Spherical Image Inpainting with Frame Transformation and Data-driven Prior Deep Networks *

Jianfei Li[†], Chaoyan Huang[‡], Raymond Chan[§], Han Feng[¶], Michael Ng[‖], and Tieyong Zeng[#]

Abstract. Spherical image processing has been widely applied in many important fields, such as omnidirectional vision for autonomous cars, global climate modelling, and medical imaging. It is non-trivial to extend an algorithm developed for flat images to the spherical ones. In this work, we focus on the challenging task of spherical image inpainting with deep learning-based regularizer. Instead of a naive application of existing models for planar images, we employ a fast directional spherical Haar framelet transform and develop a novel optimization framework based on a sparsity assumption of the framelet transform. Furthermore, by employing progressive encoder-decoder architecture, a new and better-performed deep CNN denoiser is carefully designed and works as an implicit regularizer. Finally, we use a plug-and-play method to handle the proposed optimization model, which can be implemented efficiently by training the CNN denoiser prior. Numerical experiments are conducted and show that the proposed algorithms can greatly recover damaged spherical images and achieve the best performance over purely using deep learning denoiser and plug-and-play model.

Key words. Spherical image inpainting, deep CNN, plug and play

MSC codes. 68Q25, 68R10, 68U05

1. Introduction.

In practical problems, a large amount of data comes in the form of spherical images, such as from cosmology [20], astrophysics [22], geophysics [1, 23], neuroscience [21], and omnidirectional AR/VR field [2, 27], where images are naturally defined on the 2D spherical surface. Due to the storage bottleneck and observation being costly and infeasible, these spherical images (signals) usually contain very limited pixels (observed data), especially if the observation scales involved are large. Therefore, repairing missing or damaged parts is a fundamental yet challenging task in spherical image processing. Apparently, spherical images take a different inherent domain than planar images in 2D in terms of symmetries,

*Submitted to the editors DATE.

Funding: This work was supported in part by the National Key R&D Program of China under Grant 2021YFE0203700, Grant NSFC/RGC N_CUHK 415/19, Grant NSFC Nos. 11871210, 11971215, and 61971292, Grant ITF MHP/038/20, Grant RGC 14300219, 14302920, 14301121, and CUHK Direct Grant for Research. And in part by Hong Kong Research Grant Council GRF 12300218, 12300519, 17201020, 17300021, C1013-21GF, C7004-21GF and Joint NSFC-RGC N-HKU76921. And in part by HKRGC Grants Nos. CUHK14301718, CityU11301120, C1013-21GF, CityU Grant 9380101.

[†]Co-first authors. Department of Mathematics, City University of Hong Kong (jianfeili2-c@my.cityu.edu.hk)

[‡]Co-first authors. Department of Mathematics, The Chinese University of Hong Kong (cyhuang@math.cuhk.edu.hk)

[§]Department of Mathematics, City University of Hong Kong and Hong Kong Centre for Cerebro-Cardiovascular Health Engineering. (raymond.chan@cityu.edu.hk)

[¶]Corresponding author. Department of Mathematics, City University of Hong Kong (hanfeng@cityu.edu.hk)

[‖]Department of Mathematics, University of Hong Kong (mng@maths.hku.hk.)

[#]Department of Mathematics, The Chinese University of Hong Kong (zeng@math.cuhk.edu.hk)

This manuscript is for review purposes only.

coordinate systems, and translates, which demand special processing methods. In this paper, we are concerned with spherical image restoration, which can further serve as a preliminary for subsequent tasks, like object recognition and segmentation. Mathematically, it aims to estimate $\mathbf{x}$ from observation $\mathbf{y}$ for the following model

$$\mathbf{y} = T(\mathbf{x}) + \varepsilon, \tag{1.1}$$

where T is a degradation operator, ε is assumed to be the additive noise. Different degradation operations correspond to different image restoration (IR) tasks [13, 14]. Typically, the IR task would be image denoising when T is an identity operation, image deblurring when T is a two-dimensional convolution operation, image super-resolution when T is a composite operation of convolution and down-sampling, color image demosaicing when T is a color filter array (CFA) masking operation, and image inpainting when T is the orthogonal projection onto the linear space of matrices. In this paper, we proposed a general model for spherical image inpainting with a new denoiser.

Regarding the degradation equation (1.1), the IR task model can be solved through the following optimization,

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \|\mathbf{y} - T(\mathbf{x})\| + \lambda\Phi(\mathbf{x}), \tag{1.2}$$

where the first term is the data fitting with $\|\cdot\|$ usually chosen to be the Frobenius norm, the second term $\Phi(\cdot)$ is an operator playing the role of regularity, and λ is a positive trade-off parameter. With the aid of the half quadratic splitting (HQS) algorithm, by introducing an auxiliary variable, the optimization problem (1.2) can be addressed by iteratively solving the following subproblems

$$\mathbf{x}_k = \arg\min_{\mathbf{x}} \|\mathbf{y} - T(\mathbf{x})\|^2 + \alpha \|\mathbf{x} - \mathbf{z}_{k-1}\|^2, \tag{1.3}$$

$$\mathbf{z}_k = \arg\min_{\mathbf{z}} \alpha \|\mathbf{z} - \mathbf{x}_k\|^2 + \lambda\Phi(\mathbf{z}). \tag{1.4}$$

Here α will be set accordingly to specific problems. Equation (1.3) is usually interpreted as the data fitting subproblem and (1.4) as the regularization subproblem. Many research efforts have been devoted to this hot topic and achieved extensive improvements [3, 5, 11, 24, 25].

As one can see, the fidelity term and regularization term are decoupled into two individual subproblems. Specifically, the fidelity term is associated with a quadratic regularized least-squares problem (i.e., Eq. (1.3)) which has various fast solutions for different degradation matrices. The regularization term is involved in Eq. (1.4), which can be rewritten as

$$\mathbf{z}_{k+1} = \arg\min_{\mathbf{z}} \frac{1}{2(\sqrt{\lambda/\alpha})^2} \|\mathbf{x}_{k+1} - \mathbf{z}\|^2 + \Phi(\mathbf{z}). \tag{1.5}$$

According to Bayesian probability, Eq. (1.5) corresponds to denoising the image $\mathbf{x}_{k+1}$ by a Gaussian denoiser with noise level $\sqrt{\lambda/\alpha}$. As a consequence, any Gaussian denoisers can be acted as a modular part to solve Eq. (1.2). To address this, we rewrite Eq. (1.5) by following

$$\mathbf{z}_{k+1} = \text{Denoiser}\left(\mathbf{x}_{k+1}, \sqrt{\lambda/\alpha}\right). \tag{1.6}$$

It is worth noting that, according to Eqs. (1.5) and (3.7), the image prior $\Phi(\cdot)$ can be implicitly replaced by a denoiser prior. In this paper, we apply the proposed denoiser as prior to generate a better performance.

In recent years, deep learning-based models have extensively emerged and achieved state-of-the-art restoration performance [4, 9, 15, 17, 19]. The SeaNet proposed in [6] consists of three sub-nets for single image super-resolution with the help of image soft edge. Liu et al. [18] proposed MWCNN for image restoration, which is a U shape network with DWT and IWT for downsampling and upsampling, respectively, and thus there is no information loss during subsampling. Both approaches achieved competitive performance in IR tasks.

To improve interpretability and effectively use the trained neural networks from various tasks, Plug and Play is one of the choices to combine neural networks and prior knowledge of images with an optimization model. Zhang et al. [28, 29] developed the deep prior to handling the IR tasks, named plug-and-play (PnP). Specifically, they regarded the regularization term $\Phi(x)$ as a deep denoiser with the deep CNNs. The optimization problem (1.2) was solved by the half quadratic splitting (HQS) algorithm and divided into two subproblems, in which the solution of one of the problems is replaced by the deep CNN, which is the so-called deep denoiser.

Furthermore, the term (1.4) is usually termed as denoiser prior and conducted by a single CNN denoiser [31], which is trained specifically for denoising prior and to replace solving (1.4) to exploit the advantages of CNN. Following this line, the PnP-based model has wide applications [10]. For example, Wu et al. [26] proposed a deep CNN-based PnP framework with MWCNN and has competitive performance in Cauchy noise removal. Zhao et al. [31] suggested a PnP model for image completion with a low rankness assumption. Fang and Zeng [7] applied the soft edge network [6] as a denoiser for image deblurring and denoising and gave a mathematical interpretation of the PnP-based model. Overall, the PnP-based framework has a promising performance.

Motivated by the advantages of the aforementioned PnP models, in this paper, we are going to apply them to image inpainting problem for spherical signals. Precisely, for a spherical signal, with its partially observed samplings, a novel PnP model integrating spherical framelet decomposition is proposed to restore the signal. The proposed model is based on low rank assumption under directional spherical Haar tight framelet, which is designed for testing image texture. In addition, we exploit a newly designed deep convolutional neural network to be the plug-and-play prior denoiser. The network inspired by [16] and [12] employs two-stage encoder-decoder architecture, which is termed as Double-S2HaarNet. Under ground-truth supervision at each stage it provides progressive and improved denoising.

The rest of this paper is organized as follows. In section 2, the related works about spherical signal sampling and frame decomposition are reviewed. The proposed scheme and numerical algorithm are given in section 3. Numerical results including gray image and color image inpainting are listed in section 4. Section 5 concludes this paper.

2. Spherical signal sampling and frame decomposition.

We employ a Haar tight framelet transform that developed in [16]. Let $L_2(\mathbb{S}^2)$ be a Hilbert space with inner product $\langle \cdot, \cdot \rangle$ and

norm $\|\cdot\|$ defined by

$$\langle f, g\rangle := \int_\Omega f(x)g(x)dx,$$

$$\|f\| = \left(\int_\Omega |f(x)|^2 dx\right)^{\frac{1}{2}},$$

where $f, g \in L_2(\mathbb{S}^2)$ and $\mathbb{S}^2 \in \mathbb{R}^3$ is the unit sphere. We call a countable collection $\{e_k\}_{k\in\Lambda} \subset L_2(\mathbb{S}^2)$ a *tight frame* with frame bound c if there exists a constant $c > 0$ such that

$$f = \frac{1}{c}\sum_{k\in\Lambda}\langle f, e_k\rangle e_k \quad \forall\ f \in L_2(\mathbb{S}^2).$$

The frame decomposition is a transformation $\mathcal{F}$ given by

$$\mathcal{F} : f \in L_2(\mathbb{S}^2) \to \{\langle f, e_i\rangle : e_i \in \{e_k\}_{k\in\Lambda}\},$$

and the reconstruction $\mathcal{F}^*$

$$\mathcal{F}^* : \{\langle f, e_i\rangle : e_i \in \{e_k\}_{k\in\Lambda}\} \to f \in L_2(\mathbb{S}^2).$$

A Haar tight frame on the sphere can be constructed based on a hierarchical partition.

Definition 2.1. *Let $\mathbb{N}_0$ be a set of nonnegative integers. We call $\{\mathcal{B}_j\}_{j\in\mathbb{N}_0}$ a* hierarchical partition *of $\mathbb{S}^2$ if the following three conditions are satisfied:*

a) Root property: $\mathcal{B}_0 = \{\mathbb{S}^2\}$ and each $\mathcal{B}_j$ is a partition of $\mathbb{S}^2$ having finitely many measurable sets with positive measures.

b) Nested property: for any $j \in \mathbb{N}$ and any (child) set $R_1 \in \mathcal{B}_j$, there exists a (parent) set $R_0 \in \mathcal{B}_{j-1}$ such that $R_1 \subseteq R_0$. In other word, partition $\mathcal{B}_j$ is a refinement of the partition $\mathcal{B}_{j-1}$.

c) Density property: the maximal diameters among the sets in $\mathcal{B}_j$ tend to zero as j tends to infinity.

Denote $\Lambda_j := [\ell_1] \times \cdots \times [\ell_j]$ to be an index set for the labeling sets in $\mathcal{B}_j$, where $[N] = \{1, \ldots, N\}$ for any positive integer N and

$$\mathcal{B}_j = \{R_{\vec{v}} \subseteq \mathbb{S}^2,\ \ \vec{v} \in \Lambda_j\}.$$

By the nested property, $R_{(\vec{v},i)} \subseteq R_{\vec{v}}$ for $\vec{v} \in \Lambda_{j-1}$ and $i \in [\ell_j]$. Now, for each $\vec{v} \in \Lambda_j$, define a Haar-type scaling function to be

$$\phi_{\vec{v}} := \frac{\chi_{R_{\vec{v}}}}{\sqrt{|R_{\vec{v}}|}}, \tag{2.1}$$

and for some integer $n_j \geq 1$, n_j Haar-type framelet functions to be

$$\psi_{(\vec{v},k)} = \sum_{i\in[\ell_j]} a_{k,i}^{(\vec{v})} \phi_{(\vec{v},i)}, \quad k = 1, \ldots, n_j, \tag{2.2}$$

where $a_{k,i}^{(\vec{v})}$ is the (k,i)-entry of some matrix $\boldsymbol{A}_{\vec{v}} \in \mathbb{R}^{n_j \times l_j}$. By setting proper matrices $A_{\vec{v}}$, one can construct a Haar-type tight frame and develop its fast decomposition and reconstruction algorithms. The following corollary determines the framelet and algorithm we shall use.

Corollary 2.2. *There exists a collection* $\{\phi_{\vec{u}}\}_{\vec{u}\in\Lambda_L} \cup \{\psi_{(\vec{v},k)}, k \in [6]\}_{j\geq L, \vec{v}\in\Lambda_j} \subset L_2(\mathbb{S}^2)$ *determined by a hierarchical partition with each parent containing four children that forms a Haar tight frame with frame bound* 1*, and the corresponding operators* $\mathcal{F}$ *and* $\mathcal{F}^*$ *depend on the following matrix* P

$$P = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix}. \tag{2.3}$$

As it is well known, computers can only deal with discrete signals. To do spherical signal processing, we first need a proper way to discretize an analog signal. In this work, we take the discretization sampling method based on an area-regular partition of 2-sphere [16]. It was constructed through a bijective mapping and its rotations: $T : [-1,1] \times [-1,1] \to \mathbb{S}^2$ defined by $T(x,y) = \frac{(x,y,1)}{\sqrt{x^2+y^2+1}}$. See Figure 1 below for the illustration.

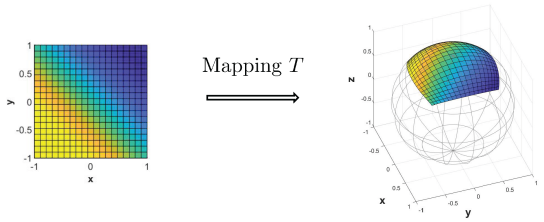


Figure 1. *Visualization of mapping* T *which maps a square to a spherical cap.*

Then for any given resolution $J \geq 0$, a 2-sphere can be divided into equal-area partitions, see Figure 2 for illustration. This forms an algorithm for a hierarchical partition on the 2-sphere.

By taking the centers of the partition patches, the samplings of an analog signal can be distributed equivalently, which takes advantage over the traditional spherical coordinates discretization.

Based on the above discussion, any signal $f \in L_2(\mathbb{S}^2)$ is discretized to $\boldsymbol{f}$, which depends on a certain resolution J. The discrete signal $\boldsymbol{f}$ is actually the set $\{f(\boldsymbol{\xi}_i) : \boldsymbol{\xi}_i \in S_i, S_i \in \mathcal{B}_j, \bigcup_i S_i =$

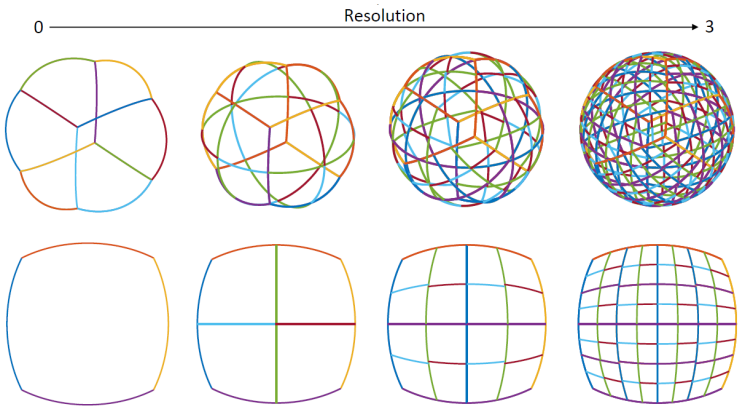


Figure 2. *Partition Process.*

$\mathbb{S}^2, S_j \cap S_k = \emptyset, \forall j \neq k\}$. We assume that the dataset is defined on some resolution level in the following and assume the sampling $\boldsymbol{f}$ of f is in $\mathbb{R}^d$. Applying the discretization and Corollary 2.2, the spherical Haar framelet and fast framelet transform algorithm are exactly constructed.

With the help of the fast decomposition and reconstruction algorithms, it, on the one hand, allows our model to capture directional texture details. On the other hand, it can reduce the spatial footprint and granularity of convolutions.

3. The proposed model and algorithm. In this section, based on the aforementioned off-the-shelf spherical signal sampling and Haar-type framelets, to enhance the inpainting performance, we shall first improve the denoiser and then exploit a plug-and-play model involving fast frame decomposition.

3.1. Improved Denoiser. As mentioned above, the iteration (1.4) will be conducted by a denoiser. The performance of the denoiser will affect the resulting restoration. In [16], a CNN spherical denoising model, S2HaarNet, was developed and achieved a competitive performance. In the present paper, we further exploit a new spherical CNN (illustrated in Figure 3), which partly follows the infrastructure of the Double-Unet [12] and S2HaarNet [16], and incorporates the skip connections and spherical frame transformations. Thus we shall call it Double-S2HaarNet. The new network consists of two feature encoder-decoder stages for which each one follows S2HaarNet. We then take ground-truth supervision at each stage for progressive and improved denoising performance. We adopt the feature concatenation by combining the feature maps from the encoder path and decoder path, which can capture multi-scale information and enrich feature representation for a better feature prediction. To bridge the two blocks, we concentrate the input and output of the first block and feed it into the second. Our model can also be readily extended to deal with color images by handling three channels independently. For the designed models, we also present the computational cost. As we know that the indexes that are often used to measure the complexity of deep learning models include parameters, FLOPs (Floating-point Operations), and running time.

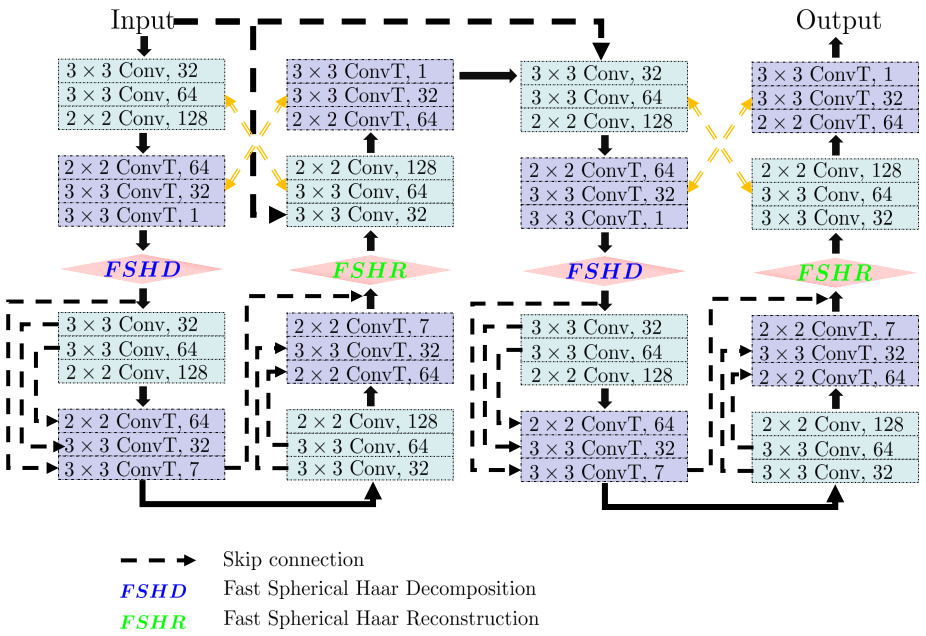


Figure 3. *Double-S2HaarNet.*

More specifically, the parameters refer to the total number of parameters that need to be trained during model training, which is used to measure the computational space complexity. FLOPs is often used as an indirect measure of the speed of the neural network model. The complexity of the proposed networks S2HaarNet and Double-S2HaarNet are listed in Table 1.

Table 1
The complexity (parameters (in MB), Flops (in GB), running times (in seconds)) of the proposed models.

Methods	Parameters	Flops	Running time
S2HaarNet	0.02MB	5.30G	0.0020s
Double-S2HaarNet	0.63MB	199.45G	0.1310s

3.2. Proposed PnP model. To develop a PnP model, besides the above pre-trained denoiser, a proper design for data fitting subproblem plays a crucial role as well. In this work, we attempt to propose a new data fitting operator to simultaneously exploit the strengths of the PnP model. In [31], data fitting was operated by using the tensor singular value decomposition and tensor nuclear norm, which promoted the low-rankness of the underlying tensor. Motivated by such an idea, we instead utilize the tight frame decomposition in our data fitting

and suggest the novel PnP model as follows:

$$\begin{aligned} &\min_{\boldsymbol{x}} \|\mathcal{F}\boldsymbol{x}\|_0 + \lambda\Phi(\boldsymbol{x}) \\ &\text{s.t.} \quad \mathcal{P}_\gamma(\boldsymbol{x}) = \mathcal{P}_\gamma(\boldsymbol{g}), \end{aligned} \tag{3.1}$$

where $\mathcal{F}\boldsymbol{x}$ is the coefficients of tight frame decomposition as mentioned in subsection 3.1, $\Phi(\boldsymbol{x})$ is an implicit regularizer by plugging our denoising Double-S2HaarNet, $\mathcal{P}_\gamma$ is a pre-defined observation operator, and λ is a positive parameter. The observation operator is defined with respect to a prior-given subset $\gamma \subset [d]$, $|\gamma| = m \leq d$, which refers to locations of known observation points. Precisely, for any signal $x \in L_2(\mathbb{S}^2)$ and its discretization $\boldsymbol{x} \in \mathbb{R}^d$, $\mathcal{P}_\gamma(\boldsymbol{x}) := \boldsymbol{O}\boldsymbol{x}$, where $\boldsymbol{O} \in \mathbb{R}^{d\times d}$ with the kth row of $\boldsymbol{O}$ being the kth one-hot row for $k \in \gamma$ and the remaining rows being zero. In addition, we denote $\mathcal{P}_{\gamma^c}(\boldsymbol{x}) := (\boldsymbol{I} - \boldsymbol{O})\boldsymbol{x}$.

3.3. Implementation Details. We apply ADMM framework to solve the optimization problem. Notice that in practice, it is more convenient to replace $\|\cdot\|_0$ by $\|\cdot\|_1$. Then we reformulate model (3.1) as

$$\begin{aligned} &\min_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}} \|\boldsymbol{y}\|_1 + \lambda\Phi(\boldsymbol{z}) \\ &\text{s.t.} \quad \mathcal{P}_\gamma(\boldsymbol{x}) = \mathcal{P}_\gamma(\boldsymbol{g}),\ \boldsymbol{y} = \mathcal{F}\boldsymbol{x},\ \boldsymbol{z} = \boldsymbol{x}. \end{aligned} \tag{3.2}$$

The augmented Lagrangian function of (3.2) is

$$\begin{aligned} \mathcal{L}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z};\Lambda_1,\Lambda_2) =& \|\boldsymbol{y}\|_1 + \lambda\Phi(\boldsymbol{z}) + \langle \boldsymbol{y} - \mathcal{F}\boldsymbol{x}, \Lambda_1\rangle + \frac{\beta_1}{2}\|\boldsymbol{y} - \mathcal{F}\boldsymbol{x}\|^2 + \langle \boldsymbol{z} - \boldsymbol{x}, \Lambda_2\rangle + \frac{\beta_2}{2}\|\boldsymbol{z} - \boldsymbol{x}\|^2, \\ &\text{s.t.} \quad \mathcal{P}_\Gamma(\boldsymbol{x}) = \mathcal{P}_\Gamma(\boldsymbol{g}), \end{aligned} \tag{3.3}$$

where $\beta_1, \beta_2 > 0$ are two penalty parameters and Λ_1, Λ_2 are the Lagrange multipliers. The ADMM iteration for solving (3.3) goes as follows,

$$\begin{cases} \boldsymbol{y} = \arg\min_{\boldsymbol{y}} \|\boldsymbol{y}\|_1 + \langle \boldsymbol{y} - \mathcal{F}\boldsymbol{x}, \Lambda_1\rangle + \frac{\beta_1}{2}\|\boldsymbol{y} - \mathcal{F}\boldsymbol{x}\|^2, \\ \boldsymbol{z} = \arg\min_{\boldsymbol{z}} \lambda\Phi(\boldsymbol{z}) + \langle \boldsymbol{z} - \boldsymbol{x}, \Lambda_2\rangle + \frac{\beta_2}{2}\|\boldsymbol{z} - \boldsymbol{x}\|^2, \\ \begin{cases} \boldsymbol{x} = \arg\min_{\boldsymbol{x}} \frac{\beta_1}{2}\|\boldsymbol{y} - \mathcal{F}\boldsymbol{x} + \frac{\Lambda_1}{\beta_1}\|^2 + \langle \boldsymbol{z} - \boldsymbol{x}, \Lambda_2\rangle + \frac{\beta_2}{2}\|\boldsymbol{z} - \boldsymbol{x}\|^2, \\ \text{s.t.} \quad \mathcal{P}_\gamma(\boldsymbol{x}) = \mathcal{P}_\gamma(\boldsymbol{g}), \end{cases} \\ \Lambda_1 = \Lambda_1 + (\boldsymbol{y} - \mathcal{F}\boldsymbol{x}), \\ \Lambda_2 = \Lambda_2 + (\boldsymbol{z} - \boldsymbol{x}). \end{cases} \tag{3.4}$$

Next, we elaborate on how to solve these subproblems respectively.

- The $\boldsymbol{y}$-subproblem is written as

$$\begin{aligned} \boldsymbol{y} &= \arg\min_{\boldsymbol{y}} \|\boldsymbol{y}\|_1 + \langle \boldsymbol{y} - \mathcal{F}\boldsymbol{x}, \Lambda_1\rangle + \frac{\beta_1}{2}\|\boldsymbol{y} - \mathcal{F}\boldsymbol{x}\|^2 \\ &= \arg\min_{\boldsymbol{y}} \|\boldsymbol{y}\|_1 + \frac{\beta_1}{2}\left\|\boldsymbol{y} - \mathcal{F}\boldsymbol{x} + \frac{\Lambda_1}{\beta_1}\right\|^2. \end{aligned} \tag{3.5}$$

Then the solution of y can be obtained by

$$\begin{aligned} \boldsymbol{y} &= \text{shrink}\left(\mathcal{F}\boldsymbol{x} - \frac{\Lambda_1}{\beta_1}, \frac{1}{\beta_1}\right) \\ &= \max\left(\left\|\mathcal{F}\boldsymbol{x} - \frac{\Lambda_1}{\beta_1}\right\|_2 - \frac{1}{\beta_1}, 0\right) \frac{\mathcal{F}\boldsymbol{x} - \frac{\Lambda_1}{\beta_1}}{\left\|\mathcal{F}\boldsymbol{x} - \frac{\Lambda_1}{\beta_1}\right\|}, \end{aligned} \tag{3.6}$$

where the shrink operator is a soft shrinkage operator.

- The $\boldsymbol{z}$-subproblem is written as

$$\begin{aligned} \boldsymbol{z} &= \arg\min_{\boldsymbol{z}} \lambda\Phi(\boldsymbol{z}) + \langle \boldsymbol{z} - \boldsymbol{x}, \Lambda_2\rangle + \frac{\beta_2}{2}\|\boldsymbol{z} - \boldsymbol{x}\|^2 \\ &= \arg\min_{\boldsymbol{z}} \lambda\Phi(\boldsymbol{z}) + \frac{\beta_2}{2}\left\|\boldsymbol{z} - \boldsymbol{x} + \frac{\Lambda_2}{\beta_2}\right\|^2, \end{aligned} \tag{3.7}$$

According to Bayes rule, Eq. (3.7) corresponds to denoising the image $\boldsymbol{x} - \Lambda_2/\beta_2$ by the CNN denoiser with noise level λ/β_2. To address this, we rewrite Eq. (3.7) as

$$\boldsymbol{z} = \text{Denoiser}(\boldsymbol{x} - \frac{\Lambda_2}{\beta_2}, \sqrt{\frac{\lambda}{\beta_2}}). \tag{3.8}$$

In this paper, we apply the Double-S2HaarNet as the denoiser.

- The $\boldsymbol{x}$-subproblem is written as

$$\begin{aligned} \boldsymbol{x} &= \arg\min_{\boldsymbol{x}} \frac{\beta_1}{2}\left\|\boldsymbol{y} - \mathcal{F}\boldsymbol{x} + \frac{\Lambda_1}{\beta_1}\right\|^2 + \langle \boldsymbol{z} - \boldsymbol{x}, \Lambda_2\rangle + \frac{\beta_2}{2}\|\boldsymbol{z} - \boldsymbol{x}\|^2 \\ &= \arg\min_{\boldsymbol{x}} \frac{\beta_1}{2}\left\|\boldsymbol{y} - \mathcal{F}\boldsymbol{x} + \frac{\Lambda_1}{\beta_1}\right\|^2 + \frac{\beta_2}{2}\left\|\boldsymbol{z} - \boldsymbol{x} + \frac{\Lambda_2}{\beta_2}\right\|^2. \\ &\text{s.t.} \quad \mathcal{P}_\gamma(\boldsymbol{x}) = \mathcal{P}_\gamma(\boldsymbol{g}), \end{aligned} \tag{3.9}$$

By minimizing the $\boldsymbol{x}$-subproblem, we have $\mathbf{1}_\Omega(\boldsymbol{x}) = 0$, i.e., $\boldsymbol{x} \in \mathbb{S}$. Then optimality condition of (3.9) is given by

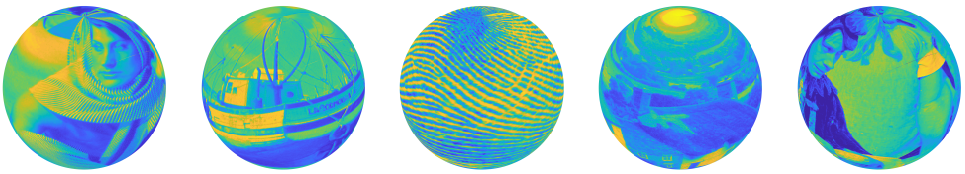
$$\beta_1\mathcal{F}^*(\mathcal{F}\boldsymbol{x} - \boldsymbol{y} - \frac{\Lambda_1}{\beta_1}) + \beta_2(\boldsymbol{x} - \boldsymbol{z} - \frac{\Lambda_2}{\beta_2}) = 0. \tag{3.10}$$

Since $\mathcal{F}^*\mathcal{F} = I$,we obtain the following linear system,

$$(\beta_1 + \beta_2)\boldsymbol{x} = \beta_1\mathcal{F}^*\boldsymbol{y} + \beta_2\boldsymbol{z} + \mathcal{F}^*\Lambda_1 + \Lambda_2. \tag{3.11}$$

Thus, the closed-form solution of $\boldsymbol{x}$-subproblem is given as $\boldsymbol{x}^* = \mathcal{P}_\gamma(\boldsymbol{x}^*) + (\mathcal{P}_{\gamma^c})(\boldsymbol{x}^*)$:

$$\begin{cases} \mathcal{P}_\gamma(\boldsymbol{x}^*) = \mathcal{P}_\gamma(\boldsymbol{g}), \\ \mathcal{P}_{\gamma^c}(\boldsymbol{x}^*) = \mathcal{P}_{\gamma^c}\left(\frac{\beta_1\mathcal{F}^*\boldsymbol{y} + \beta_2\boldsymbol{z} + \mathcal{F}^*\Lambda_1 + \Lambda_2}{\beta_1 + \beta_2}\right). \end{cases} \tag{3.12}$$



(a) Barbara (b) Boat (c) Fingerprint (d) Hill (e) Man

Figure 4. *Five grey images for testing.*

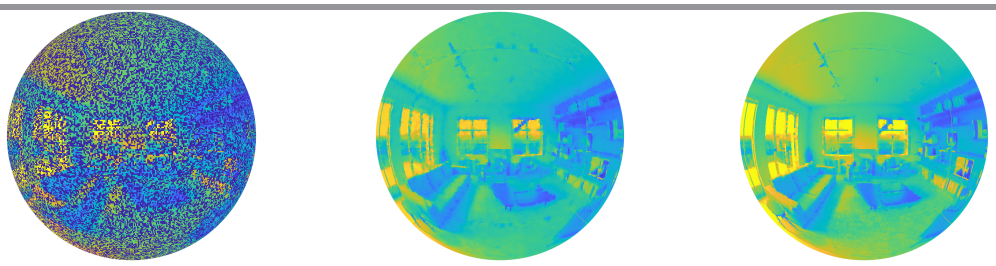
Table 2

Average inpainting results with PSNR/SSIM on F-360iSOD. In Fx, x = 6,7,8, represents the resolution level. The best results are highlighted.

dataset	methods	50%	80%	90%	95%
F6	degraded	10.08/0.2087	8.04/0.0806	7.52/0.0445	7.29/0.0268
	S2HaarNet	23.02/0.7711	20.27/0.5777	19.16/0.4962	18.02/0.4175
	S2HaarNetPnP	23.96/0.8125	20.81/0.6024	19.47/0.5225	18.33/0.4290
	DoubleS2HaarNet	24.22/0.8321	20.99/0.6470	19.55/0.5367	18.41/0.4568
	DoubleS2HaarNetPnP	**24.63/0.8470**	**21.49/0.6756**	**19.96/0.5495**	**18.63/0.4596**
F7	degraded	10.08/0.1857	8.04/0.0776	7.53/0.0454	7.29/0.0290
	S2HaarNet	23.94/0.7822	21.12/0.5920	20.03/0.5125	18.89/0.4380
	S2HaarNetPnP	24.92/0.8213	21.72/0.6200	20.33/0.5445	19.01/0.4553
	DoubleS2HaarNet	25.15/0.8349	21.83/0.6590	20.41/0.5506	19.30/0.4750
	DoubleS2HaarNetPnP	**26.14/0.8623**	**22.63/0.7009**	**20.93/0.5901**	**19.60/0.5007**
F8	degraded	10.08/0.1660	8.04/0.0755	7.53/0.0469	7.29/0.0313
	S2HaarNet	26.86/0.8402	23.79/0.6965	22.49/0.6337	21.05/0.5633
	S2HaarNetPnP	27.35/0.8613	24.18/0.7313	22.84/0.6667	21.49/0.5953
	DoubleS2HaarNet	28.55/0.8918	24.76/0.7607	23.03/0.6751	21.64/0.6006
	DoubleS2HaarNetPnP	**28.93/0.8961**	**25.02/0.7713**	**23.31/0.6862**	**21.87/0.6167**

4. Experimental results. In this section, we present experimental results to verify the performance of the proposed model Double-S2HaarNetPnP in image inpainting. As aforementioned, we proposed a plug-and-play model for the image inpainting task. The parameter of the optimization function (3.3) are set as $\lambda = 1$, $\beta_1 \in [0.1, 1]$ with step 0.1, $\beta_2 \in [1, 5]$ with step 1. For training Double-S2HaarNet, we use the ADAM algorithm and a mini-batch size of 16. The learning rate decays exponentially from the beginning value 0.001 with a multiplicative factor 0.9 in 100 epochs. Weight decay is chosen to be 0.001. Since the contrast of grayscale images is relatively low, we present the visual effects of the image with color so that the image information can be displayed more clearly.

4.1. Datasets. The dataset for training CNN denoisers is produced by converting color images of Caltech101 [8] into grey ones and then conducting spherical sampling (defined in section 2). 7677 images are for training and 1000 are for validation. For the testing, we take the dataset F-360iSOD [30] which contains 107 omnidirectional images. Additionally, we take five classical images as illustrated in Figure 4 for testing as well. Experiments for color images are conducted on RGB channels, respectively.



(a) degraded 7.12/0.0532 (b) S2HaarNet 28.32/0.8735 (c) S2HaarNetPnP 30.15/0.9349

(d) Double-S2HaarNet 30.87/0.9346 (e) Double-S2HaarNetPnP 31.87/0.95542 (f) Ground-truth infinity/1

Figure 5. *The inpainting results (PSNR (dB)/SSIM) with random missing ratio 50%. (a) the degraded image; the recovered results of (b) Haar network only; (c) plug and play with Haar network; (d) Double-S2Haar network only; (e) plug and play with Double-S2Haar network; (f) the original image.*

4.2. Evaluation metrics. To demonstrate the effectiveness of the proposed scheme, we test the images with random missing values, i.e., the missing ratio with $50\%, 80\%, 90\%, 95\%$. With the built-in function in numpy, we use the command 'numpy.random.rand' to generate random values of the same size as the input image. Then let the value which is great than the missing ratio equal to 1 and less than the missing ratio equal to 0. Then we have the missing operator, with the function 'numpy.multiply' mapping the missing operator and the groundtruth image to the observed image. The peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) are used to evaluate the performance of the inpainting results. To be specific, with the reference x and the obtained result x^*, the PSNR is defined as

$$\mathrm{PSNR}(x, x^*) = 20 \log_{10} \frac{255}{\frac{1}{mn} \|x^\star - x\|}, \tag{4.1}$$

where $\|\cdot\|$ denotes a Frobenius norm. The SSIM is defined as

$$\mathrm{SSIM}\,(x, x^\star) = \frac{(2\mu_x \mu_{x^\star} + C_1)\,(2\sigma_{xx^\star} + C_2)}{\left(\mu_x^2 + \mu_{x^\star}^2 + C_1\right)\left(\sigma_x^2 + \sigma_{x^\star}^2 + C_2\right)} \tag{4.2}$$

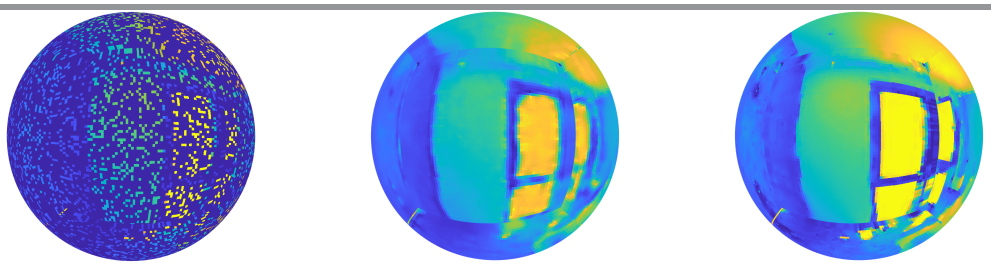
where μ_x, μ_{x^*} and σ_x, σ_{x^*}, σ_{xx^*} are the mean and the standard deviation of x and x^*, respectively. The positive constants C_1 and C_2 are used to avoid a null denominator, which are defaulted by the build-in ssim function.

Table 3
Inpainting results with PSNR/SSIM. The best results are highlighted.

Images	methods	50%	80%	90%	95%
Barbara	degraded	8.54/0.1039	6.50/0.0466	5.99/0.0281	5.75/0.0169
	S2HaarNet	26.91/0.8523	23.77/0.7092	22.57/0.6386	21.30/0.5653
	S2HaarNetPnP	27.56/0.9043	24.36/0.7547	23.14/0.6834	21.70/0.5731
	Double-S2HaarNet	28.81/0.9153	24.79/0.7732	23.17/0.6788	21.87/0.6083
	Double-S2HaarNetPnP	**29.14/0.9283**	**25.15/0.7895**	**23.75/0.6836**	**22.21/0.6799**
Boat	degraded	8.57/0.0976	6.52/0.0453	6.02/0.0281	5.78/0.0172
	S2HaarNet	30.33/0.8569	26.94/0.7618	25.08/0.6995	22.84/0.6158
	S2HaarNetPnP	32.04/0.9202	28.08/0.8079	25.77/0.7141	23.16/0.6372
	Double-S2HaarNet	32.69/0.9208	28.20/0.8099	25.83/0.7371	23.60/0.6554
	Double-S2HaarNetPnP	**33.12/0.9299**	**28.81/0.8234**	**26.63/0.7567**	**23.84/0.6946**
Fingerprint	degraded	7.39/0.1268	5.35/0.0477	4.83/0.0254	4.60/0.0136
	S2HaarNet	27.53/0.9165	23.88/0.8183	21.28/0.7101	17.90/0.4914
	S2HaarNetPnP	29.10/0.9465	24.62/0.8294	21.79/0.7342	18.39/0.5394
	Double-S2HaarNet	29.50/0.9485	25.06/0.8596	22.26/0.7630	18.92/0.5768
	Double-S2HaarNetPnP	**30.09/0.9552**	**25.51/0.8684**	**22.78/0.7773**	**19.38/0.6392**
Hill	degraded	10.25/0.0876	8.21/0.0444	7.69/0.0286	7.47/0.0182
	S2HaarNet	31.28/0.8541	28.06/0.7314	26.52/0.6656	24.68/0.5880
	S2HaarNetPnP	32.81/0.9076	28.74/0.7739	27.00/0.6909	25.17/0.5973
	Double-S2HaarNet	33.29/0.9119	29.15/0.7836	27.15/0.6991	25.48/0.6266
	Double-S2HaarNetPnP	**33.60/0.9187**	**29.54/0.7947**	**27.51/0.6994**	**25.93/0.6586**
Man	degraded	9.35/0.0830	7.31/0.0408	6.80/0.0266	6.56/0.0166
	S2HaarNet	31.08/0.8782	27.85/0.7833	26.20/0.7242	24.30/0.6521
	S2HaarNetPnP	32.70/0.9293	28.65/0.8204	26.77/0.7391	24.76/0.6733
	Double-S2HaarNet	33.00/0.9303	28.99/0.8295	26.93/0.7588	25.03/0.6890
	Double-S2HaarNetPnP	**33.37/0.9375**	**29.38/0.8369**	**27.12/0.7670**	**25.27/0.6903**

4.3. Results. We make a detailed comparison of our method. More specifically, the methods with single net (S2HaarNet and DoubleS2haarNet) and the methods with plug-and-play (S2HaarNetPnP and DoubleS2haarNetPnP) are compared. First of all, the three datasets are tested. The results of datasets 'F6', 'F7', and 'F8' are listed in Table 2, from which we know that the proposed methods are robust in different degradation. For example, when the missing ratio is from 50% up to 95%, our methods always have competitive restoration results. Besides, the results based on the plug-and-play approach are better than the one with only CNN, which also illustrate the effectiveness of the proposed plug-and-play scheme. Moreover, the testing of two different datasets and different combinations of the proposed model also gives a strong validity to our scheme. On the other hand, five grey images are also tested in this paper. We list the numerical results in Table 3, from which the plug-and-play-based models also have better restoration results.

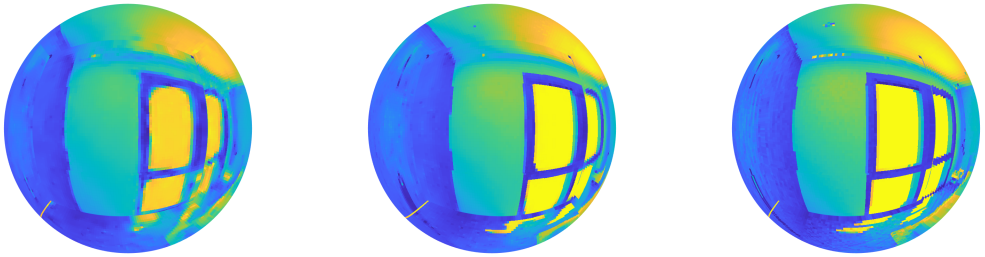
The visual results are present in Figures 5–8 with missing ratio $50\%, 80\%, 90\%, 95\%$, respectively. Figure 5 (a) is the observed image with a low missing ratio (50%). As we can see from the results (b)–(e), most of the objects in the image are recovered. However, with detailed observation, we know that the plug-and-play-based methods have more competitive performance. With the missing ratio up to 80%, there are some outlines of the original image that



(a) degraded
7.21/0.0612

(b) S2HaarNet
21.78/0.7525

(c) S2HaarNetPnP
22.33/0.8095



(d) Double-S2HaarNet
23.38/0.8227

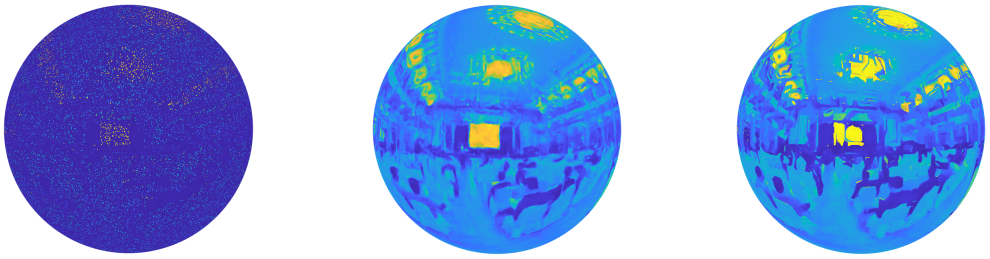
(e) Double-S2HaarNetPnP
23.68/0.8402

(f) Ground-truth
infinity/1

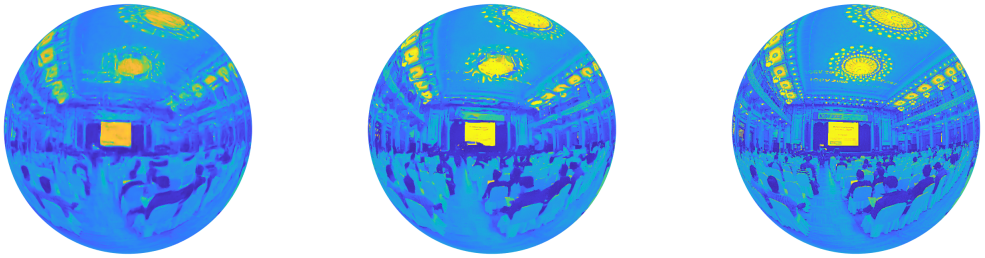
Figure 6. *The inpainting results (PSNR (dB)/SSIM) with random missing ratio 80%. (a) the degraded image; the recovered results of (b) Haar network only; (c) plug and play with Haar network; (d) Double-S2Haar network only; (e) plug and play with Double-S2Haar network; (f) the original image.*

can be seen in Figure 6 (a). It turns out that the results of the S2HaarNet and Double-S2HaarNet have quite satisfactory results. As the plug-and-play is applied in (b) and (d) respectively, the results of inpainting are greatly improved. For the low sample ratio, from Figure 7 (a) and Figure 8 (a), the details of the image are almost disappeared. With this low observation, our models also can recover the images with good quality. The above visual results demonstrate again the robustness and effectiveness of the proposed inpainting models.

The experiments on color images are also conducted in Figures 9–12, which illustrate the good generalization of our model to color images. Note that our results are slightly over-smoothed for lower sample rates, such as 95%. Figure 12 shows that, in contrast to other results, our solutions can better restore the structures of images, which is consistent with the results of the majority of plug-and-play-based works.



(a) degraded 7.77/0.0284 (b) S2HaarNet 19.12/0.5090 (c) S2HaarNetPnP 19.34/0.5303



(d) Double-S2HaarNet 19.40/0.5446 (e) Double-S2HaarNetPnP 19.68/0.6557 (f) Ground-truth infinity/1

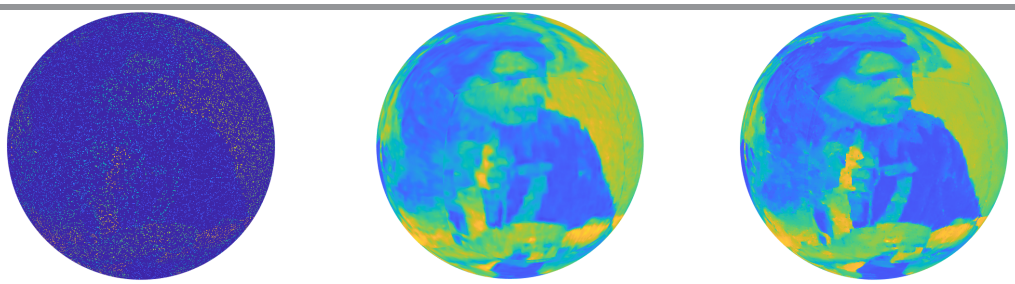
Figure 7. *The inpainting results (PSNR (dB)/SSIM) with random missing ratio 90%. (a) the degraded image; the recovered results of (b) Haar network only; (c) plug and play with Haar network; (d) Double-S2Haar network only; (e) plug and play with Double-S2Haar network; (f) the original image.*

Color1

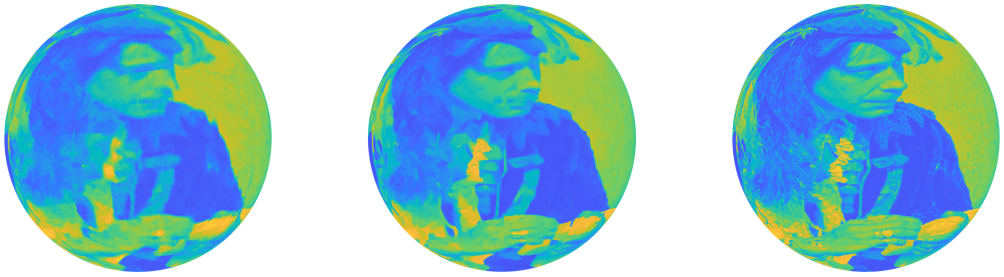
(a) degraded 8.39/0.0473 (b) S2HaarNet 31.53/0.9458 (c) S2HaarNetPnP 32.96/0.9551

(d) Double-S2Haar 36.98/0.9850 (e) Double-S2HaarPnP 37.23/0.9856 (f) Ground-truth infinity/1

Figure 9. *The inpainting results (PSNR (dB)/SSIM) with random missing ratio 50%. (a) the degraded image; the recovered results of (b) Haar network only; (c) plug and play with Haar network; (d) Double-S2Haar network only; (e) plug and play with Double-S2Haar network; (f) the original image.*



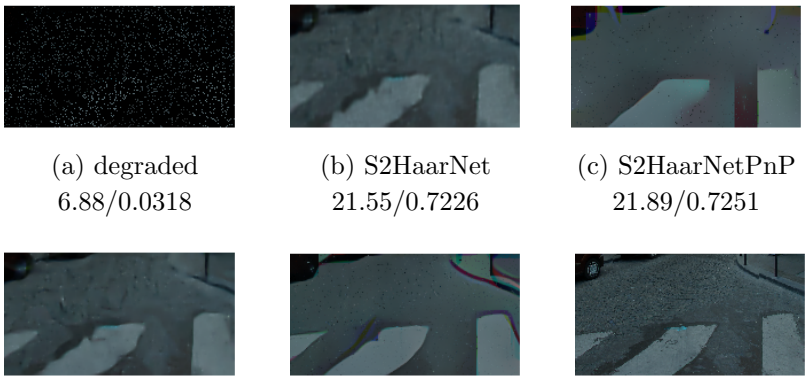
(a) degraded 6.56/0.0166 (b) S2HaarNet 24.30/0.6521 (c) S2HaarNetPnP 24.76/0.6733



(d) Double-S2HaarNet 25.03/0.6890 (e) Double-S2HaarNetPnP 25.27/0.6903 (f) Ground-truth infinity/1

Figure 8. *The inpainting results (PSNR (dB)/SSIM) with random missing ratio 95%. (a) the degraded image; the recovered results of (b) Haar network only; (c) plug and play with Haar network; (d) Double-S2Haar network only; (e) plug and play with Double-S2Haar network; (f) the original image.*

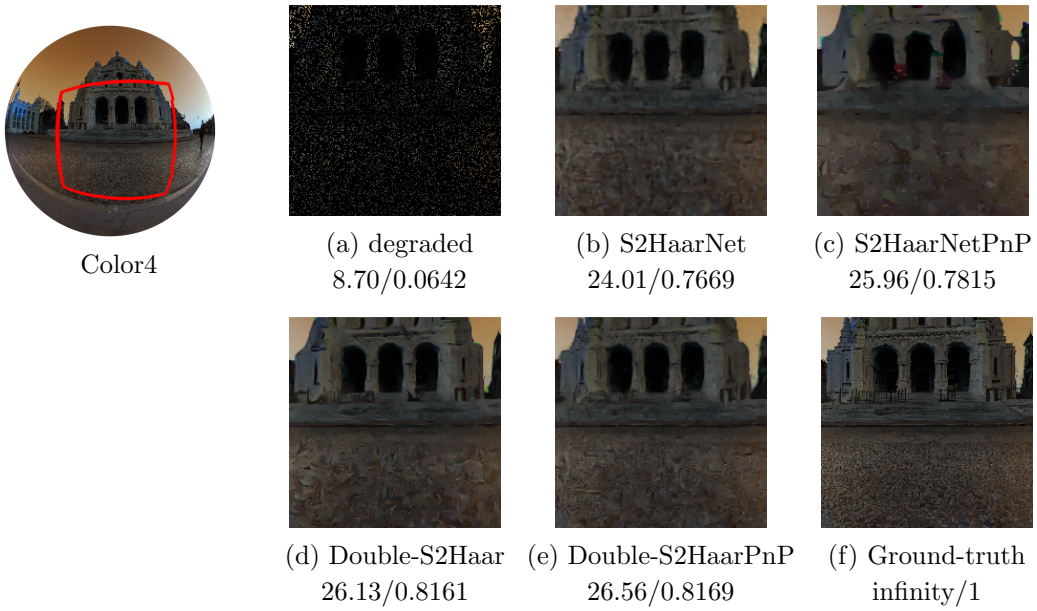
Color12



(a) degraded 6.88/0.0318 (b) S2HaarNet 21.55/0.7226 (c) S2HaarNetPnP 21.89/0.7251

(d) Double-S2Haar 21.9763/0.7485 (e) Double-S2HaarPnP 22.27/0.7666 (f) Ground-truth infinity/1

Figure 11. *The color image inpainting results (PSNR (dB)/SSIM) with random missing ratio 90%. Zoomed part of (a) the degraded image; the recovered results of (b) Haar network only; (c) plug and play with Haar network; (d) Double-S2Haar network only; (e) plug and play with Double-S2Haar network; (f) the original image.*



Color4

(a) degraded 8.70/0.0642

(b) S2HaarNet 24.01/0.7669

(c) S2HaarNetPnP 25.96/0.7815

(d) Double-S2Haar 26.13/0.8161

(e) Double-S2HaarPnP 26.56/0.8169

(f) Ground-truth infinity/1

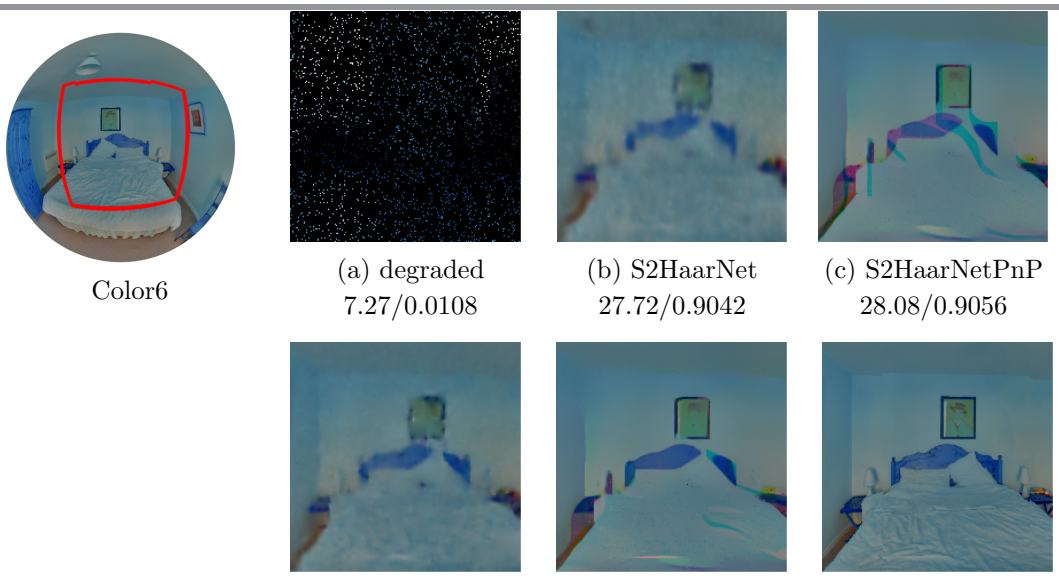
Figure 10. *The color image inpainting results (PSNR (dB)/SSIM) with random missing ratio 80%. Zoomed part of (a) the degraded image; the recovered results of (b) Haar network only; (c) plug and play with Haar network; (d) Double-S2Haar network only; (e) plug and play with Double-S2Haar network; (f) the original image.*

Table 4

Average and variance inpainting results of Double-S2HaarNet variants with PSNR/SSIM on F-360iSOD. In Fx, x=6,7,8, represents the resolution level.

Dataset	methods	50%	80%	95%
F6	avgpool+bilinear	23.50/0.8151	20.71/0.6310	18.29/0.4389
	maxpool+bilinear	**23.98/ 0.8359**	20.72/0.6328	18.36/0.4461
	random_filter	18.63/0.6733	15.77/0.4598	17.75/0.4016
	Our	23.83/0.8301	**20.97/0.6514**	**18.40/0.4479**
F7	avgpool+bilinear	25.58/0.8377	22.62/0.6785	20.11/0.5161
	maxpool+bilinear	**26.13/0.8573**	22.66/0.6828	20.19/0.5234
	random_filter	24.33/0.7634	19.96/0.5077	14.48/0.3999
	Our	25.93/0.8509	**22.96/0.7008**	**20.25/0.5252**
F8	avgpool+bilinear	27.86/0.8628	24.71/0.7283	22.00/0.5903
	maxpool+bilinear	**28.47/0.8810**	24.79/0.7344	22.11/0.5977
	random_filter	24.01/0.7545	20.20/0.6349	14.85/0.4915
	Our	28.24/0.8743	**25.16/0.7515**	**22.18/0.5992**

4.4. Ablation experiments. In this section, to analyze the improvement effect of introducing framelet transforms in image inpainting tasks, the following 3 variants of our model are designed for comparison: (a) random filters for constructing framelets, which is different from specifically designed directional framelets with respect to matrix (2.3); (b) max-pooling for downsampling and bilinear interpolation for upsampling; (c) average pooling for downsampling and bilinear interpolation for upsampling. The comparison results are illustrated



Color6

(a) degraded 7.27/0.0108

(b) S2HaarNet 27.72/0.9042

(c) S2HaarNetPnP 28.08/0.9056

(d) Double-S2Haar 28.84/0.9193

(e) Double-S2HaarPnP 29.09/0.9216

(f) Ground-truth infinity/1

Figure 12. *The color image inpainting results (PSNR (dB)/SSIM) with random missing ratio 95%. Zoomed part of (a) the degraded image; the recovered results of (b) Haar network only; (c) plug and play with Haar network; (d) Double-S2Haar network only; (e) plug and play with Double-S2Haar network; (f) the original image.*

in table 4, which shows the advantages by introducing framelet transforms, especially for the case of high inpainting rates.

5. Conclusion and further remarks. In this work, we presented a novel model, Double-S2HaarNetPnP model, for image inpainting. We remark that (a) low-rank framelet coefficient regularizer is introduced to learn, (b) a new denoiser Double-S2HaarNet for spherical image inpainting is proposed and in the experiments, it is powerful for inpainting task with the deep Plug-and-play framework, and (c) Directional spherical Haar framelet is employed to capture directional texture information to enhance the learning ability of the model and the neural network. Experiments evaluated on various images illustrate the performance of the proposed method for spherical image inpainting task. The main challenges of our work are to develop a proper hierarchical partition of a manifold and explore the corresponding convolution calculation. With them, the proposed model can be further generalized to other manifolds.

REFERENCES

[1] P. Audet, *Toward mapping the effective elastic thickness of planetary lithospheres from a spherical wavelet analysis of gravity and topography*, Physics of the Earth and Planetary Interiors, 226 (2014), pp. 48–82.

[2] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang, *Matterport3d: Learning from rgb-d data in indoor environments*, arXiv preprint arXiv:1709.06158, (2017).

[3] H. Chang, R. Glowinski, S. Marchesini, X.-C. Tai, Y. Wang, and T. Zeng, *Overlapping domain decomposition methods for ptychographic imaging*, SIAM Journal on Scientific Computing, 43 (2021), pp. B570–B597.

[4] B. Cheng, J. Li, Y. Chen, S. Zhang, and T. Zeng, *Snow mask guided adaptive residual network for image snow removal*, arXiv preprint arXiv:2207.04754, (2022).

[5] Y. Dong, P. C. Hansen, M. E. Hochstenbach, and N. A. Brogaard Riis, *Fixing nonconvergence of algebraic iterative reconstruction with an unmatched backprojector*, SIAM Journal on Scientific Computing, 41 (2019), pp. A1822–A1839.

[6] F. Fang, J. Li, and T. Zeng, *Soft-edge assisted network for single image super-resolution*, IEEE Transactions on Image Processing, 29 (2020), pp. 4656–4668.

[7] Y. Fang and T. Zeng, *Learning deep edge prior for image denoising*, Computer Vision and Image Understanding, 200 (2020), p. 103044.

[8] L. Fei-Fei, R. Fergus, and P. Perona, *Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories*, in 2004 conference on computer vision and pattern recognition workshop, IEEE, 2004, pp. 178–178.

[9] G. Gao, G. Xu, J. Li, Y. Yu, H. Lu, and J. Yang, *FBSNet: A fast bilateral symmetrical network for real-time semantic segmentation*, IEEE Transactions on Multimedia, (2022).

[10] R. Hou, F. Li, and G. Zhang, *Truncated residual based plug-and-play ADMM algorithm for MRI reconstruction*, IEEE Transactions on Computational Imaging, 8 (2022), pp. 96–108.

[11] C. Huang, Z. Li, Y. Liu, T. Wu, and T. Zeng, *Quaternion-based weighted nuclear norm minimization for color image restoration*, Pattern Recognition, 128 (2022), p. 108665.

[12] D. Jha, M. A. Riegler, D. Johansen, P. Halvorsen, and H. D. Johansen, *Doubleu-net: A deep convolutional neural network for medical image segmentation*, in 2020 IEEE 33rd International symposium on computer-based medical systems (CBMS), IEEE, 2020, pp. 558–564.

[13] Z. Jia and M. K. Ng, *Structure preserving quaternion generalized minimal residual method*, SIAM Journal on Matrix Analysis and Applications, 42 (2021), pp. 616–634.

[14] R. Ke, R. Wagner, R. Ramlau, and R. Chan, *Reconstruction of the high resolution phase in a closed loop adaptive optics system*, SIAM Journal on Imaging Sciences, 13 (2020), pp. 775–806.

[15] J. Li, F. Fang, T. Zeng, G. Zhang, and X. Wang, *Adjustable super-resolution network via deep supervised learning and progressive self-distillation*, Neurocomputing, (2022).

[16] J. Li, H. Feng, and X. Zhuang, *Convolutional neural networks for spherical signal processing via area-regular spherical haar tight framelets*, IEEE Transactions on Neural Networks and Learning Systems, (2022), pp. 1–11, https://doi.org/10.1109/TNNLS.2022.3160169.

[17] J. Li, H. Yang, Q. Yi, F. Fang, G. Gao, T. Zeng, and G. Zhang, *Multiple degradation and reconstruction network for single image denoising via knowledge distillation*, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 558–567.

[18] P. Liu, H. Zhang, K. Zhang, L. Lin, and W. Zuo, *Multi-level wavelet-cnn for image restoration*, in Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2018, pp. 773–782.

[19] F. Malgouyres and J. Landsberg, *Multilinear compressive sensing and an application to convolutional linear networks*, SIAM Journal on Mathematics of Data Science, 1 (2019), pp. 446–475.

[20] J. D. McEwen, P. Vielva, Y. Wiaux, R. B. Barreiro, L. Cayón, M. P. Hobson, A. N. Lasenby, E. Martinez-Gonzalez, and J. L. Sanz, *Cosmological applications of a wavelet analysis on the sphere*, Journal of Fourier Analysis and Applications, 13 (2007), pp. 495–510.

[21] Y. Rathi, O. Michailovich, F. Laun, K. Setsompop, P. E. Grant, and C.-F. Westin, *Multi-shell diffusion signal recovery from sparse measurements*, Medical image analysis, 18 (2014), pp. 1143–1156.

[22] J. Schmitt, J.-L. Starck, J.-M. Casandjian, J. Fadili, and I. Grenier, *Multichannel poisson denoising and deconvolution on the sphere: application to the fermi gamma-ray space telescope*, Astronomy & Astrophysics, 546 (2012), p. A114.
[23] F. J. Simons, I. Loris, G. Nolet, I. C. Daubechies, S. Voronin, J. Judd, P. A. Vetter, J. Charléty, and C. Vonesch, *Solving or resolving global tomographic models with spherical wavelets, and the scale and sparsity of seismic heterogeneity*, Geophysical Journal International, 187 (2011), pp. 969–988.
[24] H. S. Wong, H. Zhang, L. Li, Y. Fang, et al., *Incorporating the maximum entropy on the mean framework with kernel error for robust non-blind image deblurring*, Communications in Computational Physics, 31 (2022), pp. 893–912.
[25] T. Wu, X. Gu, Z. Li, Z. Li, J. Niu, and T. Zeng, *Efficient boosted DC algorithm for nonconvex image restoration with Rician noise*, SIAM Journal on Imaging Sciences, 15 (2022), pp. 424–454.
[26] T. Wu, W. Li, S. Jia, Y. Dong, and T. Zeng, *Deep multi-level wavelet-CNN denoiser prior for restoring blurred image with Cauchy noise*, IEEE Signal Processing Letters, 27 (2020), pp. 1635–1639.
[27] Y. Yoon, I. Chung, L. Wang, and K.-J. Yoon, *Spheresr: 360deg image super-resolution with arbitrary projection via continuous spherical image representation*, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 5677–5686.
[28] K. Zhang, Y. Li, W. Zuo, L. Zhang, L. Van Gool, and R. Timofte, *Plug-and-play image restoration with deep denoiser prior*, IEEE Transactions on Pattern Analysis and Machine Intelligence, (2021), pp. 1–17.
[29] K. Zhang, W. Zuo, S. Gu, and L. Zhang, *Learning deep cnn denoiser prior for image restoration*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 3929–3938.
[30] Y. Zhang, L. Zhang, W. Hamidouche, and O. Deforges, *A fixation-based 360 benchmark dataset for salient object detection*, in 2020 IEEE International Conference on Image Processing (ICIP), IEEE, 2020, pp. 3458–3462.
[31] X.-L. Zhao, W.-H. Xu, T.-X. Jiang, Y. Wang, and M. K. Ng, *Deep plug-and-play prior for low-rank tensor completion*, Neurocomputing, 400 (2020), pp. 137–149.