Supplement 1

A Brief Introduction to Reconstruction Algorithms

1. Differential Ghost Imaging (DGI) Algorithm

Differential Ghost Imaging (DGI)[1] is an enhanced ghost imaging technique aimed at improving the Signal-to-Noise Ratio (SNR) and reconstruction quality of Single-Pixel Imaging (SPI) by considering the fluctuations in illumination intensity. In traditional ghost imaging, the correlation between the target scene and illumination light fields is used to reconstruct the image, which is susceptible to changes in illumination intensity. DGI corrects for these variations by introducing the total intensity of each illumination light field.

The core of the DGI algorithm lies in normalizing the illumination intensity fluctuations' impact on the reconstructed image. Its reconstruction formula is:

$$x = \langle b a_i \rangle - {\langle b \rangle \langle a_i \rangle} \big/ {\langle s_i \rangle \langle a_i \rangle}$$

, where x represents the target scene, b is the measured value, a_i is the i-th illumination light field, s_i is the total intensity of the i-th illumination light field, and $\langle \cdot \rangle$ denotes the average over all light fields.

DGI is a non-iterative method that directly obtains the reconstructed image through correlation calculations, with low computational complexity, suitable for real-time imaging systems. By normalization, DGI exhibits strong robustness to noise, capable of resisting environmental light interference and circuit noise to a certain extent. The implementation process of DGI is relatively straightforward, without the need for complex iterative optimization processes.

DGI is suitable for single-pixel imaging systems that require high real-time performance and noise robustness, such as target imaging in scenarios with significant environmental light interference.

2. Total Variation Based Compressive Sensing (CS-TV) Algorithm

The TV-based CS algorithm (CS-TV)[2] is a nonlinear iterative reconstruction method for SPI that incorporates sparsity priors of natural images and employs TV regularization to reconstruct signals from underdetermined linear systems. The TV regularization assumes that the gradient integral of natural images is statistically low, indicating sparse edge information. By introducing TV regularization, this algorithm enables high-quality image reconstruction with fewer measurements, thus enhancing the acquisition efficiency of SPI systems.

The optimization model for CS-TV is: min $\|c\|_1$ subject to Gx=c, Ax=b, where $\|c\|_1$ denotes the l_1 norm of c, approximating the sparsity of c; G is the gradient calculation matrix for computing the total variation of the image; x is the target scene, A is the illumination light fields matrix, and b is the measurement vector. With the introduction of Lagrange multipliers and balancing parameters, this optimization problem can be solved under a gradient descent framework, involving iterative updates of c, x, Lagrange multipliers, and balancing parameters.

The CS-TV algorithm achieves high-quality image reconstruction with fewer measurements, significantly improving the acquisition efficiency of SPI systems. The TV regularization term has some noise suppression effect, allowing the algorithm to maintain good reconstruction performance in the presence of measurement noise. As a nonlinear iterative method, the CS-TV algorithm has relatively high computational complexity, especially for large-scale image reconstruction.

The CS-TV algorithm is suitable for SPI systems that require high acquisition efficiency, such as scenarios where reducing the number of measurements is needed to increase imaging speed or decrease system complexity.

3. Untrained Deep Neural Network (UNN) Algorithm

The Untrained Neural Network (UNN)[3] embeds physical models into deep neural networks, leveraging the network's optimization capabilities to automatically adjust parameters for high-quality image reconstruction without extensive pre-training of datasets. This approach exhibits strong generalization and robustness in low sampling rates and high noise levels.

The UNN typically employs the classic U-Net structure, comprising an encoder and decoder with skip connections to transmit and compensate for information. U-Net effectively extracts image features and restores resolution during decoding. To further enhance image quality, some improved UNN methods incorporate attention mechanisms to efficiently highlight main features and suppress noise.

In UNN, the GI physical model is integrated into the network's output layer. Specifically, the network's output is correlated with known illumination light fields to estimate measurement values. Then, the difference between actual and estimated measurements (loss function) is used to optimize network parameters via backpropagation. The UNN's loss function usually includes two parts: the error between measurements and the regularization term (e.g., total variation regularization). By minimizing the loss function, the network gradually optimizes parameters to improve reconstructed image quality. Typically, the Adam optimizer is used with dynamic learning rate strategies to accelerate network convergence.

The UNN does not require extensive pre-training of datasets, significantly saving time and resources, enhancing the algorithm's applicability and flexibility. Since the UNN directly utilizes physical models for optimization, it has good generalization ability across different imaging scenarios and noise levels, adapting to complex real-world applications. By introducing attention mechanisms and total variation regularization constraints, the UNN effectively suppresses noise, improving signal-to-noise ratio (SNR), thus achieving high-quality image reconstruction at low sampling rates and high noise levels. The UNN can reconstruct high-quality images at low sampling ratios (e.g., below 10%), significantly outperforming traditional DGI and CS-TV methods.

4. Reconstruction time comparison among different algorithms

	DGI	CS-TV	UNN
Digit 2	0.0863 s	5.59 s	52.5 s

The computation was performed on a PC. CPU: AMD Ryzen 7 4800U; GPU: AMD Radeon(TM) Graphics