Probability-based Global Cross-modal Upsampling for Pansharpening

Zeyu Zhu[1], Xiangyong Cao[1,2,3*], Man Zhou[5], Junhao Huang[1], Deyu Meng[1,4]

[1]Xi'an Jiaotong University; [2]School of Computer Science and Technology, Xi'an Jiaotong University
[3]Ministry of Education Key Lab For Intelligent Networks and Network Security, Xi'an Jiaotong University
[4]Macau University of Science and Technology
[5]Nanyang Technological University

{zeyuzhu2077, manzhountu, junhaoxjtu}@gmail.com

{caoxiangyong, dymeng}@mail.xjtu.edu.cn

Abstract

Pansharpening is an essential preprocessing step for remote sensing image processing. Although deep learning (DL) approaches performed well on this task, current upsampling methods used in these approaches only utilize the local information of each pixel in the low-resolution multispectral (LRMS) image while neglecting to exploit its global information as well as the cross-modal information of the guiding panchromatic (PAN) image, which limits their performance improvement. To address this issue, this paper develops a novel probability-based global cross-modal upsampling (PGCU) method for pan-sharpening. Precisely, we first formulate the PGCU method from a probabilistic perspective and then design an efficient network module to implement it by fully utilizing the information mentioned above while simultaneously considering the channel specificity. The PGCU module consists of three blocks, i.e., information extraction (IE), distribution and expectation estimation (DEE), and fine adjustment (FA). Extensive experiments verify the superiority of the PGCU method compared with other popular upsampling methods. Additionally, experiments also show that the PGCU module can help improve the performance of existing SOTA deep learning pansharpening methods. The codes are available at https://github.com/Zeyu-Zhu/PGCU.
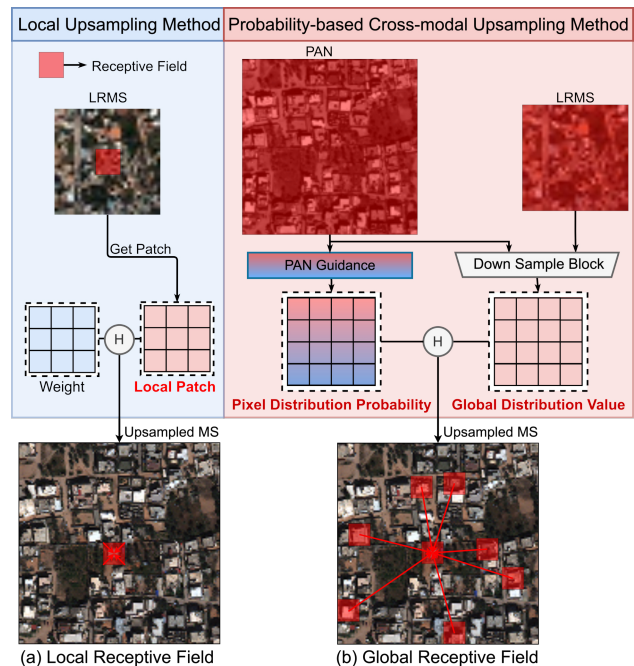
Figure 1. Comparison between local upsampling methods and our proposed PGCU method. The local method has limited receptive field and thus only utilizes the local information of LRMS for upsampling, while our proposed PGCU method can fully exploit the rich global information of LRMS and the cross-modal global information of PAN.

1. Introduction

Pansharpening aims to reconstruct a high-resolution multispectral image (HRMS) from a low-resolution multispectral image (LRMS) under the guidance of a panchromatic image (PAN). It's an indispensable pre-processing step for many subsequent remote sensing tasks, such as

object detection [11, 26], change detection [1, 19], unmixing [3] and classification [7, 8].

The last decades have witnessed the great development of pansharpening methods. The typical approaches include component substitution (CS) approaches [10, 18, 23, 24], multi-resolution analysis (MRA) methods [21, 25, 31], and variational optimization (VO) methods [12, 13, 15, 16, 38]. Recently, with the rapid development of deep learning,

---

*Corresponding author

plenty of deep learning-based methods [4,5,14,43,45] have been proposed to tackle this task due to its powerful non-linear fitting and feature extraction ability. Among these methods, almost all the approaches have a pipeline that upsamples the LRMS image first and then carries out other super-resolution operations. These approaches treat upsampling as an essential and indispensable component for this task. For instance, as for residual networks (e.g., PanNet), the upsampled image is directly added to the network's output, which makes the quality of the upsampled image an essential factor for model performance.

However, hardly any approaches explored to design a reasonable upsampling method for pansharpening but just simply utilized bicubic interpolation [9] and transposed convolution [17] as their upsampling module. At the same time, upsampling methods proposed for other tasks aren't suitable for pansharpening either, such as attention-based image upsampling (ABIU) [22] and ESPCNN [32]. Almost all the aforementioned upsampling methods are in the form of local interpolation and thus suffer from a limited receptive field issue. Therefore, these local interpolation-based upsampling methods fail to exploit similar patterns globally, while there are usually many non-local similar patches in remote sensing images, as shown in Figure 1(b). Additionally, almost all these upsampling methods are not capable of utilizing useful structure information from the PAN image. Also, some existing upsampling methods, e.g., ABIU [22] ignore channel specificity, which utilizes the same weight for the same position of all channels, which is unsuitable for pansharpening due to the significant difference among spectral image channels. In summary, these existing upsampling methods suffer from either insufficient utilization of information (i.e., global information of LRMS, structure information of PAN) or incomplete modeling of the problem (i.e., channel specificity issue).

To address the aforementioned problems, we propose a novel probability-based global cross-modal upsampling method (PGCU) to exploit cross-modal and global information while considering channel specificity. The reason why we utilize probabilistic modeling is that pansharpening is essentially an ill-posed image inverse problem. Probabilistic modeling can be used to better adapt to the characteristics of the problem itself. Specifically, an approximate global discrete distribution value is sampled from the pixel value space for each channel which can thus characterize the common property of each channel and the distinctive property of different channels. Then, we establish a cross-modal feature vector for each pixel in the upsampled HRMS image and discrete distribution value, using not only the LRMS image but also the PAN image. Inspired by the main idea of Transformer [36], we utilize vector similarity to calculate the probability value for each pixel on its channel distribution. Finally, PGCU calculates the pixel values

of the upsampled image by taking the expectation.

To implement the PGCU method, we design a network module containing three blocks, i.e., information extraction (IE) module block, distribution and expectation estimation (DEE) block, and fine adjustment (FA) block. Firstly, IE extracts spectral and spatial information from LRMS and PAN images to generate channel distribution value and cross-modal information. Next, DEE utilizes this information to construct cross-modal feature vectors for each pixel in the upsampled image and generate the distribution value, respectively. Then, they are used to estimate the distribution probability for each pixel in the upsampled image. Finally, FA further compensates for using the local information and channel correlation of the upsampled image.

To further explore the results obtained by PGCU, we utilize information theory to analyze pixel distribution. Specifically, by clustering pixels of the obtained upsampled image using JS divergence as the distance measurement, the spatial non-local correlation property of the image can be easily observed. Besides, by visualizing the information entropy image of each channel in the upsampled image, channel specificity can be easily observed as well, which also verifies that the PGCU method indeed learns the difference among channels.

To sum up, the contributions of this work are as follows:

- We propose a novel probability-based upsampling model for pan-sharpening. This model assumes each pixel of the upsampled image to obey a probability distribution given the LRMS image and PAN image.

- We design a new upsampling network module to implement the probability-based upsampling model. The module can fully exploit the global information of LRMS and the cross-modal information of PAN. As far as we know, PGCU is the first upsampling module specifically designed for pan-sharpening.

- Extensive experiments verify that the PGCU module can be embedded into the existing SOTA pansharpening networks to improve their performance in a plug-and-play manner. Also, the PGCU method is a universal upsampling method and has potential application in other guided image super-resolution tasks.

2. Related Work

2.1. Pansharpening Method

Model-based Approaches. The model-based pansharpening methods can be roughly divided into three categories, i.e., component substitution (CS) approaches, multiresolution analysis (MRA) methods, and variational optimization (VO) techniques. The main idea of the CS approach is to decompose the PAN image and LRMS image first and

then fuse the spatial information of the PAN image with the special information of the LRMS image to generate the HRMS image. Representative methods include principal component analysis (PCA) [23], Brovey method [18], intensity–hue-saturation (IHA) [10], and Gram-Schmidt (GS) method [24]. To further reduce spectral distortion, the MRA approaches reconstruct the HRMS image by injecting the structure information of the PAN image into the upsampled LRMS image. Typical methods include highpass filter (HPF) fusion [31], indusion method [21], smoothing filter-based intensity modulation (SFIM) [25] etc. The VO techniques reformulate the pansharpening as a variational optimization problem, such as Bayesian methods [38] and variational approaches [12, 13, 15, 16].

Deep Learning Approaches. In the last decade, deep learning(DL) methods have been studied for pansharpening, and this type of method directly learns the mapping from LRMS and PAN to HRMS. Typical DL-based pansharpening methods mainly contain two types of network architecture, i.e., residual and two-branch structure. The residual structure adds upsampled LRMS images to the network's output to obtain the HRMS in the form of regression residuals, such as PanNet [45], FusionNet [14], SRPPNN [4], etc [20, 34, 41, 49]. Recently, the two-branch structure is becoming more and more popular. This type of method usually conducts feature extraction for PAN and LRMS image, respectively, and fuses their features to reconstruct HRMS image, such as GPPNN [43], Proximal PanNet [5], SFIIN [50], etc [2, 6, 40, 44, 51, 52]. Both types of methods upsample LRMS first and then carry out other operations, implying that upsampling is a vital step for pan-sharpening.

2.2. Image Upsampling Method

Classical Methods. Many local interpolation-based upsampling methods are widely used in pansharpening tasks to obtain large-scale MS, especially the bicubic interpolation method [9]. Besides, there are plenty of similar techniques, such as nearest interpolation [35], bilinear interpolation [35], etc [27, 30]. However, this type of method suffers from seriously poor adaptability.

Deep Learning Methods. As deep learning blossoms, many learning-based upsampling methods have been proposed. For instance, transposed convolution [17] is widely used in many tasks to upsample low-resolution images, which can learn a self-adaptive weight for local interpolation. Following this work, an attention-based image upsampling method [22] is recently proposed for deep image super-resolution tasks by utilizing the transformer [36]. However, this method ignores the channel specificity since it uses the same weight for the same position of all channels, which is unsuitable for pansharpening due to the differences among spectral image channels. Additionally, there are also many other upsampling methods, such as Pu-Net [46], ES-

PCNN [32], etc [28, 29, 39]. Among them, ESPCNN is proposed for single-image super-resolution, which enlarges the receptive field by multi-convolution layers.

However, these upsampling methods suffer from three issues. Firstly, almost all these methods only have a local receptive field, which are unable to explore the global information of LRMS. Secondly, most of the upsampling methods can't utilize PAN information as guidance. Thirdly, channel specificity is not considered in these methods.

3. Proposed Upsampling Method

In this section, we first introduce our proposed probability-based global cross-modal upsampling (PGCU) method. Then, we design a network architecture to implement the PGCU method.

3.1. Probabilistic Modeling

Before presenting our upsampling method, we first define some necessary notations. As aforementioned, the pansharpening task aims to obtain an HRMS image from the LRMS image under the guidance of the PAN image. In our method, the upsampled image is denoted as $H \in \mathbb{R}^{C \times W \times H}$, the LRMS image is represented as $L \in \mathbb{R}^{C \times w \times h}$, and the PAN image is defined as $P \in \mathbb{R}^{W \times H}$. Additionally, we denote each pixel of the upsampled image as $h_{c,i,j} \in \mathbb{R}, c = 1, \ldots, C, i = 1, \ldots, W, j = 1, \ldots, H$. Next, we will directly model the pixel $h_{c,i,j}$ from a probabilistic perspective and propose a new upsampling method for the pansharpening task.

Generally, in our proposed upsampling method, we treat each pixel $h_{c,i,j}$ as a random variable and then aim to model its probability distribution by utilizing information from the LRMS image L and the PAN image P. More precisely, PGCU uses the expectation of a discrete distribution to approximate the one of continuous distribution. For the sake of simplicity, we don't put subscript here and assume a pixel in the HRMS image h obeys a continuous distribution which has support over the interval $[0, 1]$ and $p(\cdot)$ is its probability density function. Thus, the expectation of h is

$$\mathbb{E}(h) = \int_0^1 hp(h)\mathrm{d}h \approx \sum_{i=0}^{i=k} h_i p(h_i)\delta h_i, \qquad (1)$$

where h_i is the sample drawn from $[0, 1]$, k is sample size, and $\sum_i p(h_i)\delta h_i = 1$. Here we use the sampling method to approximate the integral numerically. Besides, there must exist a discrete distribution $\mathcal{D}(\cdot)$ satisfying condition,

$$\mathcal{D}(h_i) = p(h_i)\delta h_i = w_i, i = 1, 2, ..., k. \qquad (2)$$

w_i can thus represent the importance of the sample h_i. Then, the expectation of continuous variable h can be approximated by the expectation of discrete distribution $\mathcal{D}(\cdot)$.

So thus, we assume that $h_{c,i,j}$ obeys a discrete distribution

$$h_{c,i,j} \sim \mathcal{D}(h_{c,i,j}|\boldsymbol{v^c}, \boldsymbol{p^{c,i,j}}), \qquad (3)$$

where $\mathcal{D}(h_{c,i,j}|\boldsymbol{v^c}, \boldsymbol{p^{c,i,j}})$ is a discrete distribution with variable value $\boldsymbol{v^c} \in \mathbb{R}^n$ and probability vector parameter $\boldsymbol{p^{c,i,j}} \in \mathbb{R}^n$, i.e., samples and sample importance. Further, by considering the fact that the pixel value $h_{c,i,j}$ of the up-sampled image is dependent on the LRMS image L and the PAN image P, we hypothesize that both $\boldsymbol{v^c}$ and $\boldsymbol{p^{c,i,j}}$ are a function of L and P. Once $\boldsymbol{v^c}$ and $\boldsymbol{p^{c,i,j}}$ are known, the distribution $\mathcal{D}(\cdot)$ can be explicitly written as

$$P(h_{c,i,j} = v_k^c|L, P) = p_k^{c,i,j}, k = 1, 2, \ldots, n \qquad (4)$$

Additionally, in the definition of the discrete distribution $\mathcal{D}(\cdot)$, it should be noted that all the pixels in the c_{th} channel share a common distribution value vector $\boldsymbol{v^c}$, and different channels have different $\boldsymbol{v^c}$, which can thus characterize the common property of each channel and the distinctive property of different channels.

As aforementioned, the distribution parameters (i.e., $\boldsymbol{v^c}$ and $\boldsymbol{p^{c,i,j}}$) are defined as the function of L and P. In general, we adopt three functions, i.e., $V_{\theta_v}(\cdot)$ and $G_{\theta_g}(\cdot)$, $F_{\theta_f}(\cdot)$, to generate $\boldsymbol{v^c}$ and $\boldsymbol{p^{c,i,j}}$. Specifically, the generation process of $\boldsymbol{v^c}$ is

$$\boldsymbol{v^c} = V_{\theta_v}(L, P), \qquad (5)$$

where $V_{\theta_v}(L, P)$ is implemented by utilizing the structure information of P and the spectral information of L to generate a high expressive distribution value v^c, θ_v is parameter of V_{θ_v}, and each channel has its own $\boldsymbol{v^c}$. As for $\boldsymbol{p^{c,i,j}}$, we first generate two feature vectors as follows:

$$f_{c,i,j} = F_{\theta_f}(L, P), \qquad (6)$$
$$g_{c,k} = G_{\theta_g}(L, P), \ k = 1, \ldots, n \qquad (7)$$

where $F_{\theta_f}(L, P)$ aims to extract cross-modal information in the local patch for each pixel, $f_{c,i,j}$ is thus a feature vector which captures the cross-modal information of the corresponding pixel, $G_{\theta_g}(L, P)$ is also implemented by using the cross-modal information in local patch to capture the property of distribution value feature $\boldsymbol{v^c}$, $g_{c,k}$ is thus another feature vector which characterizes the information of the probability density function near the corresponding distribution value feature, θ_f is the parameter of F_{θ_f}, and θ_g is the parameter of G_{θ_g}. Further, by computing the similarity of the two vectors, we can obtain $\boldsymbol{p^{c,i,j}}$ as follows:

$$\tilde{p}_k^{c,i,j} = \frac{<f_{c,i,j}, g_{c,k}>}{||\vec{f}_{c,i,j}||_2 ||g_{c,k}||_2}, \ k = 1, \ldots, n \qquad (8)$$
$$\boldsymbol{p^{c,i,j}} = Softmax(\tilde{\boldsymbol{p}}^{c,i,j}), \qquad (9)$$

where $< \cdot, \cdot >$ is the inner product operator, $|| \cdot ||_2$ is the ℓ_2 norm, and *Softmax* is a normalization function, which transforms $\tilde{p}^{c,i,j}$ to be a probability (i.e., the sum is 1). So far,
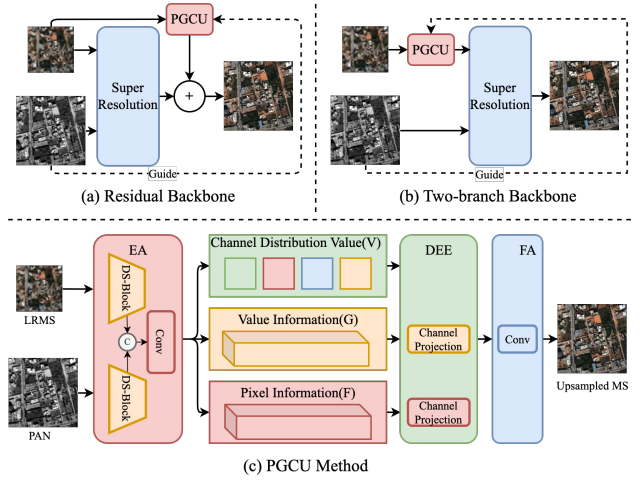

Figure 2. How our proposed PGCU module can be used in the existing pansharpening networks. (a) PGCU module is embedded into the residual backbone; (b) PGCU module is embedded into the two-branch backbone; (c) The overall flow of PGCU module.

we have defined the generation process of $\boldsymbol{v^c}$ and $\boldsymbol{p^{c,i,j}}$, and thus we can obtain the distribution of each pixel $h_{c,i,j}$, i.e., $\mathcal{D}(h_{c,i,j}|\boldsymbol{v^c}(\theta_v), \boldsymbol{p^{c,i,j}}(\theta_f, \theta_g))$. Now it should be noted that the distribution $\mathcal{D}(\cdot)$ is parameterized by θ_v, θ_f, and θ_g. Once these parameters are learned, we can easily obtain the upsampled image $\tilde{H} = (\tilde{h}_{c,i,j})_{c,i,j}$ by taking the expectation, namely,

$$\tilde{h}_{c,i,j} = \mathbb{E}(h_{c,i,j}), \qquad (10)$$

where $\mathbb{E}(\cdot)$ is the expectation operator.

In summary, the above process defines a new upsampling method for pansharpening called PGCU. Next, we will design an efficient network to implement the PGCU method.

3.2. Network Architecture

Figure 2 (c) illustrates the overall network architecture of the PGCU module, which consists of three blocks, i.e., information extraction (IE) block, distribution and expectation estimation (DEE) block, and fine adjustment (FA) block. The detailed structure of the three blocks is shown in Figure 3. Additionally, the usage of the PGCU module is also presented in Figure 2 (a) and (b), from which we can see that the PGCU module can be easily embedded into current pansharpening networks.

3.2.1 Information Extraction

The information extraction (IE) block receives PAN image P and LRMS image L as input and outputs variable value $\boldsymbol{v^c}$ of the discrete distribution $\mathcal{D}(h_{c,i,j}|\boldsymbol{v^c}, \boldsymbol{p_{c,i,j}})$ of pixel $h_{c,i,j}$ in the upsampled image H and the cross-modal features for subsequent feature vector construction. To exploit
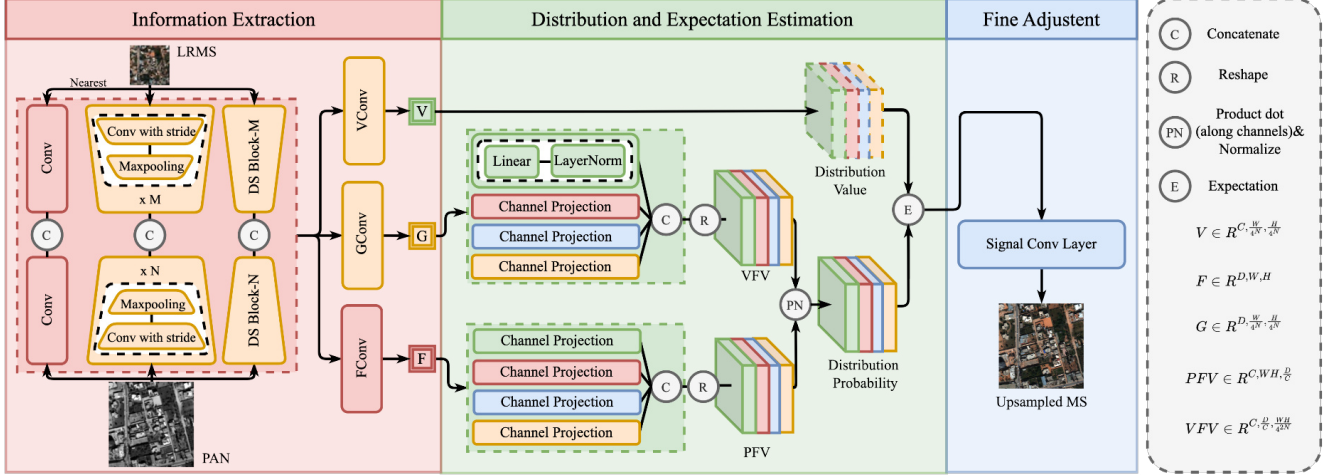
Figure 3. The detailed structure of each block in the PGCU module, where D represents the vector dimension in F and G.

information from the LMRS image and PAN image simultaneously, we first perform feature extraction on both of them. This process can be modeled by two functions (i.e., $V_{\theta_v}(\cdot)$ and $G_{\theta_g}(\cdot)$) as aforementioned. Here, we design two blocks to implement them, which is defined as

$$V = \mathrm{Conv}\{\mathrm{Cat}[\mathrm{DS}_N(P), \mathrm{DS}_M(L)]\}, \quad (11)$$

$$G = \mathrm{Conv}\{\mathrm{Cat}[\mathrm{DS}_N(P), \mathrm{DS}_M(L)]\}, \quad (12)$$

where $V = \{v^c\}_{c=1}^C$, $G = \{g_{c,k}\}_{c=1,k=1}^{C,n}$, $\mathrm{Conv}(\cdot)$ is convolutional operator, $\mathrm{Cat}(\cdot)$ is concatenate operator, $\mathrm{DS}_N(\cdot)$ is the downsampling block for PAN image and $\mathrm{DS}_M(\cdot)$ is the downsampling block for LRMS image. The downsampling (DS) block consists of one convolutional layer with stride s and one (2,2) max pooling layer.

Additionally, each pixel in the LRMS image is actually can be regarded as a degradation from the nearest corresponding points in HRMS. Considering this fact, we first simply use the nearest upsampling method to construct a large-scale MS tensor before extracting features for each pixel. As aforementioned, this process can be defined by $F_{\theta_f}(\cdot)$, and this function is explicitly implemented by the designed block as follows

$$F = \mathrm{Conv}\{\mathrm{Cat}[\mathrm{Conv}(P), \mathrm{Conv}(\mathrm{Nearest}(L))]\}, \quad (13)$$

where $F = \{f_{c,i,j}\}_{c=1,i=1,j=1}^{C,W,H}$ and $\mathrm{Nearest}(\cdot)$ is the nearest upsampling method.

3.2.2 Distribution and Expectation Estimation

The distribution and expectation estimation (DEE) block estimates a distribution probability for each pixel and computes the expected value to get the estimated value of the pixel in the upsampled image. In this DEE block, feature F and G obtained from the previous IE block are input to the

channel projection block, which is used to model the channel specificity. Each channel projection block consists of a linear layer and a LayerNorm. The Linear layer is used to map the input features (i.e., F and G) to a new feature space of the corresponding channel. The following LayerNorm is utilized to highlight the differences among each individual feature vector. The channel projection for each channel is formulated as

$$PFV = Cat\{\mathrm{LayerNorm}[\mathrm{Linear}_i(F)]_i\}_{i=1}^C, \quad (14)$$

$$VFV = Cat\{\mathrm{LayerNorm}[\mathrm{Linear}_i(G)]_i\}_{i=1}^C, \quad (15)$$

where F consists of feature vectors for each position of the upsampled image (without channel specificity) and G consists of feature vectors for each position of distribution value (without channel specificity). Then, the similarity matrix is calculated by Eq. (8) and further normalized by Eq. (9) to obtain the distribution probability. Finally, the expected pixel value is computed by taking the expectation.

3.2.3 Fine Adjustment

The fine adjustment (FA) module is implemented by a single convolutional layer, which is beneficial to better utilize the local information and the dependence among channels to compensate for the global feature and channel specificity.

Table 1. The basic information for each dataset.

Datasets	WordView2	WordView3	GaoFen2
Train/Test	768/80	2160/208	2720/208
PAN	128×128	128×128	128×128
LRMS	32×32×4	32×32×4	32×32×4
HRMS	128×128×4	128×128×4	128×128×4

Table 2. The average results of component replacement experiments. Methods with * represent the method whose upsampling method is replaced by our PGCU method without any further changes. The best results in each column are in bold.

Method	WorldView2					WorldView3					GaoFen2				
	SAM↓	EGRAS↓	SSIM↑	SCC↑	PSNR↑	SAM↓	EGRAS↓	SSIM↑	SCC↑	PSNR↑	SAM↓	EGRAS↓	SSIM↑	SCC↑	PSNR↑
PanNet [45]	0.037	1.504	0.925	0.939	37.459	0.106	4.101	0.871	0.930	28.212	0.019	0.912	0.962	0.927	42.619
PanNet*	**0.023**	**0.952**	**0.970**	**0.976**	**41.659**	**0.077**	**3.174**	**0.919**	**0.959**	**30.319**	**0.011**	**0.573**	**0.986**	**0.958**	**46.715**
MSDCNN [47]	0.028	1.109	0.960	0.967	40.344	0.080	3.254	0.916	0.956	30.076	0.018	0.837	0.968	0.940	43.254
MSDCNN*	**0.026**	**1.078**	**0.964**	**0.968**	**40.631**	**0.078**	**3.183**	**0.920**	**0.958**	**30.283**	**0.015**	**0.720**	**0.978**	**0.946**	**44.711**
FusionNet [14]	0.028	1.131	0.957	0.963	40.081	0.089	3.4834	0.901	0.947	29.541	0.018	0.877	0.966	0.932	42.974
FusionNet*	**0.024**	**0.994**	**0.967**	**0.971**	**41.255**	**0.077**	**3.203**	**0.919**	**0.958**	**30.261**	**0.013**	**0.636**	**0.983**	**0.955**	**45.839**
GPPNN [43]	0.025	1.006	0.968	0.972	41.190	0.081	3.305	0.916	0.955	29.979	0.012	0.595	0.986	0.954	46.566
GPPNN*	**0.022**	**0.942**	**0.970**	**0.975**	**41.659**	**0.075**	**3.174**	**0.920**	**0.959**	**30.349**	**0.010**	**0.519**	**0.989**	**0.963**	**47.815**
SFIIN [50]	0.024	1.007	0.967	0.971	41.115	0.079	3.239	0.917	0.956	30.143	0.012	0.628	0.986	0.947	46.199
SFIIN*	**0.023**	**0.950**	**0.970**	**0.975**	**41.617**	**0.077**	**3.145**	**0.922**	**0.960**	**30.399**	**0.010**	**0.495**	**0.989**	**0.964**	**48.156**

4. Experiments

In this section, we conduct several experiments to verify the effectiveness of our proposed PGCU method. Specifically, we first select five representative DL-based pansharpening approaches, including PanNet [45], MSDCNN [47], FusionNet [14], GPPNN [43] and SFIIN [50] as backbones and replace the upsampling method in these approaches with our PGCU method. Among these approaches, PanNet adopts transposed convolution upsampling method and the other four use bicubic interpolation for upsampling. Besides, to further prove that the improvement isn't brought from the increase of parameter quantity, we carry out an equal parameter experiment. Secondly, we compare our proposed PGCU method with five popular upsampling methods, including traditional bicubic interpolation [35], nearest interpolation [35], and the latest DL-based transposed convolution (TConv) [17], attention-based image upsampling (ABIU) [22], and ESPCNN [32]. Thirdly, we conduct an ablation study on the main factors of our method. Finally, we provide a visualization analysis of the distribution of pixels in the learned upsampled image. The hyperparameters s, N, M, and L of PGCU are set as 2, 3, 2, and 128, respectively. All the experiments are conducted on a PC with Intel Core i7-8700K CPU and one GeForce RTX 3090 Ti with 24GB memory.

4.1. Datasets and Evaluation Metrics

Three datasets are used in our experiments, which are generated from three different satellites, i.e., WordView2, WordView3, and GaoFen2. Each dataset is divided into training and testing sets. The basic information for each dataset is shown in Table 1. In all datasets, we generate LRMS images via downsampling HRMS by a scale of four using bicubic interpolation. And every pixel is normalized to $[0, 1]$ for numerical stability. Five popular metrics are chosen to evaluate the performance of each method [37], including spectral angle mapper (SAM), the relative dimen-

sionless global error in synthesis (ERGAS), the structural similarity (SSIM), the spatial correlation coefficient (SCC), and the peak signal-to-noise ratio (PSNR).

4.2. Component Replacement Experiment

To verify the effectiveness of our PGCU method, we first test our method in a plug-and-play way by directly replacing the original upsampling methods in five SOTA DL-based pansharpening methods with our PGCU method. Each pair of approaches (e.g., PanNet and PanNet*) is experimented under the same conditions. The experiment results are shown in Table 2. It can be easily observed that all five backbones have a significant performance improvement on all the datasets after replacing their upsampling methods with our PGCU method. A visual image comparison is shown in Figure 4, from which we can draw the same conclusion as Table 2.

Further, to prove the fact that the performance improvement doesn't come from the increase of model complexity (i.e., the increase in the number of parameters) but from the reasonable design of our PGCU method, we increase the parameter quantity of two backbones (i.e., PanNet and GPPNN) to the same as after component replacement. Specifically, we increase the number of ResNet blocks for PanNet and Pan-Ms blocks for GPPNN to make the parameter quantity of PanNet and GPPNN slightly greater than or equal to PanNet* and GPPNN*'s, respectively. The experimental results are illustrated in Table 3. Slight performance improvement can be seen after increasing the number of parameters in PanNet and GPPNN. However, there's still a large performance gap compared with PanNet* and GPPNN*, which implies that the performance improvement attributes to our PGCU method.

4.3. Comparison with Other Upsampling Methods

To further illustrate the superiority of our PGCU method, we compare our method with five popular above-mentioned upsampling methods. Similar to the previous experiment,

Table 3. The results of equal parameter experiment. The parameter quantity of methods with ⋄ is incremented to be the same as methods with *. The best results in each column are in bold.

Method	Param	WorldView2					WorldView3					GaoFen2				
		SAM↓	EGRAS↓	SSIM↑	SCC↑	PSNR↑	SAM↓	EGRAS↓	SSIM↑	SCC↑	PSNR↑	SAM↓	EGRAS↓	SSIM↑	SCC↑	PSNR↑
PanNet	0.10M	0.037	1.504	0.925	0.939	37.459	0.106	4.101	0.871	0.930	28.212	0.019	0.912	0.962	0.927	42.619
PanNet⋄	0.15M	0.035	1.487	0.927	0.941	37.553	0.104	4.035	0.872	0.931	28.349	0.019	0.910	0.962	0.927	42.630
PanNet*	0.15M	**0.023**	**0.952**	**0.970**	**0.976**	**41.659**	**0.077**	**3.174**	**0.919**	**0.959**	**30.319**	**0.011**	**0.573**	**0.986**	**0.958**	**46.715**
GPPNN	0.12M	0.025	1.006	0.968	0.972	41.190	0.081	3.305	0.916	0.955	29.979	0.012	0.595	0.986	0.954	46.566
GPPNN⋄	0.17M	0.024	0.994	0.968	0.973	41.318	0.0823	3.307	0.915	0.954	29.934	0.012	0.568	0.987	0.955	46.962
GPPNN*	0.17M	**0.022**	**0.942**	**0.970**	**0.975**	**41.659**	**0.075**	**3.174**	**0.920**	**0.959**	**30.349**	**0.010**	**0.519**	**0.989**	**0.963**	**47.815**

(a) PanNet(*37.111*)   (b) MSDCNN(*42.468*)   (c) FusionNet(*41.938*)   (d) GPPNN(*42.794*)   (e) SFIIN(*42.598*)

GroundTruth

(f) PanNet*(43.157)   (g) MSDCNN*(42.980)   (h) FusionNet*(*43.219*)   (i) GPPNN*(43.395)   (j) SFIIN*(42.*659*)
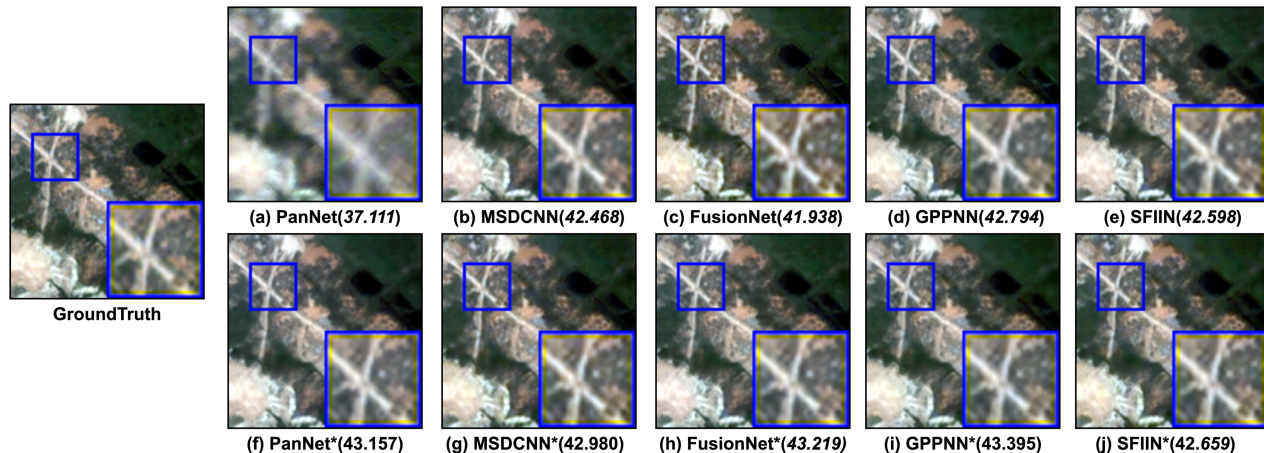
Figure 4. Visualize comparison of one sample image from the WV2 dataset. PSNR value is shown under the corresponding method.

the backbone networks are PanNet and GPPNN, and the used datasets are WorldView2 and GaoFen2.

The experimental results are recorded in Table 4. As can be seen, the backbone network with our proposed PGCU method can obtain the best performance. Specifically, all the competing methods aren't capable of exploiting the global information of LRMS. Besides, the first four methods also ignore the cross-modal information from the PAN image. As for the ABIU method, although it can utilize cross-modal information, its sampling weights are consistent for all channels at the same location, which ignores the difference among channels. Compared with these methods, our proposed PGCU can not only make full use of cross-modal and global information but also adequately model the channel specificity, which is why our method performs best.

4.4. Parameter Analysis and Ablation Study

The length of the feature vector D for distribution value and pixel in the upsampled image is a very important hyperparameter for our proposed PGCU, which determines the representational ability for each pixel. The parameter analysis experiment is reported in Table 5. As can be seen, a short vector will lead to the inability to represent pixel information, and a long vector will result in redundant infor-

mation.

Further, we conduct an ablation experiment to investigate the function of different information sources or operators. The experiment is conducted using PanNet as a backbone on the GaoFen2 dataset, and the results are shown in Table 6. Specifically, to exploit the importance of PAN image information, the feature extraction only performs on LRMS images. Then, the performance has an apparent decline, implying that utilizing cross-modal information is crucial. Additionally, the channel projection module is removed from our method to study the importance of modeling the channel specificity. We can observe that the performance also has an apparent drop, which verifies the necessity of modeling the channel specificity.

4.5. Visualization Analysis

To further explore the results obtained by PGCU, we analyze the distribution of pixels via information theory and discover some interesting phenomena. The visualization results are shown in Figure 5. The first row shows each channel of the HRMS image. The second row shows the clustering of pixels of different channels using the distribution of pixels and the Kmeans with JS divergence as the distance metric. Pixels in the same class are stained the same

Table 4. The comparison results of our PGCU method with other upsampling methods. The best and second-best results are highlighted in bold and underlined, respectively.

Backbone	Upsampling Method	WordView2					GaoFen2				
		SAM↓	EGRAS↓	SSIM↑	SCC↑	PSNR↑	SAM↓	EGRAS↓	SSIM↑	SCC↑	PSNR↑
PanNet	**Nearest** [35]	0.026	1.085	0.959	0.967	40.255	0.019	0.942	0.961	0.9238	42.359
	Bicubic [35]	0.025	1.066	0.960	0.969	40.369	0.018	0.887	0.964	0.934	42.823
	TConv [17]	**0.024**	<u>1.014</u>	<u>0.964</u>	<u>0.972</u>	<u>40.833</u>	0.019	0.839	0.969	0.940	43.076
	ESPCNN [32]	0.025	1.053	0.961	0.969	40.511	<u>0.015</u>	<u>0.718</u>	<u>0.977</u>	<u>0.951</u>	<u>44.691</u>
	ABIU [22]	0.026	1.069	0.962	0.967	40.492	-	-	-	-	-
	PGCU	**0.024**	**0.953**	**0.970**	**0.976**	**41.659**	**0.012**	**0.573**	**0.986**	**0.958**	**46.715**
GPPNN	**Nearest** [35]	0.026	1.016	0.961	0.969	41.112	0.013	0.599	0.982	0.951	46.532
	Biubic [35]	0.025	1.006	0.968	0.972	41.190	0.012	0.595	0.986	0.954	46.566
	TConv [17]	0.025	1.002	0.967	0.971	40.993	<u>0.011</u>	0.564	<u>0.987</u>	0.954	47.116
	ESPCNN [32]	<u>0.024</u>	<u>0.980</u>	<u>0.969</u>	<u>0.973</u>	<u>41.413</u>	<u>0.011</u>	0.556	<u>0.987</u>	0.956	47.282
	ABIU [22]	-	-	-	-	-	-	-	-	-	-
	PGCU	**0.022**	**0.942**	**0.970**	**0.975**	**41.659**	**0.010**	**0.519**	**0.989**	**0.963**	**47.815**

Table 5. Experimental results with different feature vector lengths.

Feature Vector Length	GaoFen2				
	SAM↓	EGRAS↓	SSIM↑	SCC↑	PSNR↑
32	0.017	0.780	0.974	0.948	43.722
64	0.014	0.670	0.981	0.954	45.334
96	<u>0.013</u>	0.621	<u>0.984</u>	0.955	46.166
128	**0.012**	**0.573**	**0.986**	**0.958**	**46.715**
160	<u>0.013</u>	<u>0.601</u>	<u>0.984</u>	<u>0.957</u>	<u>46.436</u>
192	0.014	0.622	0.983	0.956	46.003

Table 6. Ablation study on PAN information and channel specificity.

PAN Information	Channel Projection	GaoFen2				
		SAM↓	EGRAS↓	SSIM↑	SCC↑	PSNR↑
✗	✗	0.014	0.684	0.980	0.951	45.107
✗	✔	<u>0.013</u>	<u>0.612</u>	0.984	<u>0.956</u>	<u>46.181</u>
✔	✗	0.013	0.648	<u>0.985</u>	<u>0.956</u>	46.056
✔	✔	**0.012**	**0.573**	**0.986**	**0.958**	**46.715**

Figure 5. Visualization analysis on the obtained HRMS.

color, and many non-local patches in the same color can be found. The third row shows each channel's normalized information entropy of pixels. The vast difference among information entropy maps of different channels shows that the uncertainty of pixels with the same location in different channels is diverse. And our PGCU method can adaptively take full advantage of information from each channel.

5. Conclusion and Future Work

In this paper, we first propose a novel upsampling method for pansharpening from a probabilistic perspective by introducing global and PAN information into the upsampling process while fully modeling channel specificity. Then we design a network to implement this method, and th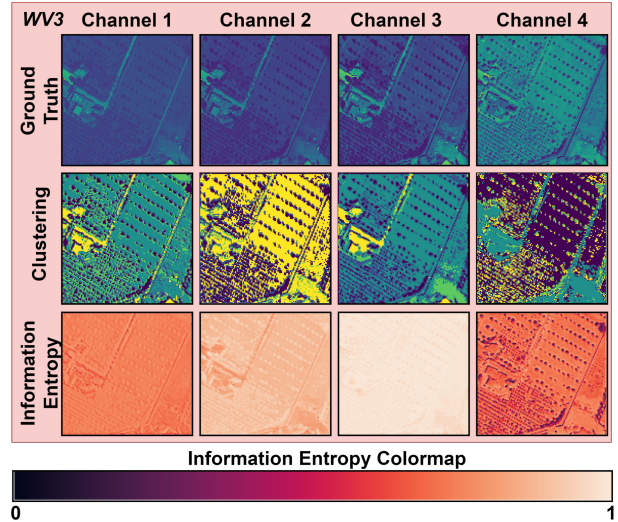is module can help improve the performance of current SOTA methods in a plug-and-play manner. In the future, we will apply our upsampling module to more guided image super-resolution tasks, i.e., depth image super-resolution [48], MRI super-resolution [33], and etc. [42].

References

[1] Anju Asokan and JJESI Anitha. Change detection techniques for remote sensing applications: a survey. *Earth Science Informatics*, 12(2):143–160, 2019. 1

[2] Wele Gedara Chaminda Bandara and Vishal M Patel. Hypertransformer: A textural and spectral feature fusion transformer for pansharpening. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1767–1777, 2022. 3

[3] José M Bioucas-Dias, Antonio Plaza, Nicolas Dobigeon, Mario Parente, Qian Du, Paul Gader, and Jocelyn Chanussot. Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches. *IEEE journal of selected topics in applied earth observations and remote sensing*, 5(2):354–379, 2012. 1

[4] Jiajun Cai and Bo Huang. Super-resolution-guided progressive pansharpening based on a deep convolutional neural network. *IEEE Transactions on Geoscience and Remote Sensing*, 59(6):5206–5220, 2020. 2, 3

[5] Xiangyong Cao, Yang Chen, and Wenfei Cao. Proximal pannet: A model-based deep network for pansharpening. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 176–184, 2022. 2, 3

[6] Xiangyong Cao, Xueyang Fu, Danfeng Hong, Zongben Xu, and Deyu Meng. Pancsc-net: A model-driven deep unfolding method for pansharpening. *IEEE Transactions on Geoscience and Remote Sensing*, DOI: 10.1109/TGRS.2021.3115501, 2021. 3

[7] Xiangyong Cao, Jing Yao, Zongben Xu, and Deyu Meng. Hyperspectral image classification with convolutional neural network and active learning. *IEEE Transactions on Geoscience and Remote Sensing*, 58(7):4604–4616, 2020. 1

[8] Xiangyong Cao, Feng Zhou, Lin Xu, Deyu Meng, Zongben Xu, and John Paisley. Hyperspectral image classification with markov random fields and a convolutional neural network. *IEEE Transactions on Image Processing*, 27(5):2354–2367, 2018. 1

[9] Ralph E Carlson and Frederick N Fritsch. Monotone piecewise bicubic interpolation. *SIAM Journal on Numerical Analysis*, 22(2):386–400, 1985. 2, 3

[10] Wjoseph Carper, Thomasm Lillesand, and Ralphw Kiefer. The use of intensity-hue-saturation transformations for merging spot panchromatic and multispectral image data. *Photogrammetric Engineering and Remote Sensing*, 56(4):459–467, 1990. 1, 3

[11] Gong Cheng and Junwei Han. A survey on object detection in optical remote sensing images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 117:11–28, 2016. 1

[12] Liang-Jian Deng, Minyu Feng, and Xue-Cheng Tai. The fusion of panchromatic and multispectral remote sensing images via tensor-based sparse modeling and hyper-laplacian prior. *Information Fusion*, 52:76–89, 2019. 1, 3

[13] Liang-Jian Deng, Gemine Vivone, Weihong Guo, Mauro Dalla Mura, and Jocelyn Chanussot. A variational pansharpening approach based on reproducible kernel hilbert space and heaviside function. *IEEE Transactions on Image Processing*, 27(9):4330–4344, 2018. 1, 3

[14] Liang-Jian Deng, Gemine Vivone, Cheng Jin, and Jocelyn Chanussot. Detail injection-based deep convolutional neural networks for pansharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 59(8):6995–7010, 2020. 2, 3, 6

[15] Faming Fang, Fang Li, Chaomin Shen, and Guixu Zhang. A variational approach for pan-sharpening. *IEEE Transactions on Image Processing*, 22(7):2822–2834, 2013. 1, 3

[16] Xueyang Fu, Zihuang Lin, Yue Huang, and Xinghao Ding. A variational pan-sharpening with local gradient constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10265–10274, 2019. 1, 3

[17] Hongyang Gao, Hao Yuan, Zhengyang Wang, and Shuiwang Ji. Pixel transposed convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(5):1218–1227, 2019. 2, 3, 6, 8

[18] Alan R Gillespie, Anne B Kahle, and Richard E Walker. Color enhancement of highly correlated images. ii. channel ratio and "chromaticity" transformation techniques. *Remote Sensing of Environment*, 22(3):343–365, 1987. 1, 3

[19] Gong Jianya, Sui Haigang, Ma Guorui, and Zhou Qiming. A review of multi-temporal remote sensing data change detection algorithms. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37(B7):757–762, 2008. 1

[20] Zi-Rong Jin, Tian-Jing Zhang, Tai-Xiang Jiang, Gemine Vivone, and Liang-Jian Deng. Lagconv: Local-context adaptive convolution kernels with global harmonic bias for pansharpening. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1113–1121, 2022. 3

[21] Muhammad Murtaza Khan, Jocelyn Chanussot, Laurent Condat, and Annick Montanvert. Indusion: Fusion of multispectral and panchromatic images using the induction scaling technique. *IEEE Geoscience and Remote Sensing Letters*, 5(1):98–102, 2008. 1, 3

[22] Souvik Kundu, Hesham Mostafa, Sharath Nittur Sridhar, and Sairam Sundaresan. Attention-based image upsampling. *arXiv preprint arXiv:2012.09904*, 2020. 2, 3, 6, 8

[23] P Kwarteng and A Chavez. Extracting spectral contrast in landsat thematic mapper image data using selective principal component analysis. *Photogrammetry Engineering Remote Sensing*, 55(1):339–348, 1989. 1, 3

[24] Craig A Laben and Bernard V Brower. Process for enhancing the spatial resolution of multispectral imagery using pansharpening, Jan. 4 2000. US Patent 6,011,875. 1, 3

[25] JG Liu. Smoothing filter-based intensity modulation: A spectral preserve image fusion technique for improving spatial details. *International Journal of Remote Sensing*, 21(18):3461–3472, 2000. 1, 3

[26] Junmin Liu, Shijie Li, Changsheng Zhou, Xiangyong Cao, Yong Gao, and Bo Wang. Sraf-net: A scene-relevant anchor-free object detection network in remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, DOI: 10.1109/TGRS.2021.3124959, 2021. 1

[27] Ming-Yu Liu, Oncel Tuzel, and Yuichi Taguchi. Joint geodesic upsampling of depth images. In *Proceedings of the*

IEEE Conference on Computer Vision and Pattern Recognition, pages 169–176, 2013. 3

[28] Davide Mazzini. Guided upsampling network for real-time semantic segmentation. *arXiv preprint arXiv:1807.07466*, 2018. 3

[29] Sachit Menon, Alexandru Damian, Shijia Hu, Nikhil Ravi, and Cynthia Rudin. Pulse: Self-supervised photo upsampling via latent space exploration of generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2437–2445, 2020. 3

[30] Jaesik Park, Hyeongwoo Kim, Yu-Wing Tai, Michael S Brown, and Inso Kweon. High quality depth map upsampling for 3d-tof cameras. In *2011 International Conference on Computer Vision*, pages 1623–1630. IEEE, 2011. 3

[31] Robert A Schowengerdt. Reconstruction of multispatial, multispectral image data using spatial frequency content. *Photogrammetric Engineering and Remote Sensing*, 46(10):1325–1334, 1980. 1, 3

[32] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1874–1883, 2016. 2, 3, 6, 8

[33] Liyao Song, Quan Wang, Ting Liu, Haiwei Li, Jiancun Fan, Jian Yang, and Bingliang Hu. Deep robust residual network for super-resolution of 2d fetal brain mri. *Scientific Reports*, 12(1):1–8, 2022. 8

[34] Yi Sun, Yuanlin Zhang, and Yuan Yuan. Adaptive detail injection-based feature pyramid network for pan-sharpening. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 1646–1650. IEEE, 2022. 3

[35] Richard Szeliski. *Computer vision: algorithms and applications*. Springer Nature, 2022. 3, 6, 8

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017. 2, 3

[37] Gemine Vivone, Luciano Alparone, Jocelyn Chanussot, Mauro Dalla Mura, Andrea Garzelli, Giorgio A Licciardi, Rocco Restaino, and Lucien Wald. A critical comparison among pansharpening algorithms. *IEEE Transactions on Geoscience and Remote Sensing*, 53(5):2565–2586, 2014. 6

[38] Tingting Wang, Faming Fang, Fang Li, and Guixu Zhang. High-quality bayesian pansharpening. *IEEE Transactions on Image Processing*, 28(1):227–239, 2018. 1, 3

[39] Zhihui Wang, Xinchen Ye, Baoli Sun, Jingyu Yang, Rui Xu, and Haojie Li. Depth upsampling based on deep edge-aware learning. *Pattern Recognition*, 103:107274, 2020. 3

[40] Xiao Wu, Ting-Zhu Huang, Liang-Jian Deng, and Tian-Jing Zhang. Dynamic cross feature fusion for remote sensing pansharpening. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14687–14696, 2021. 3

[41] Jin-Liang Xiao, Ting-Zhu Huang, Liang-Jian Deng, Zhong-Cheng Wu, and Gemine Vivone. Variational pansharpening based on coefficient estimation with nonlocal regression. *Available at SSRN 4243668*. 3

[42] Qi Xie, Minghao Zhou, Qian Zhao, Deyu Meng, Wangmeng Zuo, and Zongben Xu. Multispectral and hyperspectral image fusion by ms/hs fusion net. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1585–1594, 2019. 8

[43] Shuang Xu, Jiangshe Zhang, Zixiang Zhao, Kai Sun, Junmin Liu, and Chunxia Zhang. Deep gradient projection networks for pan-sharpening. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1366–1375, 2021. 2, 3, 6

[44] Yinsong Yan, Junmin Liu, Shuang Xu, Yicheng Wang, and Xiangyong Cao. Md^{3}net: Integrating model-driven and data-driven approaches for pansharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–16, 2022. 3

[45] Junfeng Yang, Xueyang Fu, Yuwen Hu, Yue Huang, Xinghao Ding, and John Paisley. Pannet: A deep network architecture for pan-sharpening. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5449–5457, 2017. 2, 3, 6

[46] Lequan Yu, Xianzhi Li, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. Pu-net: Point cloud upsampling network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2790–2799, 2018. 3

[47] Qiangqiang Yuan, Yancong Wei, Xiangchao Meng, Huanfeng Shen, and Liangpei Zhang. A multiscale and multidepth convolutional neural network for remote sensing imagery pan-sharpening. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(3):978–989, 2018. 6

[48] Zixiang Zhao, Jiangshe Zhang, Shuang Xu, Zudi Lin, and Hanspeter Pfister. Discrete cosine transform network for guided depth map super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5697–5707, 2022. 8

[49] Man Zhou, Jie Huang, Keyu Yan, Gang Yang, Aiping Liu, Chongyi Li, and Feng Zhao. Normalization-based feature selection and restitution for pan-sharpening. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 3365–3374, 2022. 3

[50] Man Zhou, Jie Huang, Keyu Yan, Hu Yu, Xueyang Fu, Aiping Liu, Xian Wei, and Feng Zhao. Spatial-frequency domain information integration for pan-sharpening. In *European Conference on Computer Vision*, pages 274–291. Springer, 2022. 3, 6

[51] Man Zhou, Keyu Yan, Jie Huang, Zihe Yang, Xueyang Fu, and Feng Zhao. Mutual information-driven pan-sharpening. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1798–1808, 2022. 3

[52] Man Zhou, Keyu Yan, Jinshan Pan, Wenqi Ren, Qi Xie, and Xiangyong Cao. Memory-augmented deep unfolding network for guided image super-resolution. *International Journal of Computer Vision*, 131(1):215–242, 2023. 3