

Article

A Graph Laplacian Regularizer from Deep Features for Depth Map Super-Resolution

George Gartzonikas ¹, Evaggelia Tsiligianni ¹, Nikos Deligiannis ² and Lisimachos P. Kondi ^{1,*}

¹ Department of Computer Science and Engineering, University of Ioannina, 451 10 Ioannina, Greece; g.gartzonikas@uoi.gr (G.G.); etsiligia@uoi.gr (E.T.)

² Department of Electronics and Informatics, Vrije Universiteit Brussel & IMEC, 1050 Brussels, Belgium; nikos.deligiannis@vub.be

* Correspondence: lkoni@uoi.gr

Abstract: Current depth map sensing technologies capture depth maps at low spatial resolution, rendering serious problems in various applications. In this paper, we propose a single depth map super-resolution method that combines the advantages of model-based methods and deep learning approaches. Specifically, we formulate a linear inverse problem which we solve by introducing a graph Laplacian regularizer. The regularization approach promotes smoothness and preserves the structural details of the observed depth map. We construct the graph Laplacian matrix by deploying latent features obtained from a pretrained deep learning model. The problem is solved with the Alternating Direction Method of Multipliers (ADMM). Experimental results show that the proposed approach outperforms existing optimization-based and deep learning solutions.

Keywords: depth map super-resolution; graph-based regularization; alternating direction method of multipliers



Academic Editor: Burkhard Wünsche

Received: 30 April 2025

Revised: 8 June 2025

Accepted: 13 June 2025

Published: 17 June 2025

Citation: Gartzonikas, G.; Tsiligianni, E.; Deligiannis, N.; Kondi, L.P. A Graph Laplacian Regularizer from Deep Features for Depth Map Super-Resolution. *Information* **2025**, *16*, 501. <https://doi.org/10.3390/info16060501>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Depth maps play a crucial role in various applications such as autonomous driving [1], 3D reconstruction [2], augmented reality [3], and semantic scene understanding [4]. There are two main sensing technologies to capture depth information, either passive or active. The most common approach for passive technologies is stereo reconstruction, that is, the estimation of the scene depth from two images of the scene by using stereo matching algorithms [5]. The Time-of-Flight (ToF) camera [6] and structure light scanner [7] are two representative depth sensors for real-time active depth sensing. However, due to hardware limitations and sensor noise, captured depth maps often suffer from low resolution and artifacts, posing serious issues for various 3D applications.

Depth map super-resolution (SR) aims to reconstruct high-resolution (HR) depth maps from their low-resolution (LR) counterparts. Similar to image SR, depth map SR is an ill-conditioned inverse problem, requiring additional information to be solved. In general, depth cameras are equipped with an additional RGB sensor that can capture color images with a resolution higher than that of depth maps. Most of the existing works apply guided depth map SR, that is, they employ the HR color counterpart as guidance for the SR task. Although color images are useful for depth enhancement, the edges in color images do not exactly match the edges in depth maps. Moreover, in low-light conditions, the guidance image is usually noisy and may mislead the depth restoration algorithm, making color-assisted approaches less general [8,9].

To this end, some researchers address single-depth map SR by employing methods similar to single-modal image SR [8,9]. Single-modal SR tasks have been addressed by analytical, filtering and data-driven methods. Analytical methods try to solve the inverse problem by introducing various priors [10]. Among them, graph-based regularization is a popular approach [11–15]. Filtering-based methods estimate the depth of pixels by performing a weighted average of local pixels, and the weights are obtained by the affinity calculated from RGB–D image pairs [16–18]. Data-driven methods learn mapping functions from LR depth maps to HR depth maps [9,19,20]. With the rapid development of deep neural networks, deep learning methods have achieved state-of-the-art performance in depth map SR [21,22]. Despite their impressive results, DL methods require large computational resources and datasets, which can limit their applicability in real-world scenarios. Model-based deep learning designs are an alternative approach that tries to bridge the gap between deep learning and model-based solutions [15,23,24].

In this paper, we follow a model-based approach for the solution of single depth map SR, which also leverages the representation power of deep neural networks. Specifically, we consider a graphical representation of depth maps and address the depth map SR as an inverse problem. In order to incorporate prior knowledge, promote smoothness, and preserve the structural integrity of the depth maps, we propose a regularization of the problem at a feature level. We extract latent features from LR depth maps using a pretrained neural network model and apply graph regularization by computing a graph Laplacian using the extracted features. The estimated SR depth maps are obtained as a solution to the graph-regularized problem using the Alternating Direction Method of Multipliers (ADMM) algorithm. The experimental results demonstrate the superior performance of the proposed approach against single-modal and guided SR methods.

The remainder of the paper is organized as follows: Section 2 reviews the related work. Section 3 presents the mathematical formulation and the details of the proposed algorithm. Section 4 provides the experimental setup, the datasets, the experiments, and the results of the algorithm. Section 5 investigates the impact of each component of our framework via ablation studies. Finally, Section 6 concludes the paper.

2. Related Work

2.1. Depth Map Super-Resolution

We briefly review three categories of depth map SR methods, that is, analytical or model-based methods, filtering-based methods, and data-driven methods.

Similar to color image SR, analytical approaches for depth map SR address the estimation of the HR depth map from the observed LR counterpart as an inverse problem and introduce various priors [10]. The corresponding optimization methods apply different regularization techniques. Total variation (TV) regularization aims to minimize the gradient magnitude to enforce smoothness. Although the TV constraint helps to distinguish the edge and the noise, it tends to generate “staircasing artifacts” for the smooth region [10]. Recent studies have shown the powerful capability of the graph Laplacian to deal with piecewise smooth signals, and several works have used it to exploit the smoothness of the depth map [11–15].

Filtering-based methods combine both computational efficiency and structure preservation. Tomasi and Manduchi [16] proposed a bilateral filtering method, which smooths images while preserving edges and considers both spatial proximity and intensity similarity. However, the fact that it relies on local neighborhoods limited the ability to capture long-range dependencies. Buades et al. [17] addressed this problem with non-local means (NLMs) filtering, which extends filtering beyond local neighborhoods by leveraging self-similar patches across the entire image. While NLMs filtering enhances noise reduction

and texture preservation, it suffers from high computational costs and potential artifacts in complex structures. To improve efficiency, He et al. [18] proposed an edge-aware approach that takes advantage of a high-resolution guidance image to improve depth map super-resolution.

Data-driven solutions are an alternative approach to address SR tasks and include dictionary learning and deep learning methods. Dictionary learning methods rely on sparsity assumptions and try to learn the correlation between the LR and the HR space from a set of training image pairs [9]. Wang et al. [19] introduced a depth map super-resolution technique that integrates multi-directional dictionary learning with autoregressive (AR) modeling. This method involves training multiple dictionaries, each corresponding to specific geometric directions, to effectively represent directional depth patches. Tomic and Drewes [20] proposed a method for learning overcomplete dictionaries that jointly represent image intensity and scene depth. They developed a novel Joint Basis Pursuit (JBP) algorithm to identify related sparse features across these two modalities, allowing for effective depth inpainting and improved 3D scene representation.

Deep learning (DL) methods rely on large amounts of data to find a direct mapping from the LR observations to the HR ground truth. Pioneer DL designs for general image SR include SRCNN [21], which demonstrated the potential of CNN data-driven approaches, ESRGAN [22], which relied on generative adversarial neural networks, RESNET [25] with its variants (RESNET50, RESNET101, etc.), which introduced residual learning, and U-net [26], which showed that skip connections are useful for recovering fine details. State-of-the-art DL models for depth map SR include a multi-scale fusion model proposed in [27] where the multi-scale guided features are obtained by a visual geometry group (VGG)-like neural network, a multi-modal attention-based fusion model [28], a high-frequency guidance network that employs the octave convolution [29], and the model presented in [30], which uses a guidance image only at the training stage. Super-Resolution Graph Attention Networks (SRGATs) [31] introduced a graph-based deep learning framework for single-image SR, utilizing Graph Attention Networks (GATs) to model complex and non-local relationships between pixels. By representing the image as a graph and applying attention mechanisms, the method enhances the reconstruction of high-frequency details, leading to improved visual quality in the outputs.

2.2. Graph-Based Representations

In graph-based representations, we assume that the signal is represented in the form of a weighted undirected graph \mathcal{G} , and similarities in the signal are expressed by the edges \mathcal{E} and the respective weights encoded in the adjacency matrix W . Graph-based representations provide a powerful tool in computer vision, as they capture complex image structures and relationships between pixels [14,32]. In image processing, typically, the nodes of the graph correspond to image pixels, and the graph Laplacian is used to express similarity constraints between the pixels. Graph-based techniques have been widely applied in various image processing tasks, including segmentation, denoising, and super-resolution, due to their ability to model spatial dependencies and structural patterns [33]. Spectral graph theory has been leveraged to design effective regularization strategies that enhance feature preservation and structural coherence in reconstructed images. Graph-based regularization has been incorporated both in model-based approaches [32] and deep learning designs [14,15].

Concerning depth map reconstruction, a graph Laplacian model exploiting prior information about the depth image and the corresponding color image was formulated in [11]. Instead of constructing the graph with pixels, the authors of [12] proposed to construct the graph with a group of similar patches. In [13], the edge weight distribution of

an area with sharp edges was considered to be a bimodal distribution. A reweighted graph Laplacian regularizer was proposed to preserve sharp edges and promote the bimodal distribution of edge weights. In [15], the graph Laplacian was learned from their data, and the regularized problem was solved by a DL design in an end-to-end manner.

Building upon these foundations, we consider a graph-based representation of the depth maps and compute the graph Laplacian matrix using latent features. Following an approach similar to [15], we obtain the latent representations from a deep neural network model. Unlike [15], where the latent features and the inverse mapping were learned from their data in an end-to-end learning framework, we use a pretrained model to extract the features and estimate the HR depth map by employing the ADMM algorithm to solve the corresponding graph-regularized optimization problem. We compare our results with state-of-the-art ADMM-based methods performing single depth map SR with other types of regularization [34,35]. We also compare our work with state-of-the-art guided depth map SR methods that use a neural network as a graph regularizer [15]. The experiments show that our method yielded the best results in terms of the root mean squared error (RMSE) and visual quality.

3. Depth Map Super-Resolution Using the Admm Algorithm and Graph-Based Regularization

We address depth map SR as a graph-based regularized optimization problem. We considered a graphical representation of depth maps, extracted latent features from LR depth maps using a pretrained neural network model, and computed a graph Laplacian using the extracted features. We experimented with different feature extractors. The obtained problem was solved with ADMM.

3.1. Depth Map Super-Resolution as an Inverse Problem

Consider $\mathbf{y} \in \mathbb{R}^{N_1 N_2}$ as the vectorized form of an observed $N_1 \times N_2$ depth map (LR) and $\mathbf{x} \in \mathbb{R}^{S^2 N_1 N_2}$ as the unknown HR depth map, assuming upscaling by a factor S . Then, the degrading process is described by the following equation:

$$\mathbf{y} = A\mathbf{x} + \epsilon, \quad (1)$$

where $A \in \mathbb{R}^{N_1 N_2 \times S^2 N_1 N_2}$ is the degradation matrix, and ϵ is the additive noise. Since there can be various HR depth maps with a slight difference in camera angle, illumination, material properties, and other variables for a certain LR depth map, (1) is an ill-posed inverse problem [10]. Seeking a unique solution, we need additional prior knowledge which we incorporate into the problem with regularization, that is,

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{y} - A\mathbf{x}\|_2^2 + \lambda R(\mathbf{x}), \quad (2)$$

where $R(\mathbf{x})$ is a regularization term weighted by $\lambda > 0$.

Problem (2) is of the form

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} [g(\mathbf{x}) + h(\mathbf{x})], \quad (3)$$

where $g(\mathbf{x}) = \|\mathbf{y} - A\mathbf{x}\|_2^2$ is the fidelity term that enforces consistency with the observed data, and $h(\mathbf{x}) = \lambda R(\mathbf{x})$ is the regularization term. We can rewrite (3) as follows:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} [g(\mathbf{x}) + h(\mathbf{z})], \quad \text{s.t.} \quad \mathbf{x} - \mathbf{z} = 0, \quad (4)$$

and use the ADMM algorithm for its solution [36]. We formulate the augmented Lagrangian as follows:

$$\mathcal{L}_p(\mathbf{x}, \mathbf{z}, \mathbf{u}) = g(\mathbf{x}) + h(\mathbf{z}) + \mathbf{u}^\top (\mathbf{x} - \mathbf{z}) + \frac{\rho}{2} \|\mathbf{x} - \mathbf{z}\|_2^2, \quad (5)$$

where \mathbf{u} is the dual variable, and ρ is the augmented Lagrangian parameter [36]. Then, at the k -th iteration of ADMM, we perform the following: (i) We minimize the augmented Lagrangian with respect to the primal variables \mathbf{x} and \mathbf{z} . (ii) We update the dual variable \mathbf{u} with the minimizers. By performing these operations, ADMM takes the form of Algorithm 1 [36], where $\text{prox}_{\rho g}(\cdot)$ [$\text{prox}_{\rho h}(\cdot)$] denotes the proximal operator of ρg (ρh); the arrow \leftarrow denotes an assignment operation. The proximal operator can be calculated according to [36] as follows:

$$\text{prox}_{\rho g}(\mathbf{v}) = \arg \min_{\mathbf{x}} \{\rho g(\mathbf{x}) + \frac{1}{2} \|\mathbf{x} - \mathbf{v}\|_2^2\}. \quad (6)$$

Algorithm 1 ADMM algorithm

```

1: Input:  $\mathbf{u}^0 = 0$ ,  $\mathbf{x}^0$ , and  $\rho > 0$ 
2: for  $k = 1, 2, \dots, t$  do
3:    $\mathbf{z}^k \leftarrow \text{prox}_{\rho g}(\mathbf{x}^{k-1} - \mathbf{u}^{k-1})$ 
4:    $\mathbf{x}^k \leftarrow \text{prox}_{\rho h}(\mathbf{z}^k + \mathbf{u}^{k-1})$ 
5:    $\mathbf{u}^k \leftarrow \mathbf{u}^{k-1} + (\mathbf{z}^k - \mathbf{x}^k)$ 
6: end for
7: Return:  $\mathbf{x}^t$ 

```

3.2. Graph-Based Regularization for Depth Map Super-Resolution

In order to promote smooth upscaling and also preserve the structural information, we consider a graphical representation of a depth map and propose a regularizer that encodes the spatial relationships between pixels by employing the graph Laplacian matrix. Assuming that each pixel in the depth map corresponds to a node in a graph and connections between neighboring pixels are represented by edges, the graph Laplacian matrix is given by

$$L = D - W, \quad (7)$$

where W is the adjacency matrix of the graph, and D is the degree matrix. Therefore, we define a regularizer of the form

$$R(\mathbf{x}) = \mathbf{x}^T L \mathbf{x}. \quad (8)$$

We calculate the proximal operator for $h(\mathbf{x}) = \lambda \mathbf{x}^T L \mathbf{x}$ by solving the following optimization problem:

$$\text{prox}_{\rho h}(\mathbf{v}) = \arg \min_{\mathbf{x}} \{\rho h(\mathbf{x}) + \frac{1}{2} \|\mathbf{x} - \mathbf{v}\|_2^2\}. \quad (9)$$

We minimize (9) by setting the gradient to zero. Then, the calculation of the proximal operator induces to the solution of a quadratic minimization problem, which can be reformulated as a linear system:

$$(\rho I + 2\lambda L)\mathbf{x} = \rho \mathbf{v}, \quad (10)$$

where ρ is the penalty parameter for the ADMM algorithm, and I is the identity matrix with shape $S^2 N_1 N_2 \times S^2 N_1 N_2$. This is a linear system of equations that can be efficiently solved using modern numerical methods, such as conjugate gradient descent. The matrix $(\rho I + 2\lambda L)$ is sparse, which makes these methods computationally feasible even for large-scale depth map SR tasks.

3.3. Feature-Based Graph Laplacian Matrix

Our approach considers the construction of a Laplacian matrix based on the similarity between pixels on a latent space. Following the approach of recent methods that employ pretrained neural networks [37–40], we deploy a deep feature extractor to obtain features for every pixel of the depth map and encode the similarity between features using a Gaussian kernel. For each pixel in the depth map, we consider its 4-connected neighbors. Then, the (i, j) element of the adjacency matrix W of the pixels graph contains the weights of the graph edges and is determined as follows:

$$W_{ij} = \exp\left(-\frac{\|\mathbf{f}_i - \mathbf{f}_j\|_2^2}{\sigma^2}\right), \quad (11)$$

where \mathbf{f}_i and \mathbf{f}_j are the feature vectors with size equal to fdim , and σ is a scaling parameter to control the sensitivity of the similarity to feature differences. For non-neighboring pixels, we set $W_{ij} = 0$.

As a deep feature extractor, we employed a pretrained neural network. Since we wanted to learn per pixel features and our task involved spatial dependencies, we explored different encoder–decoder deep learning models that have been used successfully in dense prediction tasks, that is, tasks that require representations at a pixel level. Specifically, we investigated the use of the following models.

DeepLabv3 [41] is an excellent feature extraction method for pixel-level information and has been deployed in tasks like semantic segmentation. The model captures multi-scale contextual information by applying dilated (atrous) convolutions with different rates and can understand both local and global features. This is important when working with complex structures in depth estimation tasks.

LinkNet [42] is designed to be lightweight while maintaining strong feature extraction performance through residual connections. Although simpler than the rest of the models tested, it has proven effective in segmentation tasks.

PAN [43] combines spatial attention mechanisms with multi-scale feature extraction. The model includes both global context and fine detail, which are valuable components when dealing with predictions like depth map estimation. Its attention mechanisms help refine features at different levels, making it an interesting candidate for our comparisons.

Finally, **U-Net** [26] has shown an excellent performance in tasks requiring detailed structure preservation. Its skip connections and balanced encoder–decoder architecture allow for rich feature extraction while retaining spatial precision. All the aforementioned models have an encoder–decoder architecture. We obtained a deep feature representation of a considered depth map as the output of the decoder part. Concerning their implementation, all encoders can be deployed with different backbone neural network blocks, e.g., ResNet34, ResNet50, etc. We tested various encoder backbones and ultimately selected ResNet-50 pretrained on ImageNet, as it consistently yielded the best results in our experiments.

The whole process for the construction of the Laplacian matrix is depicted in Figure 1. The feature extractor receives as input a bicubic interpolated depth map and computes features with dimensions equal to $\text{fdim} \times SN_1 \times SN_2$, with $\text{fdim} = 64$. After computing the adjacency matrix, the degree matrix D is calculated as

$$D_{ii} = \sum_j W_{ij}. \quad (12)$$

Finally, the Laplacian matrix is obtained from (7).

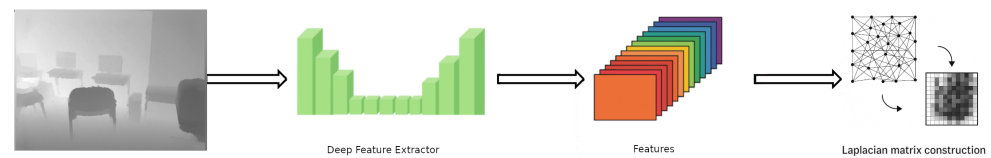


Figure 1. Flowchart presenting the process of the Laplacian matrix construction. We pass the depth map into the deep feature extractor and construct the Laplacian matrix from the obtained features instead of traditionally using the depth map pixels.

Figure 2 provides a visual summary of the complete workflow of our proposed depth map super-resolution method. Starting from a low-resolution depth map, we first extract deep feature representations using a pretrained encoder–decoder neural network. These features capture semantic and spatial information, which is used to define the affinity between pixels in a graph structure. The resulting graph adjacency matrix allows us to compute the Laplacian, which acts as a regularizer promoting smoothness and structure preservation. The reconstruction step solves a convex optimization problem using the ADMM algorithm, integrating both the fidelity to the observed depth values and the structural priors encoded in the graph. The final result is a refined, high-resolution depth map.

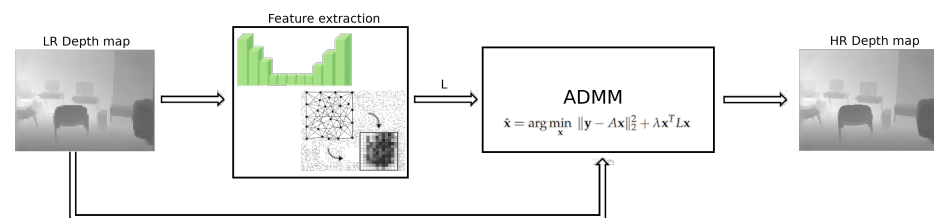


Figure 2. Overview of the proposed depth map super-resolution pipeline. A low-resolution depth map is first passed through a pretrained feature extractor to obtain latent features. These features are used to compute the adjacency matrix of a graph, which is then used to construct the graph Laplacian regularizer. The depth map is subsequently reconstructed by solving an optimization problem that balances data fidelity and graph-based smoothness via ADMM. The output is a high-resolution depth map with preserved structure and reduced artifacts.

Our method integrates graph-based regularization directly into the ADMM framework, and unlike learning-based methods that require extensive training on large datasets, our method does not include any task-specific training or fine-tuning. This significantly reduces the need for large-scale training datasets and allows for broader applicability without the overhead of specialized training. Moreover, we do not use complementary information from a guided color HR image; we only use a single depth map as input.

4. Results and Evaluation

In this section, we investigate the performance of the proposed method by providing numerical and visual experimental results. We performed two categories of experiments. The first involved the deployment of different feature extractors which determined the selection of the proposed model. The second involved the evaluation of the proposed framework against state-of-the-art methods.

4.1. Experimental Setup

We conducted experiments on two widely used depth map datasets, namely, DIML and NYUv2. DIML is a dataset which contains both indoor and outdoor real-world depth maps; we focused on the indoor depth maps. NYUv2 also contains indoor depth maps

captured from RGB–D sensors. The experiments involved 5030 samples from the DIML dataset and 600 samples from the NYUv2.

For our experiments, we used depth map patches of size 256×256 . We also rescaled the input and output depth maps in the range $[0, 1]$. For $\times 4$ upsample, the parameters of the proposed ADMM method were set as follows: $\lambda = 1.61$, $\rho = 0.11$, and $\sigma = 2.71$. For the $\times 8$ upsample, we used $\lambda = 1.01$, $\rho = 0.11$, and $\sigma = 3.21$. These parameters were selected through tuning on a validation set. Specifically, we performed a grid search and evaluated performance using RMSE. The chosen values correspond to those that achieved the best performance on the validation set. All the deployed models used RESNET50 as a backbone encoder pretrained on ImageNet. We ran the algorithm for a total of 15 iterations. For numerical evaluation, we used the average root mean squared error (RMSE).

To obtain the features from the deployed feature extractors, we used the Segmentation Models PyTorch, which is a library with pretrained neural networks for image segmentation based on PyTorch (<https://segmentation-models-pytorch.readthedocs.io/en/latest/docs/api.html#api>) (accessed on 29 April 2025)). For the code implementation, we used the Scientific Computational Imaging Code (SCICO) [44], which is a powerful open-source software framework designed for solving imaging and inverse problems using state-of-the-art computational algorithms. The hardware setup we used was the following: cpu Intel i3 12100, 16 GB ram, and MSI Geforce RTX 4060 graphics card.

4.2. Performance Comparison Across Selected Models

We present results of the proposed approach by deploying four well-established deep learning models for feature extraction, that is, U-net [26], DeepLabv3 [41], LinkNet [42], and the Pyramid Attention Network (PAN) [43]. The datasets were processed in the same way for all models to ensure a fair and unbiased comparison.

The numerical results obtained for each model are presented in Table 1. For $\times 4$ up-sampling, all models demonstrated a high performance level; however, U-Net consistently outperformed the others. For $\times 8$ upsampling, the difference between U-Net and all other models became noticeable, showcasing that for higher upsampling rates, the rest of the models performed worse.

Table 1. Comparison of U-Net with other segmentation models on NYUv2 and DIML datasets in terms of RMSE.

	U-Net [26]	DeepLabV3 [41]	LinkNet [42]	PAN [43]
$\times 4$ NYUv2	0.015409	0.015611	0.015600	0.015612
$\times 8$ NYUv2	0.026526	0.027544	0.027555	0.027546
$\times 4$ DIML	0.015288	0.015473	0.015462	0.015473
$\times 8$ DIML	0.025857	0.026851	0.026865	0.026852

In Figure 3, we present visual results to compare the reconstructed depth maps for the four tested models at upsampling factors $\times 4$ and $\times 8$. As for the $\times 4$ case, the visual differences between the models are not particularly pronounced. This aligns with the RMSE results, where all the models performed similarly without noticeable differences. However, for $\times 8$ upsampling, the differences became more evident. All other models appeared slightly noisier, less consistent, and with more artifacts, whereas the U-Net model maintained smooth transitions. These visual observations confirm that U-Net is more effective under more challenging upsampling conditions.

The results we obtained reinforced our choice to choose U-Net as the foundational model for our work. While more complex architectures like DeepLabV3 [41] and PAN [43] offer powerful context-aware mechanisms, they did not match the accuracy of the U-Net.

Also, although more simplistic architectures like LinkNet [42] could offer lightweight and accurate features, they could not produce results as good as U-Net.

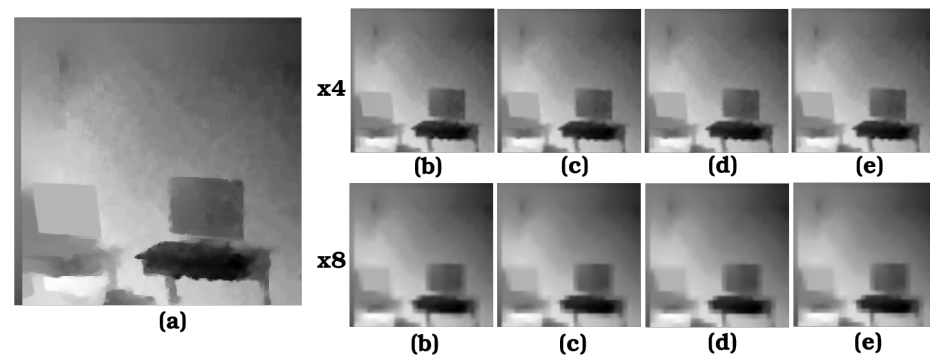


Figure 3. Depth map SR results for upsampling factors $\times 4$ and $\times 8$. Picture (a) represents the original image, whereas the rest of the pictures represent the reconstructed depth maps for (b) the U-Net method [26], (c) the DeepLabV3 method [41], (d) the LinkNet method [42], and (e) the PAN method [43]. In $\times 4$ upsampling rate, the visual differences are subtle, as the RSME differs only by a small margin. In $\times 8$ upsampling rate, the visual difference becomes more clear—for example, in the left corner of the bolder computer, depth map (b) is less pixelated and more smooth than the others.

4.3. Comparison with Other Methods

To assess the effectiveness of our method, we compared it against state-of-the-art ADMM-based methods that used different regularizers. Specifically, we compared it with Plug-and-Play (PnP) methods like DnCNN-ADMM [45], which is a method integrating deep learning within ADMM, using the DnCNN denoiser for regularization. We also compared it with BM3D-ADMM [35], which leverages non-local self-similarities to remove noise and enhance textures, and TV-ADMM [34], which enforces smoothness in images by using total variation. We further made a comparison against the guided depth map SR method presented in [15], which applies graph-based regularization using a deep learning framework. Finally, we conducted experimental comparison on the Deep Attentional Guided Image Filtering (DAGF) method [46], which uses attention-based filters to adaptively transfer structural information from a guidance image to a target image, enabling accurate and edge-aware image restoration across tasks like super-resolution and texture removal. For visual comparison, we also included the results from bicubic upsampling.

Table 2 presents average results for $\times 4$ and $\times 8$ upsampling factors. As can be seen, the results indicate that our method outperformed all the single-modal ADMM-based methods used as a baseline. On the NYUv2 dataset, it also outperformed the state-of-the-art DL method presented in [15], which employs a guidance HR color image. Since our method achieved better results than [15], it follows that it also outperformed all other methods used for comparison in [15], that is, DKN [47], FDKN [47], and FDSR [29]. We also observed that in most cases ($\times 8$ upsampling rate on NYUv2 dataset and $\times 4$, $\times 8$ upsampling rates on DIML dataset) our method outperformed by a higher margin the second state-of-the-art DL method presented in [46], for which we made a comparison, and only in the $\times 4$ upsampling rate on the NYUv2 dataset did it perform worse, but with a small difference. In summary, our method outperformed all existing single-modal ADMM-based methods, whereas we remained competitive and in some cases outperformed the state-of-the-art DL methods that used the RGB image as side information.

Note that our method does not need any adaptation to the data and instead relies on a pretrained network integrated into an ADMM framework. As a result, as seen in Table 3, while the per-image execution time may appear higher than that of some deep learning methods (e.g., DAGF [46] and the approach by de Lutio et al. [15]), those reported

times only reflect inference and omit the substantial training cost required to obtain the model. In contrast, our method leverages a fixed pretrained model and does not require any retraining, making it more practical and computationally efficient in scenarios where training resources are limited or when applying the method to new domains. On the other hand, compared to the other ADMM variants, our method has a slightly higher per-image processing time compared to TV-ADMM and DnCNN-ADMM due to the computation of the Laplacian matrix that returned better image quality. In comparison to BM3D-ADMM, we have a lower execution time.

Table 2. Comparison of methods on NYUv2 and DIML datasets in terms of RMSE.

	Single Depth Map SR				Guided Depth Map SR	
	Proposed	DnCNN-ADMM [45]	TV-ADMM [34]	BM3D-ADMM [35]	DAGF [46]	de Lutio et al. [15]
×4 NYUv2	0.0154	0.0188	0.0232	0.0273	0.0141	0.0203
×8 NYUv2	0.0265	0.0338	0.0413	0.0666	0.0292	0.0277
×4 DIML	0.0152	0.0161	0.0212	0.0242	0.0201	0.0127
×8 DIML	0.0258	0.0303	0.0393	0.0617	0.0309	0.0187

Table 3. Comparison of the considered methods in terms of execution time (seconds). For the deep learning methods [15,46], we only consider testing time. Execution time does not depend on the upsampling factor.

	Proposed	DnCNN-ADMM [45]	TV-ADMM [34]	BM3D-ADMM [35]	DAGF [46]	de Lutio et al. [15]
NYUv2	8.15	1.75	1.21	8.84	0.05	0.11
DIML	8.30	1.3	1.99	9.5	0.09	0.08

While the numerical results provide an objective measure of performance, the visual quality is also a significant aspect for depth map SR. In Figure 4, we present the reconstructed depth maps for ×4 and ×8 upsampling factors and different methods. Specifically, we present (a) the HR color image, (b) the reference HR depth map patch, (c) the corresponding LR patch, the reconstructed HR depth maps obtained with (d) bicubic interpolation, (e) BM3D-ADMM [35], (f) the method of de Lutio et al. [15], (g) DnCNN-ADMM [45], (h) TV-ADMM [34], (j) DAGF [46] and (i) our method.

We also include additional visualizations presented in Figure 5 in the form of error maps, where darker regions (black) indicate low reconstruction error, and brighter colors—ranging from red to orange—highlight areas of higher error. These maps provide further insight into the spatial distribution of reconstruction inaccuracies across different methods. The first three rows correspond to the ×4 upsampling rate, while the last three illustrate results for the ×8 rate. As shown, our method consistently produced the lowest reconstruction error across all scenes, with noticeably fewer high-error regions compared to the other approaches.

As can be seen, our method demonstrates sharper edge preservation and better depth consistency in structured regions and object boundaries. We observe that the TV-ADMM method produces overly smoothed and blurry depth maps that lose lots of information of depth and also quality. BM3D and DnCNN produced better results compared to TV-ADMM, but they also have the same problem, that is, the depth maps are smoother than expected, and they do not have sufficient details. We also observed that BM3D tended to introduce residual noise in texture-heavy regions. The method of De Lutio et al. [15], which uses a guidance HR image, outperformed the other baseline methods, likely due to the additional learned weights enhancing depth variations. However, our method reconstructed the image with finer details and fewer artifacts and provided a good balance between sharpness and smoothness.



Figure 4. Depth map SR results for upsampling factors $\times 4$ and $\times 8$. The reconstructed depth maps obtained with our method in (i) are compared with (d) bicubic interpolation, (e) BM3D-ADMM [35], (f) the method of de Lutio et al. [15], (g) DnCNN-ADMM [45], (h) TV-ADMM [34], and (j) DAGF [46]. We also present (a) the HR color image, (b) the ground truth, and (c) the downsampled depth maps.

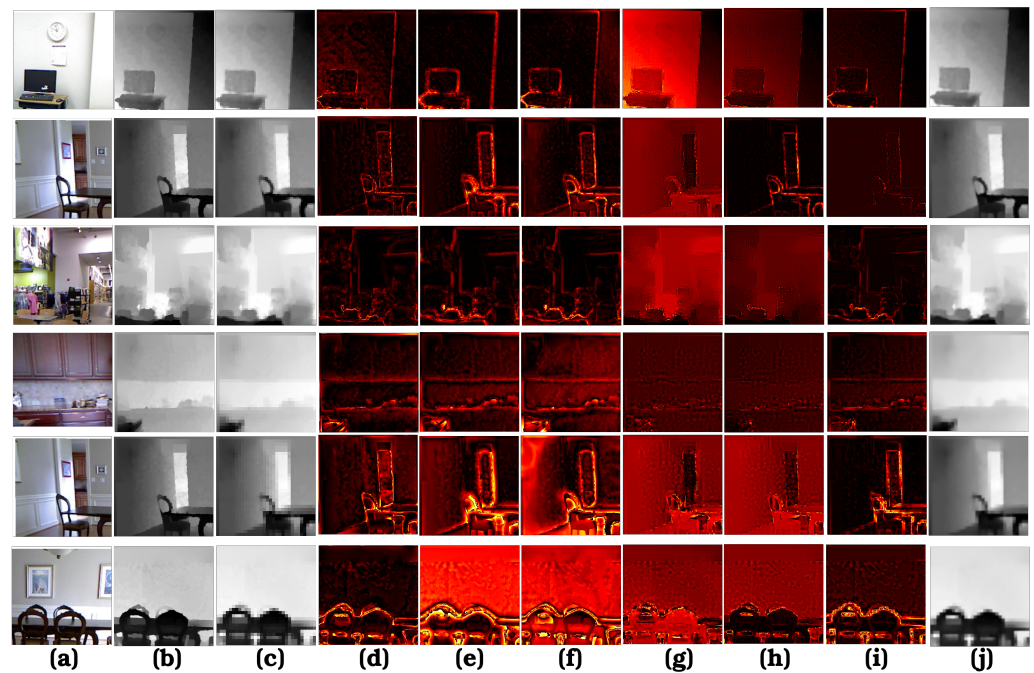


Figure 5. Error maps computed as the absolute difference between the ground truth HR depth maps and the reconstructed outputs for various methods. (a) represents the HR RGB image, (b) the HR depth map, (c) the downsampled depth map, the error maps of (d) DnCNN-ADMM [45], (e) TV-ADMM [34], (f) BM3D-ADMM [35], (g) DAGF [46], (h) the method of de Lutio et al. [15], (i) our method, and (j) the predicted depth map of our method. First three rows display results for $\times 4$ upsampling rate, whereas the remaining three display the $\times 8$ upsampling rate. Low-error areas appear in darker colors, whereas areas where the error becomes higher are represented with increasingly brighter colors ranging from red to orange.

5. Ablation Study

In this section, we present the ablation studies we implemented. Specifically, we analyzed the contribution of key components in our proposed graph-based SR framework. Our study initially focused on varying the number of output fdim in U-Net, which affects the capacity of the neural network. Then, we conducted experiments on various number of ADMM iterations to investigate the convergence and also the output of the framework numerically and visually. Finally, we explored the employment of complementary information from a guidance RGB image, which we used as side information.

Experimental Setup and Results

We performed experiments on the NYUv2 and DIML datasets for both $\times 4$ and $\times 8$ upsampling factors, using RMSE as an evaluation metric. As a baseline, we used the single depth map SR framework, deploying a U-Net feature extractor with fdim = 64, and a total of 15 iterations of the ADMM algorithm. In order to test how each of the numbers of fdim values and the number of iterations affect the framework, we altered one variable at a time and kept the others unchanged.

In Table 4, we observe that changing the number of output fdim values in U-Net had an insignificant impact on the RMSE across both datasets and scaling factors. The performance remained nearly the same for 64, 128, and 256 fdim values. The model with 64 fdim values achieved in most cases either the best or equivalent results compared to larger values of the number of fdim values. This suggests that increasing the number of output fdim values does not provide any benefit, while it increases the computational

and memory demands. Therefore, choosing 64 as the number of fdim values is the most efficient choice.

Table 4. Results of different number of fdim values of U-Net.

fdim	64	128	256
×4 NYUv2	0.01539	0.01539	0.01539
×8 NYUv2	0.02652	0.02654	0.02656
×4 DIML	0.01528	0.01528	0.01527
×8 DIML	0.02585	0.02588	0.02590

Next, Table 5 showcases the effect of an HR guidance color image as side information. The input depth map and the corresponding RGB image were concatenated into a 3D array of size $4 \times \text{height} \times \text{width}$ of the depth map. The concatenated input was passed to the feature extractor, and the graph Laplacian matrix was constructed as described in our proposed framework. The results indicate that including side information offered no measurable improvement, and in some cases, it performed slightly worse. For example, in the ×8 upsampling rate on both NYUv2 and DIML, removing side information led to better RMSE scores. This indicates that the network is able to extract sufficient information from the single-input modality (depth map), and adding the corresponding RGB image as side information introduces more noise rather than useful guidance. As a result, we conclude that side information is unnecessary for this task in our framework.

Table 5. Comparison of guided vs. single depth map SR.

Side Info	Yes	No
×4 NYUv2	0.01540	0.01540
×8 NYUv2	0.02665	0.02652
×4 DIML	0.01530	0.01528
×8 DIML	0.02598	0.02585

Finally, Table 6 presents the impact of different numbers of ADMM iterations. We observed a consistent improvement in the RMSE as the number of iterations increased from 5 to 15. However, beyond 15 iterations, the performance became stable, with minimal or no further improvement. This was especially evident in the ×4 NYUv2 case, where the RMSE remained constant from 10 to 20 iterations. Similarly, for DIML at ×8, the difference between 15 and 20 iterations was very small. Therefore, 15 iterations appear to offer the best tradeoff between reconstruction quality and computational cost. Using more iterations increased runtime without clear benefits.

Table 6. Results of different numbers of ADMM iterations.

Iterations	5	10	15	20
×4 NYUv2	0.01551	0.01540	0.01540	0.01540
×8 NYUv2	0.02781	0.02678	0.02652	0.02654
×4 DIML	0.01535	0.01529	0.01528	0.01529
×8 DIML	0.02717	0.02615	0.02585	0.02583

6. Conclusions

In this work, we addressed single depth map SR with a model-based approach incorporating priors of the form of a graph-based regularizer. The obtained optimization problem was solved with the ADMM algorithm. The proposed method relied on a pretrained deep

neural network model to capture the latent features of the considered depth maps and estimate the regularizer. The experimental results on two benchmark datasets have shown that the proposed method outperformed other ADMM-based solutions as well as deep learning-based guided SR approaches in terms of both the numerical and visual results.

Author Contributions: Conceptualization, G.G., E.T., N.D. and L.P.K.; methodology, G.G., E.T., N.D. and L.P.K.; software, G.G.; validation, G.G., E.T., N.D. and L.P.K.; resources, G.G., E.T., N.D. and L.P.K.; writing—original draft preparation, G.G., E.T., N.D. and L.P.K.; writing—review and editing, G.G., E.T., N.D. and L.P.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Availability of data and material: The DIML dataset is available in the following link: <https://dimlrgbd.github.io/> (accessed on 29 April 2025). The NyuV2 dataset is available in the following link: https://cs.nyu.edu/~fergus/datasets/nyu_depth_v2.html (accessed on 29 April 2025).

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ADMM	Alternating Direction Method of Multiplier
HR	High Resolution
LR	Low Resolution
TV	Total Variation
NLM	Non-Local Mean
AR	AutoRegressive
JBP	Joint Basis Pursuit
DL	Deep Learning
CNNs	Convolutional Neural Networks
SRCNNs	Super Resolution Convolutional Neural Networks
ESRGANs	Enhanced Super-Resolution Generative Adversarial Networks
RESNET	Residual Neural Network
VGG	Visual Geometry Group
SRGATs	Super Resolution Graph Attention Networks
GAT	Graph Attention Network
RMSE	Root Mean Squared Error
SCICO	Scientific Computational Imaging Code
PnP	Plug-and-Play
PAN	Pyramid Attention Network
DAGF	Deep Attentional Guided Image Filtering

References

1. Song, Z.; Lu, J.; Yao, Y.; Zhang, J. Self-supervised depth completion from direct visual-LiDAR odometry in autonomous driving. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 11654–11665. [[CrossRef](#)]
2. Li, J.; Gao, W.; Wu, Y. High-quality 3D reconstruction with depth super-resolution and completion. *IEEE Access* **2019**, *7*, 19370–19381. [[CrossRef](#)]
3. Ndjiki-Nya, P.; Koppel, M.; Doshkov, D.; Lakshman, H.; Merkle, P.; Muller, K.; Wiegand, T. Depth image-based rendering with advanced texture synthesis for 3-D video. *IEEE Trans. Multimed.* **2011**, *13*, 453–465. [[CrossRef](#)]

4. Wang, F.; Pan, J.; Xu, S.; Tang, J. Learning discriminative cross-modality features for RGB-D saliency detection. *IEEE Trans. Image Process.* **2022**, *31*, 1285–1297. [[CrossRef](#)]
5. Shankar, K.; Tjersland, M.; Ma, J.; Stone, K.; Bajracharya, M. A learned stereo depth system for robotic manipulation in homes. *IEEE Robot. Autom. Lett.* **2022**, *7*, 2305–2312. [[CrossRef](#)]
6. Lange, R.; Seitz, P. Solid-state time-of-flight range camera. *IEEE J. Quantum Electron.* **2001**, *37*, 390–397. [[CrossRef](#)]
7. Herrera, D.; Kannala, J.; Heikkilä, J. Joint depth and color camera calibration with distortion correction. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2058–2064. [[CrossRef](#)] [[PubMed](#)]
8. Xie, J.; Feris, R.S.; Sun, M.T. Edge-guided single depth image super resolution. *IEEE Trans. Image Process.* **2015**, *25*, 428–438. [[CrossRef](#)]
9. Xu, W.; Zhu, Q.; Qi, N. Depth map super-resolution via joint local gradient and nonlocal structural regularizations. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 8297–8311. [[CrossRef](#)]
10. Zhong, Z.; Liu, X.; Jiang, J.; Zhao, D.; Ji, X. Guided depth map super-resolution: A survey. *ACM Comput. Surv.* **2023**, *55*, 1–36. [[CrossRef](#)]
11. Zhang, Y.; Feng, Y.; Liu, X.; Zhai, D.; Ji, X.; Wang, H.; Dai, Q. Color-guided depth image recovery with adaptive data fidelity and transferred graph Laplacian regularization. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 320–333. [[CrossRef](#)]
12. Yan, C.; Li, Z.; Zhang, Y.; Liu, Y.; Ji, X.; Zhang, Y. Depth image denoising using nuclear norm and learning graph model. *ACM Trans. Multimed. Comput. Commun. Appl. TOMM* **2020**, *16*, 1–17. [[CrossRef](#)]
13. Wang, J.; Sun, L.; Xiong, R.; Shi, Y.; Zhu, Q.; Yin, B. Depth map super-resolution based on dual normal-depth regularization and graph Laplacian prior. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 3304–3318. [[CrossRef](#)]
14. Xu, B.; Yin, H. Graph convolutional networks in feature space for image deblurring and super-resolution. In Proceedings of the 2021 International Joint Conference on Neural Networks (IJCNN), Virtual, 18–22 July 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–8.
15. De Lutio, R.; Becker, A.; D’Aronco, S.; Russo, S.; Wegner, J.D.; Schindler, K. Learning graph regularisation for guided super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 1979–1988.
16. Tomasi, C.; Manduchi, R. Bilateral filtering for gray and color images. In Proceedings of the Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271), Bombay, India, 4–7 January 1998; IEEE: Piscataway, NJ, USA, 1998; pp. 839–846.
17. Buades, A.; Coll, B.; Morel, J.M. A non-local algorithm for image denoising. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision And Pattern Recognition (CVPR’05), San Diego, CA, USA, 20–25 June 2005; IEEE: Piscataway, NJ, USA, 2005; Volume 2, pp. 60–65.
18. He, K.; Sun, J.; Tang, X. Guided image filtering. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 1397–1409. [[CrossRef](#)]
19. Wang, J.; Xu, W.; Cai, J.F.; Zhu, Q.; Shi, Y.; Yin, B. Multi-direction dictionary learning based depth map super-resolution with autoregressive modeling. *IEEE Trans. Multimed.* **2019**, *22*, 1470–1484. [[CrossRef](#)]
20. Tomic, I.; Drewes, S. Learning joint intensity-depth sparse representations. *IEEE Trans. Image Process.* **2014**, *23*, 2122–2132. [[CrossRef](#)]
21. Dong, C.; Loy, C.C.; He, K.; Tang, X. Learning a deep convolutional network for image super-resolution. In *Computer Vision—ECCV 2014, Proceedings of the 13th European Conference, Zurich, Switzerland, 6–12 September 2014*; Proceeding Part IV 13; Springer: Berlin/Heidelberg, Germany, 2014; pp. 184–199.
22. Wang, X.; Yu, K.; Wu, S.; Gu, J.; Liu, Y.; Dong, C.; Qiao, Y.; Change Loy, C. Esrgan: Enhanced super-resolution generative adversarial networks. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018.
23. Marivani, I.; Tsiligianni, E.; Cornelis, B.; Deligiannis, N. Multimodal deep unfolding for guided image super-resolution. *IEEE Trans. Image Process.* **2020**, *29*, 8443–8456. [[CrossRef](#)] [[PubMed](#)]
24. Tsiligianni, E.; Zerva, M.; Marivani, I.; Deligiannis, N.; Kondi, L. Interpretable deep learning for multimodal super-resolution of medical images. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Strasbourg, France, 27 September–1 October 2021; Springer: Berlin/Heidelberg, Germany, 2021; pp. 421–429.
25. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 29–30 June 2016; pp. 770–778.
26. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015, Proceedings of the 18th International Conference, Munich, Germany, 5–9 October 2015*; Proceedings Part III 18; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.
27. Hui, T.W.; Loy, C.C.; Tang, X. Depth map super-resolution by deep multi-scale guidance. In *Computer Vision—ECCV 2016, Proceedings of the 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016*; Proceedings Part III 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 353–369.

28. Zhong, Z.; Liu, X.; Jiang, J.; Zhao, D.; Chen, Z.; Ji, X. High-resolution depth maps imaging via attention-based hierarchical multi-modal fusion. *IEEE Trans. Image Process.* **2021**, *31*, 648–663. [\[CrossRef\]](#)
29. He, L.; Zhu, H.; Li, F.; Bai, H.; Cong, R.; Zhang, C.; Lin, C.; Liu, M.; Zhao, Y. Towards fast and accurate real-world depth super-resolution: Benchmark dataset and baseline. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 9229–9238.
30. Sun, B.; Ye, X.; Li, B.; Li, H.; Wang, Z.; Xu, R. Learning scene structure guidance via cross-task knowledge transfer for single depth super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 7792–7801.
31. Yan, Y.; Ren, W.; Hu, X.; Li, K.; Shen, H.; Cao, X. SRGAT: Single image super-resolution with graph attention network. *IEEE Trans. Image Process.* **2021**, *30*, 4905–4918. [\[CrossRef\]](#)
32. Rossi, M.; Frossard, P. Geometry-consistent light field super-resolution via graph-based regularization. *IEEE Trans. Image Process.* **2018**, *27*, 4207–4218. [\[CrossRef\]](#)
33. Shuman, D.I.; Narang, S.K.; Frossard, P.; Ortega, A.; Vandergheynst, P. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Process. Mag.* **2013**, *30*, 83–98. [\[CrossRef\]](#)
34. Rudin, L.I.; Osher, S.; Fatemi, E. Nonlinear total variation based noise removal algorithms. *Phys. D* **1992**, *60*, 259–268. [\[CrossRef\]](#)
35. Dabov, K.; Foi, A.; Katkovnik, V.; Egiazarian, K. Image restoration by sparse 3D transform-domain collaborative filtering. In *Image Processing: Algorithms and Systems VI*; SPIE: Bellingham, WA, USA, 2008; Volume 6812, pp. 62–73.
36. Parikh, N.; Boyd, S. Proximal algorithms. *Found. Trends Optim.* **2014**, *1*, 127–239. [\[CrossRef\]](#)
37. Renaud, M.; Prost, J.; Leclaire, A.; Papadakis, N. Plug-and-play image restoration with stochastic denoising regularization. In Proceedings of the Forty-First International Conference on Machine Learning, Vienna, Austria, 21–27 July 2024.
38. Zhu, Y.; Zhang, K.; Liang, J.; Cao, J.; Wen, B.; Timofte, R.; Van Gool, L. Denoising diffusion models for plug-and-play image restoration. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 1219–1229.
39. Wu, Y.; Zhang, Z.; Wang, G. Unsupervised deep feature transfer for low resolution image classification. In Proceedings of the IEEE/CVF International Conference On Computer Vision Workshops, Seoul, Republic of Korea, 27–28 October 2019.
40. Zhang, K.; Li, Y.; Zuo, W.; Zhang, L.; Van Gool, L.; Timofte, R. Plug-and-play image restoration with deep denoiser prior. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 6360–6376. [\[CrossRef\]](#) [\[PubMed\]](#)
41. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.
42. Chaurasia, A.; Culurciello, E. Linknet: Exploiting encoder representations for efficient semantic segmentation. In Proceedings of the 2017 IEEE Visual Communications And Image Processing (VCIP), St. Petersburg, FL, USA, 10–13 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1–4.
43. Li, H.; Xiong, P.; An, J.; Wang, L. Pyramid attention network for semantic segmentation. *arXiv* **2018**, arXiv:1805.10180.
44. Balke, T.; Davis Rivera, F.; Garcia-Cardona, C.; Majee, S.; McCann, M.T.; Pfister, L.; Wohlberg, B.E. Scientific computational imaging code (SCICO). *J. Open Source Softw.* **2022**, *7*, 4722. [\[CrossRef\]](#)
45. Kamilov, U.S.; Bouman, C.A.; Buzzard, G.T.; Wohlberg, B. Plug-and-play methods for integrating physical and learned models in computational imaging: Theory, algorithms, and applications. *IEEE Signal Process. Mag.* **2023**, *40*, 85–97. [\[CrossRef\]](#)
46. Zhong, Z.; Liu, X.; Jiang, J.; Zhao, D.; Ji, X. Deep attentional guided image filtering. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *35*, 12236–12250. [\[CrossRef\]](#)
47. Kim, B.; Ponce, J.; Ham, B. Deformable kernel networks for joint image filtering. *Int. J. Comput. Vision* **2021**, *129*, 579–600. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.