSIVE INFORMATION ACQUISITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Designing on-sensor compression scheme for efficient information acquisition has always been a challenging task. Compressive sensing is a state-of-the-art sensing scheme used for on-sensor compression. However, the undesired computational complexity involved in the sensing stage of compressive sensing limits its practical application in resource-constrained sensor devices or high-data-rate sensor devices dealing with high-dimensional signals. In this paper, we propose a selective sensing framework that adopts the novel concept of data-driven nonuniform subsampling for acquiring signal information in a compressive and computation-free fashion. Selective sensing adopts a co-optimization methodology to co-train a selective sensing operator with a subsequent information decoding neural network. We take image as the sensing modality and reconstruction as the information decoding task to demonstrate the 1st proof-of-concept of selective sensing. The experiment results on CIFAR10, Set5 and Set14 datasets show that selective sensing can achieve an average reconstruction accuracy improvement in terms of PSNR/SSIM by 3.73dB/0.07 and 9.43dB/0.16 over compressive sensing and uniform subsampling counterparts across the compression ratios of 4-32x, respectively.

1 INTRODUCTION

In the era of Internet-of-things (IoT) data explosion Biookaghazadeh et al. (2018), efficient information acquisition and on-sensor data compression techniques are in great need. Compressive sensing is a state-of-the-art compressive information acquisition technique that is applicable to on-sensor data compression. Compressive sensing performs the sensing and compression of a signal simultaneously Hegde & Baraniuk (2009) by performing a linear mapping of the signal onto a low-dimensional space. Due to the high computational complexity of the linear transformation ($O(n^2)$), its implementation on sensor devices requires either a high cost (implemented in the analog domain) or large computation resources (implemented in the digital domain). To mitigate this problem, several approaches have been proposed to reduce the computational complexity of sensing stage in compressive sensing by constraining the sensing matrices to be sparse, binary, or ternary Wang et al. (2016); Nguyen et al. (2017); Zhao et al. (2018); Hong et al. (2019). These approaches can reduce the computational complexity by a constant factor ($O(cn^2)$, where c can be as low as 10^{-2}). Nonetheless, the reduced computational complexity can be still too high thus hardly affordable for resource-constrained sensor devices, *e.g.*, low-cost IoT sensors Djelouat et al. (2018), or high-data-rate sensor devices dealing with high-dimensional signals, *e.g.*, LiDAR and depth map Chodosh et al. (2019).

In this paper, we propose a selective sensing framework that fundamentally addresses the above-mentioned problem. Selective sensing adopts the novel concept of data-driven nonuniform subsampling for acquiring signal information in a compressive and computation-free fashion. Specifically, the sensing process in selective sensing is a nonuniform subsampling (or selection) process that simply selects the most informative entries of a signal vector based on an optimized selection index vector informed by training data. Since no computation is involved for any form of data encoding, the computational complexity of the selective sensing operator is simply $O(1)$, leading to the computation-free acquisition of the signal information of interest.[1] Selective sensing adopts a co-optimization methodology to co-train a selective sensing operator with a subsequent information decoding neural network. As the trainable parameters of the sensing operator (the selection index) and the information decoding neural network are discrete- and continuous-valued, respectively, the co-optimization problem in selective sensing is a mixed discrete-continuous optimization problem that is inherently difficult to solve. We propose a feasible solution to solve it by transforming the mixed discrete-continuous optimization problem into two continuous optimization subproblems through interpolation and domain extension techniques. Both of the subproblems can be then efficiently solved using gradient-descent-based algorithms. We take image as the sensing modality and reconstruction as the information decoding task to demonstrate the 1st proof-of-concept of selective sensing. The experiments on CIFAR10, Set5 and Set14 datasets show that the selective sensing framework can achieve an average reconstruction accuracy improvement in terms of PSNR/SSIM by 3.73dB/0.07 and 9.43dB/0.16 over compressive sensing and uniform subsampling counterparts across the compression ratios of 4-32x, respectively.

The contributions of this paper are summarized as follows:

1. We propose a new information acquisition method—selective sensing. To the best of our knowledge, this is the first work that proposes a computation-free sensing operator for efficiently acquiring signal information of interest in a compressive format. The computation-free nature of selective sensing makes it a highly suitable solution for performing compressive information acquisition on resource-constrained sensor devices or high-data-rate sensor devices dealing with high-dimensional signals.

2. We propose and apply the novel concept of data-driven nonuniform subsampling. Specifically, we first formulate the problem of co-optimizing a selective sensing operator with a subsequent information decoding neural network as a mixed discrete-continuous optimization problem. Furthermore, we propose a viable solution that transforms the problem into two continuous optimization subproblems that can be efficiently solved by gradient-descent-based algorithms, which makes the co-training feasible.

3. We empirically show that data-driven nonuniform subsampling can well preserve signal information under the presence of the co-trained information decoding network.

2 Related Work

2.1 Model-based Nonuniform Subsampling

Model-based nonuniform subsampling has been proposed in Chepuri et al. (2016) in the name of sparse sensing. It should be noted that there is a vast difference between sparse sensing and selective sensing. Sparse sensing requires a hand-crafted sparsity model of a signal as a prior knowledge. Consequently, the process to learn a model of a signal from data is not needed. Differently, selective sensing requires no prior knowledge about the sparsity model of a signal as all the necessary information needed for reconstruction can be learned from data through the training process. Therefore, selective sensing has a much broader range of applications, especially in IoT, than sparse sensing, considering a vast majority of IoT signals are not well studied nor understood yet, but huge amount of IoT data are already available for training and learning.

2.2 Sensing Matrix Simplification Methods

Various approaches have been proposed to reduce the computational complexity of the sensing stage in compressive sensing. The computational complexity of the linear transformation in compressive sensing is $O(n^2)$. Some recent work Zhao et al. (2018); Hong et al. (2019) proposes model-based methods to construct sparse sensing matrices. Wang et al. (2016); Nguyen et al. (2017) propose data-driven methods to build binary or ternary sensing matrices. However, all these approaches

[1] From the hardware implementation perspective, the selection operation can be simply implemented in the digital domain with a counter and a mux that already exists in the control logic of most sensors. We consider such operations as control rather than computation as no data needs to be computed during the process. For spatial signals such as images, the selective sensing operator can be also implemented as a low-cost masked sensor array with no computation involved.

could only reduce the computational complexity by constant factors, i.e. $O(cn^2)$, where c can be as low as 10^{-2}). A key differentiator of selective sensing is that its sensing stage adopts the novel concept of data-driven non-uniform subsampling, while is computation-free and has a computational complexity of $O(1)$.

2.3 Data-driven Compressive Sensing

Several approaches Kulkarni et al. (2016); Mousavi & Baraniuk (2017); Yao et al. (2019) propose to directly learn the inverse mapping of compressive sensing through the training of reconstruction neural network models. In addition, Mousavi et al. (2017; 2018); Wu et al. (2018) propose to co-train a customized sensing scheme with a reconstruction neural network to improve the reconstruction accuracy. These approaches inspired us to develop a framework that co-trains a selective sensing operator and a subsequent information decoding network to acquire signal information in a compressive format while keeping the sensing process computation-free.

3 Methodology

In this section, we first formulate the co-optimization of a selective sensing operator and a subsequent information decoding network as a mixed discrete-continuous optimization problem. Then, by applying continuous interpolation and domain extension on the integer variables, we reformulate the mixed discrete-continuous optimization problem into two continuous optimization problems, both of which can be solved by conventional gradient-descent-based algorithms. Based on the new formulation, we extend the conventional backpropagation(BP) algorithm to derive a general co-training algorithm to co-optimize a selective sensing operator and a subsequent information decoding network. At last, by taking images as the sensing modality and using reconstruction as the information decoding task, we propose a practical approach, referred to as SS+Net, to compose a selective sensing framework for image selective sensing and reconstruction.

In this paper, a lowercase letter denotes a scalar or a scalar-valued function, and a uppercase letter denotes a vector, a matrix, a tensor, or a vector-valued function. We use brackets to index the element of a vector, a matrix, or a tensor. For example, assume X denotes a n-dimensional vector $X = [x_0, ..., x_{n-1}]$, then $X[i] = x_i$ for $i = 0, \cdots, n-1$.

3.1 Problem Formulation

Consider the original signal X is an n-dimensional vector, the subsampling rate is $\frac{m}{n}$, and the subsampled measurement Y is a m-dimensional vector. The selective sensing of X is a nonuniform subsampling or a selection process that can be formulated as

$$Y = S(X, I) = [X[I[0]], \cdots, X[I[m-1]]], \tag{1}$$

where $S(X, I)$ is a function that stands for the selective sensing operator. I is a m-dimensional vector denoting the selection set, which contains the indices (integer values between 0 and $n-1$) of the elements to be selected. Consider $N(Y, \Theta)$ is a subsequent information decoding network and Θ is the trainable parameters. o is a differentiable objective function that measures the information loss throughout the entire selective sensing process with respect to a information acquisition task. For instance, in a signal reconstruction task, the objective function can be defined as a loss function which measures the difference between the reconstructed signal and the original signal. The co-optimization problem of the sensing operator S and the information decoding network N can be formulated as

$$\begin{gathered} I_{opt}, \Theta_{opt} = \arg\min_{I,\Theta} o(N(S(X, I), \Theta)), \\ \text{subject to } i_0, \ldots, i_{m-1} \text{ are integers within interval } [0, n-1] \end{gathered} \tag{2}$$

Given the entries of Θ are continuous variables, and the entries of I are constrained to be integer variables within $[0, n-1]$, the problem in (2) is a mixed discrete-continuous optimization problem that can not be directly solved with conventional gradient-descent-based algorithms. This is because the gradient of o with respect to I does not exist.

3.2 Reformulation by Continuous Interpolation and Domain Extension

By applying the continuous interpolation on S with respect to I and the extension on the domain of S, we can reformulate the problem in (2) into two subproblems. For simplicity, we adopt a linear interpolation in our method. However, nonlinear interpolation methods can be also applied.

We define a piece-wise linear function $f(X, i)$ as

$$\begin{gathered} f(X,i) = (X[r_u] - X[r_d])(i - i_d) + X[r_d], \\ where\ i_u = floor(i) + 1,\ i_d = floor(i),\ r_u = i_u \bmod n \text{ and } r_d = i_d \bmod n. \end{gathered} \tag{3}$$

In (3), i is a real-valued scalar, $floor()$ is the flooring function returning the closest integer that is less than or equal to the input, and mod is the modulo operation. $f(X, i)$ essentially interpolates the value of X over a continuous index i in a piece-wise linear fashion and extends the range of i to $(-\infty, \infty)$. Given a X, $f(X, i)$ is periodic over every n-length interval of i. At integer values of i, we have $f(X, i) = X[i \bmod n]$, which returns the original value of the $[i \bmod n]$-th element of X. Specifically, when i is an integer in interval $[0, n-1]$, we have $f(X, i) = X[i]$. Due to the continuous interpolation and domain extension, $f(X, i)$ is almost everywhere differentiable over i except for all the integer points. The choice of the gradient value at integer points turn out to be insensitive to the algorithm performance. For simplicity, we define the derivatives of $f(X, i)$ at integer values of i as zero. As such, we have the gradient value of f with respect to i in the whole space which can be expressed as

$$\frac{\partial f}{\partial i} = \begin{cases} 0, & \text{if } i \text{ is a integer,} \\ (X[r_u] - X[r_d]), & \text{otherwise.} \end{cases} \tag{4}$$

Based on (4), we define a continuous selective sensing operator function S' as

$$S'(X, I) = [f(X, i_0), ..., f(X, i_{m-1})]. \tag{5}$$

Leveraging (3) and (5), we reformulate the mixed discrete-continuous optimization problem in (2) into two subproblems defined as

$$I_R, \Theta_R = \arg\min_{I,\Theta} o(N(S'(X, I), \Theta)), \tag{6}$$

and

$$\begin{gathered} I_{opt} = [round(i) \bmod n \text{ for each entry } i \text{ in } I_R], \\ \Theta_{opt} = \arg\min_{\Theta} o(N(S(X, I_{opt}), \Theta)), \end{gathered} \tag{7}$$

where $round()$ is an even rounding function that returns the closest integer value of the input, and the initial values of Θ in (7) is Θ_R. Note that both the subproblems in (6) and (7) are unconstrained and the gradient of o with respect to I and Θ can be calculated over the whole space in (6). Therefore, we can solve the subproblems in (6) and (7) sequentially using gradient-descent based algorithms. For the brevity of illustration, we refer to the process of solving the subproblem in (6) and (7) as the initial-training and the fine-tuning step, respectively, in the rest of this paper.

3.3 Extension of the Backpropagation Algorithm

Generally, neural network models are trained over multiple training samples and the gradients of trainable parameters are calculated using the BP algorithm. We extend the BP algorithm and derive the gradient calculation (with respect to I) over a batch of training samples as follows.

Given a batch of b samples $X_1, \cdots, X_b$ of the signal X for training, the forward pass of the BP algorithm in the initial-training step can be derived as

$$Y_i = S'(X_i, I),\ Z_i = N(Y_i, \Theta),\ o_i = o(Z_i),\ o_{batch} = \frac{1}{b}\sum_{i=1}^{b} o_i, \tag{8}$$

where $i = 1, \cdots, b$, Z_i is the representation of the information decoded by the network and o_{batch} is the loss function that measures the average information loss throughout the selective sensing process. One can also choose to use the total information loss here.

Algorithm 1 Main algorithm

```
**Input:** training samples $X_1, \cdots, X_N$, number of iterations $maxiters$ for initial-training step,
batch size $b$
Initialize $I, \Theta$
**Initial-training**
**for** $iter = 1$ **to** $maxiters$ **do**
    **for** $batch = 1$ **to** $\frac{N}{b}$ **do**
        **Forward pass**
        Execute (8)
        **Backward pass**
        Using the BP algorithm to calculate the gradient with respect to $\Theta$
        Execute (9)
        Execute (10) to calculate the gradient with respect to $I$
        Optimize $I$ and $\Theta$ using the calculated gradients
    **end for**
**end for**
$I_R, \Theta_R = I, \Theta.$
Execute the rounding and modulo operations over entries of $I_R$ as in (7) to get $I_{opt}$.
**Fine-tuning**
Initialize $\Theta = \Theta_R$, further optimize $\Theta$ with the gradients calculated by the BP algorithm while
keeping $I = I_{opt}$
**End**
$I_{opt}, \Theta_{opt} = I, \Theta$
```

In the backward pass of the BP algorithm, the gradient calculation with respect to Θ is the same as in regular neural network training. The gradient calculation with respect to I can be derived using the chain rule of derivative. Specifically, the gradient calculation of o_{batch} with respect to Y_i can be derived as

$$\frac{\partial o_{batch}}{Y_i} = \frac{1}{b} \frac{\partial o_i}{Y_i} \text{ for } i = 1, \cdots, b. \tag{9}$$

Subsequently, the gradient calculation of o_{batch} with respect to I over a batch of training samples can be derived as

$$\begin{aligned} \frac{\partial o_{batch}}{\partial I} &= \frac{1}{b} \sum_{j=1}^{b} \frac{\partial o_j}{\partial Y_j} \frac{\partial Y_j}{\partial I} \\ = [\frac{1}{b} \sum_{j=1}^{b} \frac{\partial o_j}{\partial Y_j[0]} \frac{\partial f(X_j, I[0])}{\partial I[0]}, &\cdots, \frac{1}{b} \sum_{j=1}^{b} \frac{\partial o_j}{\partial Y_j[m-1]} \frac{\partial f(X_j, I[m-1])}{\partial I[m-1]}]. \end{aligned} \tag{10}$$

Leveraging the gradient calculations in (9) and (10), the subproblem in (6) can be therefore solved by using gradient-descent-based algorithms. The outputs from the initial-training step include the optimized selection set I_R and the corresponding reconstruction network parameters Θ_R. As the entries of I_R are continuous over interval $(-\infty, \infty)$, one needs further convert I_R to an integer selection set I_{opt} as shown in (7). To compensate for the accuracy loss due to rounding, the reconstruction network shall be further fine tuned in the fine-tuning step while keeping I_{opt} fixed as shown in (7).

The entire algorithm of co-training a information decoding network N and a selective sensing operator S is summarized in **Algorithm** 1.

3.4 Image Selective Sensing and Reconstruction

In the rest of the paper, we take image as the sensing modality and reconstruction as the information decoding task to demonstrate the first proof-of-concept of selective sensing. The prior work discussed in subsection 2.3 shows that neural network models can be trained to directly approximate the inverse mapping of compressive sensing to perform the reconstruction. Therefore, we hypothesize that there exists a direct mapping from the selective sensed (nonuniformly subsampled) domain to the original image domain, and such a mapping can be well approximated by a neural network co-trained with the selective sensing operator. Furthermore, we hypothesize that the existing image compressive sensing reconstruction networks can be also used for image selective sensing reconstruction.

Based on our hypotheses, we use the loss function $l(\hat{X}, X)$ as the objective function in (2), where l is a function that measures the distance between $\hat{X}$ and X, *e.g.* a mean-square-error function, and $\hat{X}$ is the output (Z in (8)) of the information decoding network N. As such, N is trained to directly reconstruct the original image from the selective sensing measurement as

$$X \xrightarrow[\text{Sensing}]{S'} Y \xrightarrow[\text{Reconstruction}]{N} \hat{X}. \tag{11}$$

We refer to the image selective sensing and reconstruction frameworks composed in such way as SS+Net.

To evaluate the performance of SS+Net, we compare it against the compressive sensing and uniform subsampling counterparts referred to as CS+Net and US+Net, respectively. CS+Net and US+Net use same reconstruction networks but replace the selective sensing operator in SS+Net with a Gaussian random sensing matrix and a uniform subsampling operator, respectively. Additionally, we set all the hyper-parameters in SS+Net, CS+Net, and US+Net to be the same during the training for a fair comparison. The purpose of using CS+Net and US+Net as the reference methods is to reveal the true impact of selective sensing on compressive information acquisition in comparison to the compressive sensing and uniform subsampling counterparts.

4 Experiments

4.1 Experiment Setup

We conduct experiments on two datasets with two different reconstruction networks at the measurement/subsampling rates ranging from 0.03125 to 0.25 (corresponding to the compression ratios of 32-4x). The first dataset is CIFAR10 Krizhevsky et al. (2009). The second dataset is composed in the same way as illustrated in Xu et al. (2018), which has 228,688 training samples and 867 testing samples. The training samples are the patches of the augmented (rotation and flip) images from Arbelaez et al. (2010) and Yang et al. (2010). The testing samples are the non-overlapped image patches from Set5 Bevilacqua et al. (2012) and Set14 Zeyde et al. (2010). All the samples are of size 64x64 with three color channels(RGB). Therefore, the sensing is performed channel-wise in all the experiments, i.e. for each framework of SS-Net, CS-Net and US-Net, there will be three sensing operators in parallel corresponding to three color channels and the sensed measurements from three color channels are grouped together then fed into the reconstruction network. In both training sets, 5% of the training samples are randomly selected and separated out prior to the training as the validation set.

The two reconstruction networks we experimented with are DCNet and ReconNetKulkarni et al. (2016). DCNet is the generator network of DCGAN Radford et al. (2015). As ReconNet is designed to reconstructs grayscale images, we made some modifications to the network structure of ReconNet in order to reconstruct color images. Specifically, we replace the first layer of ReconNet, which is a fully-connected layer, with three fully-connected layers in parallel to generate the feature map corresponding to each of the three color channels. In addition, we modify the third and the sixth convolutional layers to have three convolution kernels each in order to maintain the dimensionality of the original image. The batch normalization layers are also added right behind each convolutional layer (except for the last convolutional layer which is the output layer) to accelerate the training.

In the training process of SS+Net, we co-train the selective sensing operator and the reconstruction network for 300 iterations, of which the first 150 iterations are used for the initial-training step and the rest are used for the fine-tuning step. The models are stored and tested on the validation set at the end of each training iteration. The stored model with the highest validation performance is used for testing. We use two different optimizers to optimize different components of SS+Net: An Adam optimizer with a learning rate of 0.001 is used to optimize reconstruction networks and A SGD optimizer with a learning rate of 100000 is used to optimize sensing operators. Using a high learning rate for training the selective sensing operators is because the gradient values with repsect to index variables I turns out to be orders of magnitude smaller than the the rest of the gradient

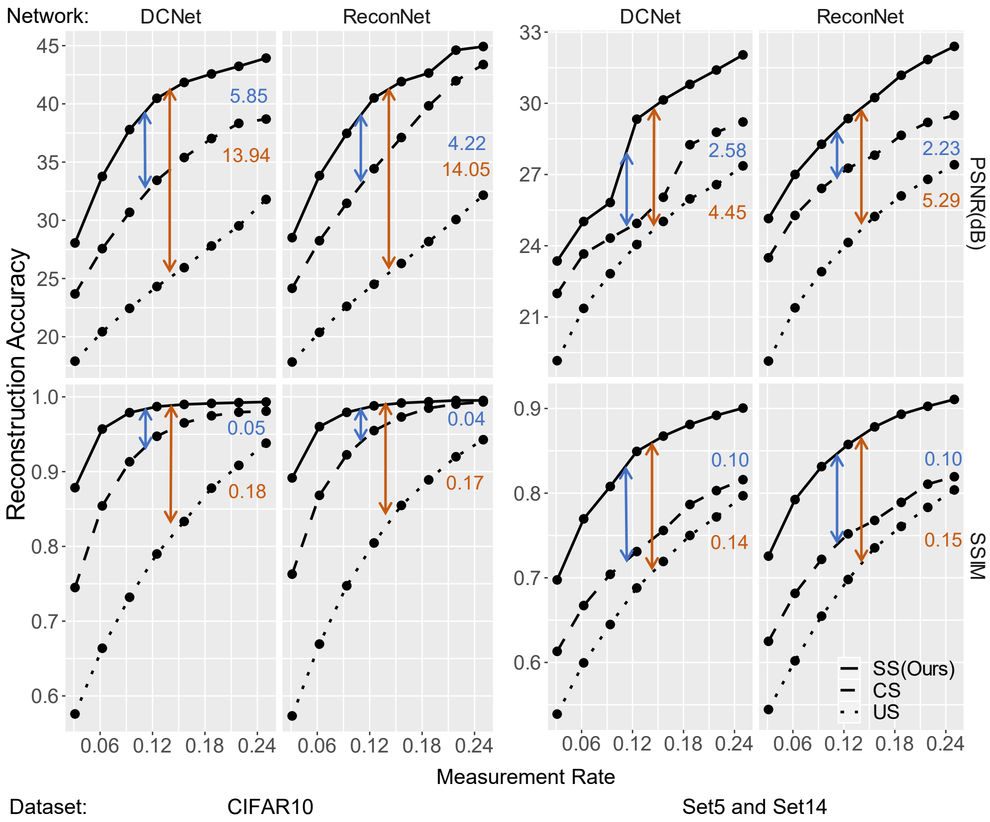

Figure 1: Comparison of information acquisition performance among selective sensing (SS), compressive sensing (CS), and uniform subsampling (US) measured in PSNR and SSIM. The results of different combinations of reconstruction networks(DCNet and ReconNet) and datasets(CIFAR10, Set5 and Set14) are plotted in different columns. The average PSNR and SSIM improvements of SS over CS and US across all eight measurement rates are annotated on the figure.

values and the learning rate of 100000 performs well in the experiments. For the training of the CS+Net and US+Net counterparts, except that there is no optimizer for sensing operators, all the other experiment setups remain the same with SS+Net.

4.2 Results and Analysis

The reconstruction accuracy is measured as the average reconstruction PSNR and SSIM over all the testing cases. The experiment results of PSNR and SSIM are plotted in Figure 1. As shown in Figure 1, selective sensing achieves up to 44.92dB/0.9952 reconstruction PSNR/SSIM at the measurement rate of 0.25 (compression ratio of 4). Even at the low measurement rate of 0.03125 (compression ratio of 32), selective sensing still achieves at least 23.35dB/0.6975 reconstruction PSNR/SSIM. The experiment results validate our hypothesis that the direct mapping from the selective sensed domain to the original image domain can be well approximated by existing reconstruction neural networks co-trained with the selective sensing operator, and data-driven nonuniform subsampling can well preserve signal information under the presence of the co-trained information decoding neural network. Furthermore, the experiment results show that selective sensing consistently outperforms compressive sensing and uniform subsampling, especially at higher compression ratios. The average PSNR/SSIM improvement of selective sensing over compressive sensing and uniform subsampling across all the experiments is 3.73dB/0.07 and 9.43dB/0.16, respectively. As the only difference between SS+Net, CS+Net, and US+Net is the sensing operator used, the experiment results imply that selective sensing better preserves signal information than compressive sensing and uniform subsampling as a result of the co-optimization of the sensing and reconstruction

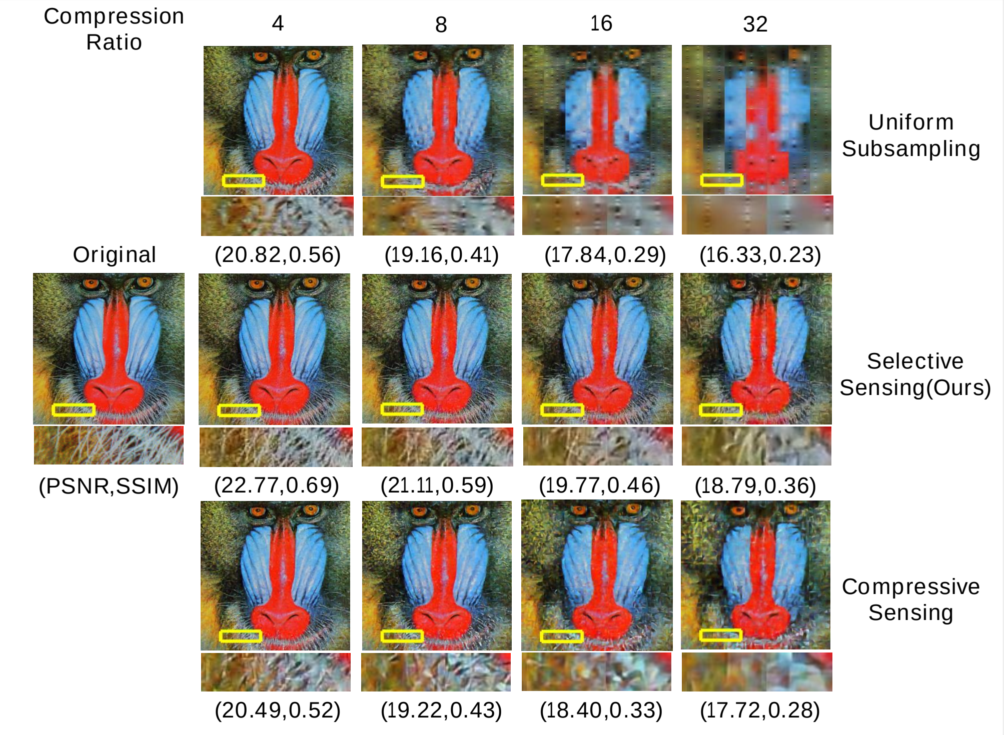

Figure 2: Visual reconstruction quality comparison among selective sensing, compressive sensing and uniform subsampling. The reconstruction network used is ReconNet, and the example image is from the Set14. Across all the compression ratios, selective sensing produces a sharper image with finer details presented, *e.g.* the beard and the skin textures.

stages. The detailed numerical PSNR and SSIM results are presented in Table 1, Table 2, Table 3 and Table 4 in the appendix.

Compared with the images in CIFAR10, the images in Set5 and Set14 have more details, which makes the reconstruction of the images in Set5 and Set14 inherently more difficult. We take one image from Set14 as an example to illustrate the visual reconstruction quality comparison. As shown in Figure 2, selective sensing reconstructs the image with finer and sharper details, such as the beard and the textures of the skin, than compressive sensing and uniform subsampling across all compression ratios. More visual reconstruction quality comparison showing the same results are illustrated in Figure 3, Figure 4 and Figure 5 in the appendix. These visual reconstruction quality comparisons show strong evidence that selective sensing better preserves signal information than compressive sensing and uniform subsampling.

5 Conclusion

In this paper, we propose a selective sensing framework that adopts the novel concept of data-driven nonuniform subsampling for acquiring signal information in a compressive and computation-free fashion. Selective sensing adopts a co-optimization methodology to co-train a selective sensing operator with a subsequent information decoding neural network. The co-training of selective sensing is first formulated as a mixed-discrete-continuous optimization problem. By applying continuous interpolation and domain extension to the sensing index domain with quantization and fine-tuning techniques, we reformulate the problem into two continuous optimization subproblems that can be solved by gradient-descent-based algorithms. This is the key to enabling the co-training of the selective sensing operator with the subsequent information decoding neural network. The experiments of image selective sensing empirically show that data-driven nonuniform subsampling can well preserve signal information under the presence of the co-trained information decoding neural network. The experiments on CIFAR10, Set5, and Set14 datasets show that the proposed selective sensing framework can achieve an average reconstruction accuracy improvement in terms of PSNR/SSIM of 3.73dB/0.07 and 9.43dB/0.16 over compressive sensing and uniform subsampling counterparts across the compression ratios of 4-32x, respectively. The computation-free nature of selective sensing makes it a highly suitable solution for performing compressive information acquisition on resource-constrained sensor devices or high-data-rate sensor devices dealing with high-dimensional signals.

REFERENCES

Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):898–916, 2010.

Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012.

Saman Biookaghazadeh, Ming Zhao, and Fengbo Ren. Are fpgas suitable for edge computing? In *{USENIX} Workshop on Hot Topics in Edge Computing (HotEdge 18)*, 2018.

Sundeep Prabhakar Chepuri, Geert Leus, et al. Sparse sensing for statistical inference. *Foundations and Trends® in Signal Processing*, 9(3–4):233–368, 2016.

Nathaniel Chodosh, Chaoyang Wang, and Simon Lucey. Deep convolutional compressed sensing for lidar depth completion. In *Computer Vision – ACCV 2018*, pp. 499–513, 2019.

Hamza Djelouat, Abbes Amira, and Faycal Bensaali. Compressive sensing-based iot applications: A review. *Journal of Sensor and Actuator Networks*, 7(4), 2018.

Chinmay Hegde and Richard G Baraniuk. Compressive sensing of streams of pulses. In *2009 47th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 44–51. IEEE, 2009.

Tao Hong, Xiao Li, Zhihui Zhu, and Qiuwei Li. Optimized structured sparse sensing matrices for compressive sensing. *Signal Processing*, 159:119–129, 2019.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Kuldeep Kulkarni, Suhas Lohit, Pavan Turaga, Ronan Kerviche, and Amit Ashok. Reconnet: Non-iterative reconstruction of images from compressively sensed measurements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 449–458, 2016.

Ali Mousavi and Richard G Baraniuk. Learning to invert: Signal recovery via deep convolutional networks. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 2272–2276. IEEE, 2017.

Ali Mousavi, Gautam Dasarathy, and Richard G Baraniuk. Deepcodec: Adaptive sensing and recovery via deep convolutional neural networks. *arXiv preprint arXiv:1707.03386*, 2017.

Ali Mousavi, Gautam Dasarathy, and Richard G Baraniuk. A data-driven and distributed approach to sparse signal representation and recovery. 2018.

Duc Minh Nguyen, Evaggelia Tsiligianni, and Nikos Deligiannis. Deep learning sparse ternary projections for compressed sensing of images. In *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 1125–1129. IEEE, 2017.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

Yuhao Wang, Xin Li, Kai Xu, Fengbo Ren, and Hao Yu. Data-driven sampling matrix boolean optimization for energy-efficient biomedical signal acquisition by compressive sensing. *IEEE transactions on biomedical circuits and systems*, 11(2):255–266, 2016.

Shanshan Wu, Alexandros G Dimakis, Sujay Sanghavi, Felix X Yu, Daniel Holtmann-Rice, Dmitry Storcheus, Afshin Rostamizadeh, and Sanjiv Kumar. Learning a compressed sensing measurement matrix via gradient unrolling. *arXiv preprint arXiv:1806.10175*, 2018.

Kai Xu, Zhikang Zhang, and Fengbo Ren. Lapran: A scalable laplacian pyramid reconstructive adversarial network for flexible compressive sensing reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 485–500, 2018.

Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma. Image super-resolution via sparse representation. *IEEE transactions on image processing*, 19(11):2861–2873, 2010.

Hantao Yao, Feng Dai, Shiliang Zhang, Yongdong Zhang, Qi Tian, and Changsheng Xu. Dr2-net: Deep residual reconstruction network for image compressive sensing. *Neurocomputing*, 359: 483–493, 2019.

Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pp. 711–730. Springer, 2010.

Wenfeng Zhao, Biao Sun, Tong Wu, and Zhi Yang. On-chip neural data compression based on compressed sensing with sparse sensing matrices. *IEEE transactions on biomedical circuits and systems*, 12(1):242–254, 2018.

A Appendix

The code, datasets and pretrained models can be downloaded from:

`https://figshare.com/s/860b61b124c92a8cb31e`.

The experiments are conducted in parallel on four RTX 2080 Ti GPU cards. One training process(300 iterations) runs on one GPU card takes around 80 minutes.

Table 1: Reconstruction performance comparison on CIFAR10 using DCNet as the reconstruction network.

Metrics		PSNR (dB)				
Sensing operator		SS	CS	US	SS over CS	SS over US
Measurement rate	0.03125	28.055576	23.678496	17.903165	4.3770796	10.152411
	0.0625	33.76353	27.567673	20.432642	6.1958566	13.330888
	0.09375	37.788893	30.687367	22.436043	7.1015266	15.352851
	0.125	40.4781	33.434791	24.314229	7.043309	16.163872
	0.15625	41.843324	35.391706	25.934857	6.4516183	15.908467
	0.1875	42.571765	37.023333	27.796602	5.5484323	14.775163
	0.21875	43.227788	38.326256	29.525659	4.9015314	13.702128
	0.25	43.928165	38.693752	31.793804	5.2344128	12.134361
Metrics		SSIM				
Sensing operator		SS	CS	US	SS over CS	SS over US
Measurement rate	0.03125	0.8785	0.745	0.5759	0.1335	0.3026
	0.0625	0.957	0.8542	0.6638	0.1028	0.2932
	0.09375	0.9789	0.9132	0.732	0.0657	0.2469
	0.125	0.987	0.9473	0.7899	0.0397	0.1971
	0.15625	0.9899	0.9652	0.8334	0.0247	0.1565
	0.1875	0.9913	0.9747	0.8779	0.0166	0.1134
	0.21875	0.9923	0.9795	0.9084	0.0128	0.0839
	0.25	0.9932	0.9809	0.9381	0.0123	0.0551

Table 2: Reconstruction performance comparison on CIFAR10 using ReconNet as the reconstruction network.

Metrics		PSNR (dB)				
Sensing operator		SS	CS	US	SS over CS	SS over US
Measurement rate	0.03125	28.510575	24.162001	17.833156	4.3485742	10.677419
	0.0625	33.835395	28.244744	20.37912	5.5906511	13.456275
	0.09375	37.465897	31.460441	22.615043	6.005456	14.850854
	0.125	40.505386	34.428422	24.507658	6.0769649	15.997729
	0.15625	41.91067	37.112229	26.295812	4.7984413	15.614858
	0.1875	42.64523	39.830833	28.173974	2.8143966	14.471256
	0.21875	44.613196	41.977892	30.068614	2.6353032	14.544581
	0.25	44.922706	43.377697	32.157417	1.5450094	12.765289
Metrics		SSIM				
Sensing operator		SS	CS	US	SS over CS	SS over US
Measurement rate	0.03125	0.8916	0.7629	0.5733	0.1287	0.3183
	0.0625	0.9602	0.8683	0.6694	0.0919	0.2908
	0.09375	0.9793	0.9225	0.7474	0.0568	0.2319
	0.125	0.988	0.9549	0.8046	0.0331	0.1834
	0.15625	0.9918	0.9731	0.8548	0.0187	0.137
	0.1875	0.9935	0.9849	0.889	0.0086	0.1045
	0.21875	0.9952	0.9903	0.9199	0.0049	0.0753
	0.25	0.9952	0.9932	0.9427	0.002	0.0525

Table 3: Reconstruction performance comparison on Set5 and Set14 using DCNet as the reconstruction network.

Metrics		PSNR (dB)				
Sensing operator		SS	CS	US	SS over CS	SS over US
Measurement rate	0.03125	23.356654	21.988134	19.158458	1.3685191	4.1981953
	0.0625	25.020666	23.655472	21.360483	1.3651944	3.6601834
	0.09375	25.821571	24.320524	22.822223	1.5010468	2.9993483
	0.125	29.330996	24.941122	24.049497	4.3898736	5.2814991
	0.15625	30.142491	26.041837	25.023534	4.1006538	5.1189574
	0.1875	30.798615	28.254278	25.973201	2.5443367	4.8254134
	0.21875	31.402867	28.782112	26.572408	2.6207543	4.8304582
	0.25	32.040132	29.214131	27.364247	2.8260015	4.6758851
Metrics		SSIM				
Sensing operator		SS	CS	US	SS over CS	SS over US
Measurement rate	0.03125	0.6975	0.6131	0.539	0.0844	0.1585
	0.0625	0.7696	0.6672	0.5995	0.1024	0.1701
	0.09375	0.8079	0.7041	0.6448	0.1038	0.1631
	0.125	0.8493	0.731	0.688	0.1183	0.1613
	0.15625	0.8674	0.7559	0.7193	0.1115	0.1481
	0.1875	0.8811	0.7866	0.75	0.0945	0.1311
	0.21875	0.8919	0.803	0.7719	0.0889	0.12
	0.25	0.9005	0.8159	0.7969	0.0846	0.1036

Table 4: Reconstruction performance comparison on Set5 and Set14 using ReconNet as the reconstruction network.

Metrics		PSNR (dB)				
Sensing operator		SS	CS	US	SS over CS	SS over US
Measurement rate	0.03125	25.14523	23.492886	19.141701	1.652344	6.0035286
	0.0625	27.002097	25.27621	21.385219	1.725887	5.6168772
	0.09375	28.276923	26.41306	22.907161	1.8638628	5.3697623
	0.125	29.357467	27.2674	24.135962	2.0900672	5.2215052
	0.15625	30.240035	27.814269	25.232525	2.4257659	5.0075104
	0.1875	31.18125	28.651692	26.099135	2.5295576	5.0821148
	0.21875	31.844453	29.195588	26.796003	2.6488653	5.0484498
	0.25	32.397313	29.493561	27.412122	2.9037527	4.9851912
Metrics		SSIM				
Sensing operator		SS	CS	US	SS over CS	SS over US
Measurement rate	0.03125	0.7256	0.625	0.5444	0.1006	0.1812
	0.0625	0.7924	0.6816	0.6019	0.1108	0.1905
	0.09375	0.8314	0.7218	0.6547	0.1096	0.1767
	0.125	0.8575	0.7519	0.698	0.1056	0.1595
	0.15625	0.8783	0.7678	0.7354	0.1105	0.1429
	0.1875	0.8932	0.7891	0.7607	0.1041	0.1325
	0.21875	0.9026	0.8106	0.7832	0.092	0.1194
	0.25	0.9107	0.8194	0.8038	0.0913	0.1069

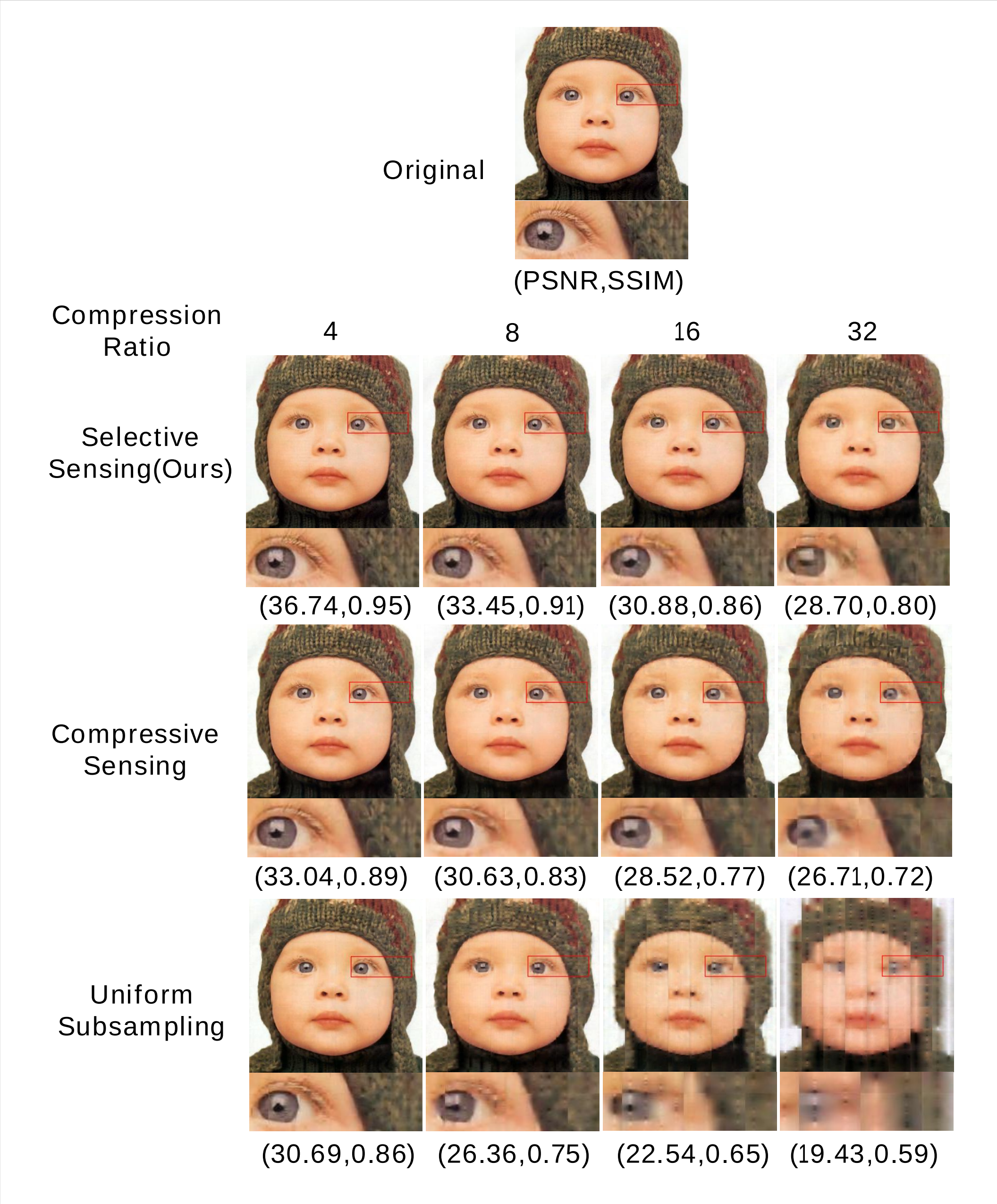

Figure 3: Visual reconstruction quality comparison among selective sensing, compressive sensing and uniform subsampling. The reconstruction network used is ReconNet, and the example image is from the Set5 dataset. Across all the compression ratios, selective sensing produces a sharper image with finer details presented, *e.g.* the eyelashes and the textures of the hat.

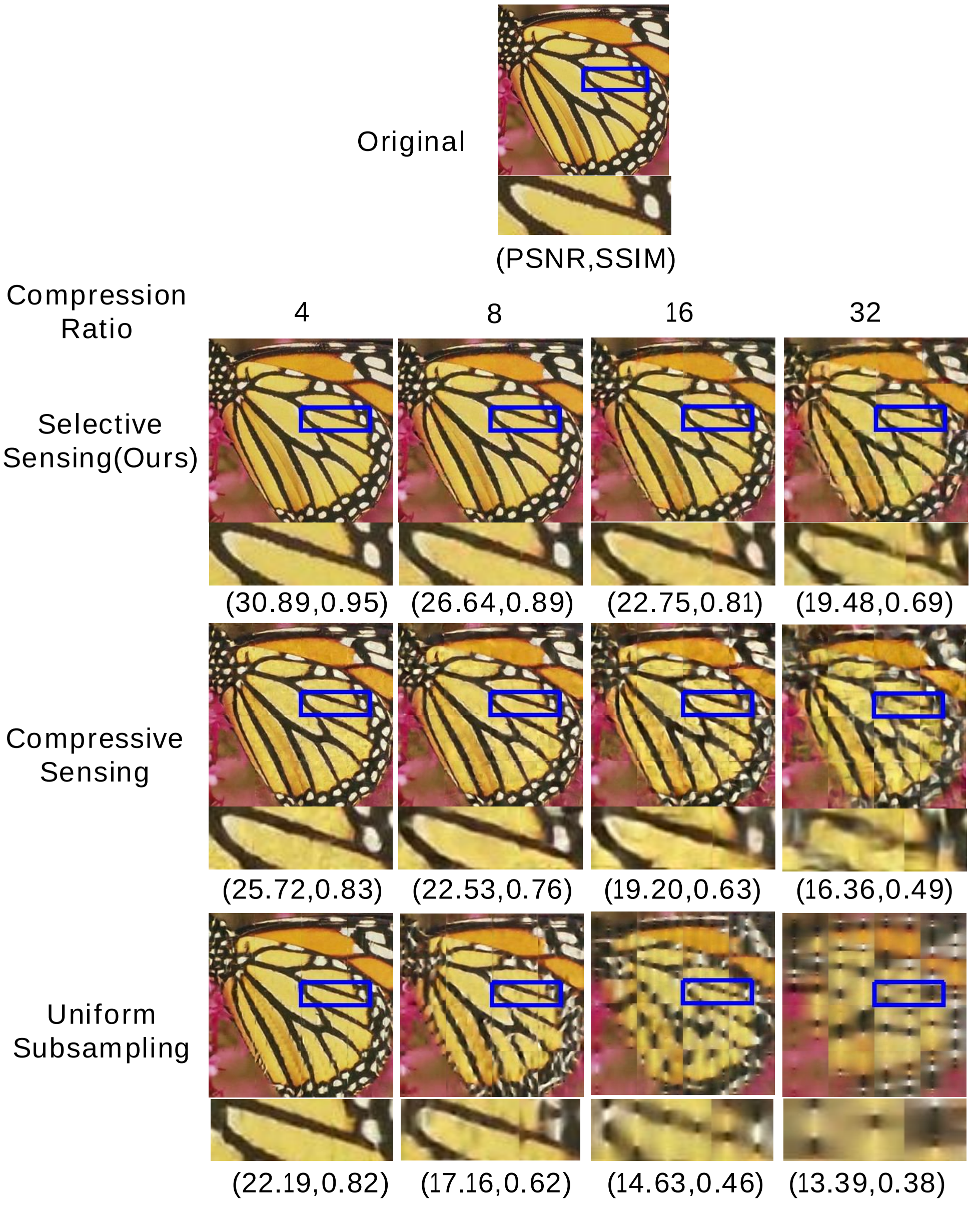

Figure 4: Visual reconstruction quality comparison among selective sensing, compressive sensing and uniform subsampling. The reconstruction network used is ReconNet, and the example image is from the Set5 dataset. Across all the compression ratios, selective sensing produces a sharper image with finer details presented, *e.g.* the textures on the wings.

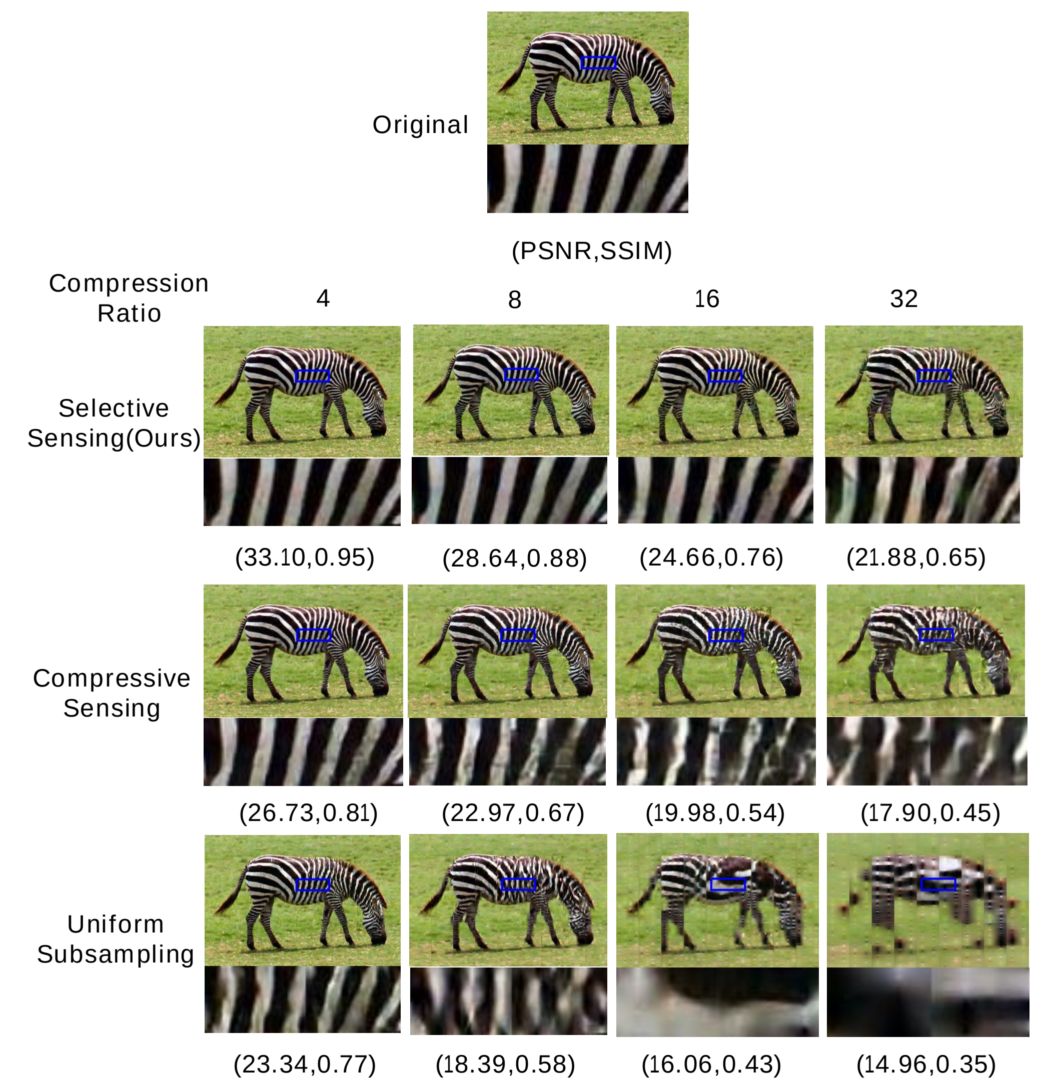

Figure 5: Visual reconstruction quality comparison among selective sensing, compressive sensing and uniform subsampling. The reconstruction network used is ReconNet, and the example image is from the Set14 dataset. Across all the compression ratios, selective sensing produces a sharper image with finer details presented, *e.g.* the edges of the stripes.