Optimization-Inspired Cross-Attention Transformer for Compressive Sensing

Jiechong Song\textsuperscript{1,4}, Chong Mou\textsuperscript{1}, Shiqi Wang\textsuperscript{2}, Siwei Ma\textsuperscript{3,4}, Jian Zhang\textsuperscript{1,4}\textsuperscript{*}

\textsuperscript{1}Peking University Shenzhen Graduate School, Shenzhen, China
\textsuperscript{2}Department of Computer Science, City University of Hong Kong, China
\textsuperscript{3}School of Computer Science, Peking University, Beijing, China
\textsuperscript{4}Peng Cheng Laboratory, Shenzhen, China

\{songjiechong, swma, zhangjian.sz\} @pku.edu.cn eechongm@stu.pku.edu.cn shiqwang@cityu.edu.hk

Abstract

By integrating certain optimization solvers with deep neural networks, deep unfolding network (DUN) with good interpretability and high performance has attracted growing attention in compressive sensing (CS). However, existing DUNs often improve the visual quality at the price of a large number of parameters and have the problem of feature information loss during iteration. In this paper, we propose an Optimization-inspired Cross-attention Transformer (OCT) module as an iterative process, leading to a lightweight OCT-based Unfolding Framework (OCTUF) for image CS. Specifically, we design a novel Dual Cross Attention (Dual-CA) sub-module, which consists of an Inertia-Supplied Cross Attention (ISCA) block and a Projection-Guided Cross Attention (PGCA) block. ISCA block introduces multi-channel inertia forces and increases the memory effect by a cross attention mechanism between adjacent iterations. And, PGCA block achieves an enhanced information interaction, which introduces the inertia force into the gradient descent step through a cross attention block. Extensive CS experiments manifest that our OCTUF achieves superior performance compared to state-of-the-art methods while requiring significantly cheaper parameters. Codes are available at https://github.com/songjiechong/OCTUF.

1. Introduction

Compressive sensing (CS) is a considerable research interest from signal/image processing communities as a joint acquisition and reconstruction approach [5]. The signal is first sampled and compressed simultaneously with linear random transformations. Then, the original signal can be reconstructed from far fewer measurements than that required by Nyquist sampling rate [29, 38]. So, the two main concerns of CS are the design of the sampling matrix [7, 16] and recovering the original signal [60], and our work focuses on the latter. Meanwhile, the CS technology achieves great success in many image systems, including medical imaging [31, 45], single-pixel cameras [15, 37], wireless remote monitoring [59], and snapshot compressive imaging [4, 50, 51], because it can reduce the measurement and storage space while maintaining a reasonable reconstruction of the sparse or compressible signal.

Mathematically, a random linear measurement $y \in \mathbb{R}^M$ can be formulated as $y = \Phi x$, where $x \in \mathbb{R}^N$ is the original signal and $\Phi \in \mathbb{R}^{M \times N}$ is the measurement matrix with $M \ll N$. $\frac{M}{N}$ is the CS ratio (or sampling rate). Obviously, CS reconstruction is an ill-posed inverse problem. To obtain a reliable reconstruction, the conventional CS methods

Figure 1. The PSNR (dB) performance (y-axis) of our OCTUF and some recent methods (ISTA-Net [54], DPA-Net [44], AMP-Net [60], MAC-Net [19], COAST [53], MADUN [41], CASNet [7], TransCS [39], FSOINet [10], MR-CCSNet [16]) under different parameter capacities (x-axis) on Set11 [24] dataset in the case of CS ratio $= 25\%$. Our proposed method outperforms previous methods while requiring significantly cheaper parameters.

\textsuperscript{*}Corresponding author. This work was supported in part by Shenzhen Research Project under Grant JCY202205311093215035 and Grant JSGGZD2022082209580001.
commonly solve an energy function as:

$$\arg\min_x \frac{1}{2} \| \Phi x - y \|^2 + \lambda R(x), \quad (1)$$

where $\frac{1}{2} \| \Phi x - y \|^2$ denotes the data-fidelity term for modeling the likelihood of degradation and $\lambda R(x)$ denotes the prior term with regularization parameter $\lambda$. For traditional model-based methods [17, 20, 26, 32, 56, 57, 64], the prior term can be the sparsifying operator corresponding to some predefined transform basis, such as discrete cosine transform (DCT) and wavelet [61, 62]. They enjoy the merits of strong convergence and theoretical analysis in most cases but are usually limited in high computational complexity and low adaptivity [63]. Recently, fueled by the powerful learning capacity of deep networks, several network-based CS algorithms have been proposed [24, 44]. Although network-based methods can solve CS problem adaptively with fast inferences, the architectures of most of these methods are the black box design and the advantages of traditional algorithms are not fully considered [36].

More recently, some deep unfolding networks (DUNs) with good interpretability are proposed to combine network with optimization and train a truncated unfolding inference through an end-to-end learning manner, which has become the mainstream for CS [52–55, 60]. However, existing deep unfolding algorithms usually achieve excellent performance with a large number of iterations and a huge number of parameters [41, 42], which are easily limited by storage space. Furthermore, the image-level transmission at each iteration fails to make full use of inter-stage feature information.

To address the above problems, in this paper, we propose an efficient Optimization-inspired Cross-attention Transformer (OCT) module as the iterative process and establish a lightweight OCT-based Unfolding Framework (OCTUF) for image CS, as shown in Fig. 2. Our OCT module maintains maximum information flow in feature space, which consists of a Dual Cross Attention (Dual-CA) sub-module and a Feed-Forward Network (FFN) sub-module to form each iterative process. Dual-CA sub-module contains an Inertia-Supplied Cross Attention (ISCA) block and a Projection-Guided Cross Attention (PGCA) block. ISCA block calculates cross attention on adjacent iteration information and adds inertial/memory effect to the optimization algorithm. And, PGCA block uses the gradient descent step and inertial term as inputs of Cross Attention (CA) block to guide the fine fusion of channel-wise features. With the proposed techniques, OCTUF outperforms state-of-the-art CS methods with much fewer parameters, as illustrated in Fig. 1. The main contributions are summarized as follows:

- We design a compact Dual Cross Attention (Dual-CA) sub-module to guide the efficient multi-channel information interactions, which consists of a Projection-Guided Cross Attention (PGCA) block and an Inertia-Supplied Cross Attention (ISCA) block.
- Extensive experiments demonstrate that our proposed OCTUF outperforms existing state-of-the-art methods with cheaper computational and memory costs.

2. Related Work

2.1. Deep Unfolding Network

The main idea of deep unfolding networks (DUNs) is that conventional iterative optimization algorithms can be implemented equivalently by a stack of recurrent trainable blocks. Such correspondence has been proposed to solve different image inverse tasks, such as denoising [11, 25], deblurring [23, 47], and demosaicking [22]. The solution is usually formulated as a bi-level optimization problem:

$$\min_{\Theta} \sum_{j=1}^{N} \mathcal{L}(\hat{x}_j, x_j),$$

$$\text{s.t. } \hat{x}_j = \arg\min_x \frac{1}{2} \| \Phi x - y_j \|^2 + \lambda R(x), \quad (2)$$

where $\Theta$ denotes the trainable parameters and $\mathcal{L}(\hat{x}_j, x_j)$ represents the loss function of estimated clean image $\hat{x}_j$ with respect to the original image $x_j$.

In the community of compressive sensing, DUN-based methods usually integrate some effective convolutional neural network (CNN) denoisers into some optimization methods, e.g., proximal gradient descent (PGD) algorithm [7, 9, 10, 36, 41, 53, 54], approximate message passing (AMP) [96], and inertial proximal algorithm for nonconvex optimization ( iPiano) [43]. Different optimization methods lead to different optimization-inspired DUNs. In most DUNs, the input and output of each iteration are inherently images $x_j$, which seriously hamper information transmission, resulting in limited representation capability [58]. Recently, some methods [10, 36] propose the idea of combining information flow into each iteration process in feature space to enhance information transmission. However, existing solutions usually lack flexibility in dealing with channel-wise information and are beset by high model complexity. In this paper, we present an efficient solution.

2.2. Vision Transformer

Inspired by the success of Transformers [46] in natural language processing, recent researchers also extend the Transformer structure for various computer vision tasks, e.g., image classification [14, 28], object detection [6, 66], segmentation [35, 48]. Transformer-based methods are also applied to image restoration tasks. PIT [8] is the first...
work to introduce Transformer to image restoration and achieves promising performance in several image restoration tasks. Subsequently, several novel designs are proposed. [27,49] utilize the Swin Transformer [28] to perform image restoration. Recently, some research [2,34] finds

\[
\text{F}_{\text{verse problems}} \text{[12].}
\]

Figure 2. Architecture of our OCTUF, which consists of Transformer block and Shen et al. image restoration. Recently, Cai et al. [39] design an ISTA-based Transformer (OCT) module that is the 6th iterative process, and \(\hat{x}\) is the recovered result gotten from the output \(X^{(k)}\) in the \(K\)th iteration.

3. Proposed Method

3.1. Overall Architecture

The proximal gradient descent (PGD) algorithm is a well-suited approach for solving many large-scale linear inverse problems [12]. Recently, some research [2,34] finds that such an algorithm adding the inertial term always succeeds to converge to the global optimum. Ochs et al. [34] propose an inertial proximal algorithm for nonconvex optimization (iPiano), which combines the gradient descent term with an inertial force. Inspired by iPiano, the whole update steps (for the \(k\)th iteration) can be expressed as:

\[
\begin{align*}
\text{s}^{(k)} &= \text{x}^{(k-1)} - \rho^{(k)} \Phi^\top (\Phi \text{x}^{(k-1)} - \text{y}) + \alpha^{(k)} ((\text{x}^{(k-1)} - \text{x}^{(k-2)}), \\
\text{x}^{(k)} &= \text{arg min}_x \frac{1}{2} \|x - \text{s}^{(k)}\|_2^2 + \lambda \mathcal{R}(x),
\end{align*}
\]

where \(\text{s}^{(k)}\) is the output image of the \(k\)th iteration, \(\text{y}\) is the sampled image, \(\rho^{(k)}\), \(\alpha^{(k)}\) are the learnable step size parameters and \(\Phi^\top\) is the transpose of the measurement matrix \(\Phi\).

Eq. (3) denotes the projection step, which introduces an inertial term \(\alpha^{(k)} ((\text{x}^{(k-1)} - \text{x}^{(k-2)})\) to a gradient descent term \(\mathcal{F}(\text{x}^{(k-1)}) = \text{x}^{(k-1)} - \rho^{(k)} \Phi^\top (\Phi \text{x}^{(k-1)} - \text{y})\), relaxing the monotonically decreasing constraints and helping to achieve a better convergence result [33]. As mentioned previously, such traditional implementation lacks adaptability and has information loss due to image-level inter-stage transmission. To rectify these weaknesses, we propose a Dual Cross Attention (Dual-CA) sub-module to achieve feature-level transmission by adding a multi-channel inertial force and enhancing the information interaction in the projection step. Eq. (4) is achieved by a proximal mapping step which is actually a Gaussian denoiser. Here like most DUNs, we implement it with a trainable model, i.e., a Feed Forward Network (FFN) sub-module that is detailed in Fig. 3(e). FFN consists of two sets of LayerNorm and Feed Forward Block (FFB) with a global skip connection, where the architecture of FFB is similar to [3].

Therefore, as the process in the \(k\)th iteration of OCTUF, our Optimization-inspired Cross-attention Transformer (OCT) module can be formulated as \((k \in \{1, 2, \ldots, K\})\):

\[
\begin{align*}
\text{s}^{(k)} &= \mathcal{H}_{\text{Dual-CA}}(\text{x}^{(k-1)}, \text{Z}^{(k-2)}), \\
\text{x}^{(k)} &= \mathcal{H}_{\text{FFN}}(\text{s}^{(k)}),
\end{align*}
\]

where \(\text{s}^{(k)}\), \(\text{x}^{(k)}\) \(\in \mathbb{R}^{H \times W \times C}\) are the outputs in the feature domain, and \(\text{Z}^{(k-2)}\) \(\in \mathbb{R}^{H \times W \times (C-1)}\) is obtained by clipping latter \(C-1\) channels from \(\text{x}^{(k-2)}\). For the first iteration, the input \(\text{x}^{(0)}\) is generated by a \(3 \times 3\) convolution (\(\text{Conv}_3(\cdot)\) on the initialization \(\text{x}^{(0)}\), and the inertial term is not needed [34], as shown in Fig. 2. And the recovered result \(\hat{x}\) is gotten by splitting the first channel from \(\text{x}^{(K)}\).

Therefore, our proposed OCTUF can skillfully integrate the inter-stage feature-level information and achieves the perfect combination with the optimization steps.

3.2. Dual Cross Attention

To ensure maximum information flow and powerful feature correlation, we design a Dual Cross Attention (Dual-CA) sub-module to efficiently fuse information in the projection step. As shown in Fig. 3(a), to make the network more compact and maintain the potential mathematical interpretation, we finely split the input \(\text{x}^{(k-1)} \in \mathbb{R}^{H \times W \times C}\) into two chunks, including \(\text{x}^{(k-1)} \in \mathbb{R}^{H \times W \times 1}\) (from the first channel) and \(\text{Z}^{(k-2)} \in \mathbb{R}^{H \times W \times (C-1)}\) (from the last \(C-1\) channels). \(\text{r}^{(k-1)}\) and \(\text{Z}^{(k-1)}\) are the input of the gradient descent term and the inertial term, respectively. So, to make full use of the information of the multi-channel inertial term, we design an Inertia-Supplied Cross Attention (ISCA) block, yielding \(\mathcal{H}_{\text{ISCA}}(\text{Z}^{(k-1)}, \text{Z}^{(k-2)})\). And we also propose a Projection-Guided Cross Attention (PGCA) block to perform the fusion of the gradient descent term and
Figure 3. The architecture of Optimization-inspired Cross-attention Transformer (OCT) module. (a) OCT module consists of a Dual Cross Attention (Dual-CA) sub-module which contains an Inertia-Supplied Cross Attention (ISCA) block and a Projection-Guided Cross Attention (PGCA) block, and a Feed-Forward Network (FFN) sub-module. (b) Illustration of Cross Attention (CA) block, which is the basic component of two attention blocks. (c) ISCA block is composed of Layer Normalization (LN) and CA. (d) PGCA block is composed of Gradient Descent Block (GDB), LN, and CA. (e) FFN sub-module is composed of two sets of LN and Feed-Forward Block (FFB).

Among ISCA and PGCA blocks, Cross Attention (CA) plays an important role as the basic block. In the following part of Sec. 3.2, we first introduce the Cross Attention block and then present ISCA and PGCA blocks, respectively.

Cross Attention. Motivated by modeling complex relations for generating context-aware objects in multi-modal task [65], we design a Cross Attention (CA) block to aggregate the key information from the different components in the projection step of deep unfolding network, as shown in Fig. 3(b). The input Q comes from a different component than V and K. They are first embedded by a 1x1 convolution (ConvV,K,Q(·)) to obtain feature with the size being $H \times W \times (C - 1)$. Then a 3x3 depth-wise convolution (DeconvV,K,Q(·)) is used to encode channel-wise spatial context. Finally, a reshape operation (R(·)) reformulates V, K, and Q into tokens $\{\tilde{V}, \tilde{K}, \tilde{Q}\} \in \mathbb{R}^{H\times W \times (C - 1)}$. Therefore, this process can be defined as the following function:

$$
\begin{align*}
\tilde{V} &= R(\text{Deconv}(\text{Conv}(V))), \\
\tilde{K} &= R(\text{Deconv}_K(\text{Conv}_K(K))), \\
\tilde{Q} &= R(\text{Deconv}_Q(\text{Conv}_Q(Q))).
\end{align*}
$$

Next, a transposed attention map $A \in \mathbb{R}^{(C - 1) \times (C - 1)}$ is generated by applying softmax function to re-weight the matrix multiplication $\tilde{K}^T \tilde{Q}$, yielding

$$
A = \text{Softmax}(\tilde{K}^T \tilde{Q}),
$$

where $\tilde{K}^T$ denotes the transposed matrix of $\tilde{K}$. The aggregation result is calculated as $\tilde{V}A$, which is reshaped into the features of size $H \times W \times (C-1)$. Finally, we apply a 1x1 convolution ConvA(·) to enhance the feature extraction. Overall, the Cross Attention block is defined as:

$$
G_{CA}(V, K, Q) = \text{Conv}_A(R(\tilde{V}A)).
$$

Cross Attention block helps to extract useful information via channel-wise similarity with low computational cost.

Inertia-Supplied Cross Attention. As shown in Eq. (3), the general inertial term usually adopts the simple operation by directly subtracting the adjacent iteration output, which is proved to be ineffective in the ablation of Tab. 4. To enrich the information interaction of the inertial term, we introduce a multi-channel inertial term and propose an Inertia-Supplied Cross Attention (ISCA) block. Our ISCA block consists of LayerNorm (LN) function and CA block as shown in Fig. 3(c). Specifically, we set the $(k-2)$th iteration output $Z^{(k-2)}$ as value ($V^{(k)}_{ISCA}$) and key ($K^{(k)}_{ISCA}$), and we set the $(k-1)$th iteration output $Z^{(k-1)}$ as query ($Q^{(k)}_{ISCA}$), pass through CA block after normalization by LN function, so $\tilde{Z}^{(k-1)} = H_{ISCA}(Z^{(k-1)}, Z^{(k-2)})$ as:

$$
\begin{align*}
V^{(k)}_{ISCA}, K^{(k)}_{ISCA}, Q^{(k)}_{ISCA} &= \text{LN}(Z^{(k-2)}), \text{LN}(Z^{(k-2)}), \text{LN}(Z^{(k-1)}), \\
\tilde{Z}^{(k-1)} &= G_{CA}(V^{(k)}_{ISCA}, K^{(k)}_{ISCA}, Q^{(k)}_{ISCA}) + Z^{(k-1)}.
\end{align*}
$$

ISCA block adaptively learns more useful multi-channel inertial force and enhances memory effect to our network.
Projection-Guided Cross Attention. In order to adaptively combine the gradient descent term and the inertial term, we introduce a Projection-Guided Cross Attention (PGCA) block in Fig. 3(d). Similar to ISCA block, PGCA block captures rich feature information based on channel-wise similarity. Specifically, given $X^{(k-1)}$, the input of gradient descent term is gotten by its first channel (i.e., $r^{(k-1)}$). So, the calculation of the term has the following expression:

$$\tilde{r}^{(k-1)} = r^{(k-1)} - \rho^{(k)} \Phi^T (\Phi r^{(k-1)} - y).$$

Next, $\tilde{r}^{(k-1)}$ and the ISCA output $\tilde{Z}^{(k-1)}$ pass through LayerNorm function and CA block, yielding

$$V^{(k)}_{\text{PGCA}}, K^{(k)}_{\text{PGCA}}, Q^{(k)}_{\text{PGCA}} = \text{LN}(\tilde{Z}^{(k-1)}), \text{LN}(\tilde{Z}^{(k-1)}), \text{LN}(\tilde{r}^{(k-1)}),$$

$$O^{(k)} = G_C (V^{(k)}_{\text{PGCA}}, K^{(k)}_{\text{PGCA}}, Q^{(k)}_{\text{PGCA}}) + \tilde{Z}^{(k-1)}.$$  \hspace{1cm} (13)

It is worth noting that $\tilde{r}^{(k-1)}$ and $\tilde{Z}^{(k-1)}$ are concatenated, reshaped to match the original channel dimensions and mixed with a $1 \times 1$ convolution ($\text{Conv}_O(\cdot)$):

$$S^{(k)} = \text{Conv}_O(\text{Concat}(O^{(k)}, \tilde{r}^{(k-1)})).$$  \hspace{1cm} (14)

PGCA not only inherits the advantage of Eq. (3) but also tactfully achieves the multi-channel feature fusion between the gradient descent term and the inertial term.

### 3.3. Loss Function

Given a set of full-sampled images $\{x_j\}_{j=1}^{N_a}$ and some sampling patterns with the specific sampling rate, the compressed measurements can be obtained by $y_j = \Phi x_j$, producing the train data pairs $\{(y_j, x_j)\}_{j=1}^{N_a}$. Our model takes $y_j$ as input and generates the reconstruction result $\hat{x}_j$ as output. We employ the MSE loss function with respect to $x_j$ and $\hat{x}_j$ as following shows:

$$L(\Theta) = \frac{1}{NN_a} \sum_{j=1}^{N_a} ||x_j - \hat{x}_j||_2^2,$$  \hspace{1cm} (15)

where $N_a$ and $N$ represent the number of the training images and the size of each image respectively. $\Theta$ denotes the learnable parameter set of our proposed OCTUF and can be formulated as $\Theta = \{\Phi, \text{Conv}_O(\cdot)\} \cup \{\mathcal{H}^{(k)}_{\text{Dual-CA}}(\cdot), \mathcal{H}^{(k)}_{\text{FPN}}(\cdot)\}_{k=1}^{K}$.

### 4. Experiments

#### 4.1. Implementation Details

For training, we use 400 images from the training and test dataset of BSDS500 dataset [1]. The training images are cropped to 89600 patches of $96 \times 96$ pixel size with data augmentation following [40]. For a given CS ratio $\frac{M}{N}$, the corresponding learnable measurement matrix $\Phi$ is constructed by a convolution layer with the kernel size of $M \times 1 \times \sqrt{N} \times \sqrt{N}$ to sample the original image $x$. And then, we utilize the transpose convolution whose kernel weight is the sampling matrix to obtain initialization $x^{(0)}$.

For the network parameters, the block size is 32, i.e. $N = 1, 024$, the default batch size is 16, the default number of feature maps $C$ is 32 and the learnable parameter $\rho^{(k)}$ is initialized to 0.5. We use Adam [21] optimizer to train the network with the initial learning rate, which decreased to $5 \times 10^{-5}$ through 100 epochs using the cosine annealing strategy [10, 30] and the warm-up epochs are 3. For testing, we utilize two widely-used benchmark datasets, yielding Set11 [24] and Urban100 [13]. Color images are processed in the YCbCr space and evaluated on the Y channel. Two common-used image assessment criteria, Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM), are adopted to evaluate the reconstruction results.
We also show the number of parameters and the computation cost (including the computations of the convolution, the fully-connected layer and matrix multiplication) measured in floating-point operations per second (FLOPs).

### 4.2. Qualitative Evaluation

We compare our proposed methods with recent representative CS reconstruction methods. The average PSNR/SSIM reconstruction performances on Set11 dataset [24] with respect to five CS ratios are summarized in Tab. 1. For our OCTUF, we set the iteration number as 10 and set the initial learning rate as $5 \times 10^{-4}$. To further improve the model performance, we also present a plus version, namely OCTUF+, whose iteration number is 16 and initial rate is $2 \times 10^{-4}$. From Tab. 1, one can observe that our OCTUF and OCTUF+ outperform all the other competing methods in PSNR and SSIM across all the cases. On average, OCTUF+ outperforms ISTA-Net+ [54], DPA-Net [44], AMP-Net [60], MAC-Net [19], COAST [53], MADUN [41], CASNet [7], TransCS [39] and FSOINet [10] by 3.60 dB, 3.91 dB, 1.25 dB, 3.36 dB, 1.16 dB and 0.29 dB in terms of PSNR on Set11 dataset, respectively. In addition, the average SSIM of OCTUF+ can be improved 0.0389, 0.0280, 0.0113, 0.0381, 0.0202, 0.0013, 0.0015, 0.0073 and 0.0010, respectively. Fig. 4 further show the visual comparisons on challenging images when CS ratio is 25%, which can be seen that our OCTUFs can recover much clear edge information than other methods. Furthermore, in Tab. 2, we compare OCTUFs with other methods on Urban100 dataset [13] that contains more high-resolution images and incorporates more abundant image distributions. It shows that both OCTUF and OCTUF+ achieve a better reconstruction quality at all sampling ratios. Fig. 5 presents the visual comparisons on challenging images. Our OCTUF and OCTUF+ generate images that are visually pleasant and faithful to the groundtruth. It
Table 2. Average PSNR(dB)/SSIM performance comparisons of recent deep network-based CS methods on Urban100 dataset [13] with different CS ratios. The best and second best results are highlighted in red and blue colors, respectively.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Methods</th>
<th>CS Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>10%</td>
</tr>
<tr>
<td>Urban100</td>
<td>ISTA-Net+ (CVPR 2018) [54]</td>
<td>23.61/0.7238</td>
</tr>
<tr>
<td></td>
<td>DPA-Net (TIP 2020) [44]</td>
<td>24.55/0.7841</td>
</tr>
<tr>
<td></td>
<td>AMP-Net (TIP 2020) [60]</td>
<td>26.04/0.8151</td>
</tr>
<tr>
<td></td>
<td>MAC-Net (ECCV 2020) [9]</td>
<td>24.21/0.7445</td>
</tr>
<tr>
<td></td>
<td>COAST (TIP 2021) [53]</td>
<td>25.94/0.8035</td>
</tr>
<tr>
<td></td>
<td>MADUN (ACM MM 2021) [41]</td>
<td>27.13/0.8393</td>
</tr>
<tr>
<td></td>
<td>CASNet (TIP 2022) [7]</td>
<td>24.21/0.7445</td>
</tr>
<tr>
<td></td>
<td>TransCS (TIP 2022) [39]</td>
<td>26.72/0.8413</td>
</tr>
<tr>
<td></td>
<td>FSOINet (ICASSP 2022) [10]</td>
<td>27.53/0.8627</td>
</tr>
<tr>
<td></td>
<td>OCTUF (Ours)</td>
<td>27.79/0.8621</td>
</tr>
<tr>
<td></td>
<td>OCTUF+ (Ours)</td>
<td>27.92/0.8652</td>
</tr>
</tbody>
</table>

Table 3. Ablation study of our approach on Set11 dataset [24] in the case of CS ratio = 50%. The best performance is labeled in bold.

<table>
<thead>
<tr>
<th>Cases</th>
<th>Dual-CA</th>
<th>FFN</th>
<th>LayerNorm</th>
<th>Learning rate</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>5e-4(warmup)</td>
<td>38.25</td>
</tr>
<tr>
<td>(b)</td>
<td>-</td>
<td>√</td>
<td>-</td>
<td>5e-4(warmup)</td>
<td>38.96</td>
</tr>
<tr>
<td>(c)</td>
<td>√</td>
<td>-</td>
<td>-</td>
<td>5e-4(warmup)</td>
<td>41.16</td>
</tr>
<tr>
<td>(d)</td>
<td>√</td>
<td>√</td>
<td>-</td>
<td>5e-4(warmup)</td>
<td>41.21</td>
</tr>
<tr>
<td>(e)</td>
<td>√</td>
<td>√</td>
<td>-</td>
<td>1e-4(fix)</td>
<td>41.17</td>
</tr>
<tr>
<td>(f)</td>
<td>√</td>
<td>-</td>
<td>√</td>
<td>2e-4(fix)</td>
<td>41.27</td>
</tr>
<tr>
<td>(g)</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>5e-4(fix)</td>
<td>41.30</td>
</tr>
<tr>
<td>OCTUF</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>5e-4(warmup)</td>
<td>41.34</td>
</tr>
</tbody>
</table>

4.3. Ablation Study

In this part, we conduct ablation studies on Set11 dataset for our OCTUF whose iteration number is 10.

Break-down Ablation. We first conduct a break-down ablation experiment in the case of CS ratio = 50% to investigate the effect of each component towards higher performance. The results are listed in Tab. 3. Case (a) is our baseline which contains ResBlock [18] with a similar number of parameters as OCT module. When we successively apply our FFN and Dual-CA sub-modules respectively, namely Cases (b) and (c), the model achieves 0.71 dB and 2.91 dB improvements. And the model can greatly enhance 3.09 dB gains with little storage place when both sub-modules are used together. We also discuss the effect of the LayerNorm (LN) function in Case (d), which addresses that our OCTUF achieves better performance with the LayerNorm function. Note that without “LayerNorm” represents removing all LN from our OCTUF. What is more, we train our models with different learning rates as seen from Cases (e), (f), and (g). “fix” denotes that the learning rate is not changed during training, and “warmup” denotes that the training strategy is the same with our work as shown in Sec. 4.1. Our proposed OCTUF has a stable training process with large learning rates and meanwhile, we use the “warmup” strategy to improve its anytime performance when training.

Dual Cross Attention. We also do elaborate ablation experiments on the components of Dual-CA sub-module in the case of CS ratio = 30% in Tab. 4, where “IF” denotes the inertial force achieved by the easy way and “FD” denotes the enhanced iterative process in the feature domain. The best PSNR(dB) is labeled in bold.

<table>
<thead>
<tr>
<th>Cases</th>
<th>FFN</th>
<th>GDB</th>
<th>IF</th>
<th>FD</th>
<th>PGCA</th>
<th>ISCA</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>√</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>34.59</td>
</tr>
<tr>
<td>(b)</td>
<td>√</td>
<td>√</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>35.93</td>
</tr>
<tr>
<td>(c)</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>36.82</td>
</tr>
<tr>
<td>(d)</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>-</td>
<td>-</td>
<td>36.83</td>
</tr>
<tr>
<td>(e)</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>-</td>
<td>√</td>
<td>-</td>
<td>37.13</td>
</tr>
<tr>
<td>(f)</td>
<td>√</td>
<td>-</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>-</td>
<td>37.08</td>
</tr>
<tr>
<td>OCTUF</td>
<td>√</td>
<td>-</td>
<td>√</td>
<td>-</td>
<td>√</td>
<td>√</td>
<td>37.21</td>
</tr>
</tbody>
</table>
Table 5. Ablation of Feed-Forward Network on Set11 dataset [24] when CS ratio = 30%. The best performance is labeled in bold.

<table>
<thead>
<tr>
<th>Method</th>
<th>Baseline</th>
<th>LN+FFB</th>
<th>LN+2×FFB</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSNR(dB)</td>
<td>37.05</td>
<td>37.04</td>
<td>37.12</td>
<td>37.21</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.9663</td>
<td>0.9664</td>
<td>0.9672</td>
<td>0.9673</td>
</tr>
<tr>
<td>Parameters</td>
<td>0.61 M</td>
<td>0.52 M</td>
<td>0.61 M</td>
<td>0.61 M</td>
</tr>
</tbody>
</table>

contains a neural network. Then, the performance can continuously improve by 0.89 dB after using “FD” shown in Case (c). We also conduct fine contrast experiments for the inertial force in Cases (c), (d), and (e), demonstrating that our ISCA block can more fully play the role of inertia force. And as shown in Cases (e)(f), applying PGCA and ISCA blocks can get better performance. Our proposed Dual-CA sub-module takes into account the gradient descent algorithm and the inertial force, allocates the different channels reasonably, and gives full play to the structural characteristics. Tab. 4 should be also noted that “GDB” is included in “PGCA” so it is not selected when “PGCA” is selected. Moreover, to intuitively show the advantages of Dual-CA sub-module, we visualize the feature map in the fifth iteration for four cases. The result in Fig. 6 presents that both our ISCA and PGCA blocks pay more high-fidelity attention to the detailed contents and structural textures.

Feed-Forward Network. As shown in Fig. 3 (e), our proposed Feed-Forward Network (FFN) sub-module consists of two groups of LayerNorm and Feed-Forward Block (FFB) with a global skip connection. We do ablations to investigate the effects of the group number and LayerNorm (LN) number in Tab. 5. “Baseline” is the same setting with Case (c) of Tab. 3. “LN+FFB” denotes one group, and “LN+2×FFB” denotes that Norm is only added to the first group. Therefore, as can be seen from the table, our proposed method achieves the best performance.

4.4. Complexity Analysis

Table 6. Comparison of the parameters, the model size and FLOPs for reconstructing a 256×256 image in the case of CS ratio = 10%. The best performance is labeled in bold.

<table>
<thead>
<tr>
<th>Method</th>
<th>MADUN</th>
<th>CASNet</th>
<th>FSOINet</th>
<th>OCTUF</th>
<th>OCTUF +</th>
</tr>
</thead>
<tbody>
<tr>
<td>Params(M)</td>
<td>3.14</td>
<td>16.90</td>
<td>0.64</td>
<td>0.40</td>
<td>0.58</td>
</tr>
<tr>
<td>Size(MB)</td>
<td>11.9</td>
<td>66.3</td>
<td>7.8</td>
<td>5.2</td>
<td>7.5</td>
</tr>
<tr>
<td>FLOPs(G)</td>
<td>419.2</td>
<td>13391.5</td>
<td>266.6</td>
<td>189.3</td>
<td>294.6</td>
</tr>
</tbody>
</table>

![Figure 6. Visual analysis of the feature map in the fifth iteration of our proposed OCTUF. It shows that both ISCA and PGCA blocks pay more high-fidelity attention to details.](image)

4.5. Sensitivity to Noise

In the real application, the imaging model may be affected by noise and currently, no real open dataset is suitable for such CS reconstruction methods. So to test the robustness of our designed OCTUF, we first add the Gaussian noise with different noise levels to the original images for Set11 dataset. Then, OCTUF and the other methods take the noisy images as input, and sample and recover the reconstruction images when CS ratios are 10% and 25%. Fig. 7 shows the plots of PSNR values of all methods versus various standard variances noise. It can be seen that our OCTUF possesses strong robustness to noise corruption.

5. Conclusion

In this paper, we propose a novel optimization-inspired cross-attention Transformer (OCT) module as an iteration, leading to a lightweight OCT-based unfolding framework (OCTUF) for CS. Specifically, we present a Dual Cross Attention (Dual-CA) sub-module, which contains an inertia-supplied cross attention (ISCA) block and a projection-guided cross attention (PGCA) block as the projection step in iterative optimization. ISCA block precisely helps to achieve a good convergence by introducing a feature-level inertial term. PGCA block utilizes a cross attention mechanism to fuse the gradient descent term and the inertial term while ensuring the maximum information flow. Extensive experiments show that our OCTUF achieves superior performance compared to state-of-the-art methods with lower complexity. In the future, we will extend our OCTUF to other image inverse problems and video applications.
References


[17] Xingwei Gao, Jian Zhang, Wenbin Che, Xiaopeng Fan, and Debin Zhao. Block-based compressive sensing coding of natural images by local structural measurement matrix. In *Proceedings of Data Compression Conference (DCC)*, 2015. 2


[65] Lichen Zhao, Daigang Cai, Lu Sheng, and Dong Xu. 3DVG-Transformer: Relation modeling for visual grounding on point clouds. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021. 4

[66] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transform-