applied sciences

MDPI

Article

Depth Completion in Autonomous Driving: Adaptive Spatial Feature Fusion and Semi-Quantitative Visualization

Hantao Wang [1], Ente Guo [2,*], Feng Chen [1] and Pingping Chen [1]

[1] College of Physics and Information Engineering, Fuzhou University, Fuzhou 350108, China
[2] College of Computer and Control Engineering, Minjiang University, Fuzhou 350108, China
* Correspondence: guoente@mju.edu.cn

Abstract: The safety of autonomous driving is closely linked to accurate depth perception. With the continuous development of autonomous driving, depth completion has become one of the crucial methods in this field. However, current depth completion methods have major shortcomings in small objects. To solve this problem, this paper proposes an end-to-end architecture with adaptive spatial feature fusion by encoder–decoder (ASFF-ED) module for depth completion. The architecture is built on the basis of the network architecture proposed in this paper, and is able to extract depth information adaptively with different weights on the specified feature map, which effectively solves the problem of insufficient depth accuracy of small objects. At the same time, this paper also proposes a depth map visualization method with a semi-quantitative visualization, which makes the depth information more intuitive to display. Compared with the currently available depth map visualization methods, this method has stronger quantitative analysis and horizontal comparison ability. Through experiments of ablation study and comparison, the results show that the method proposed in this paper exhibits a lower root-mean-squared error (RMSE) and better small object detection performance on the KITTI dataset.

Keywords: autonomous driving; depth completion; multi-source information fusion for sensing; image processing; computer vision

Citation: Wang, H.; Guo, E.; Chen, F.; Chen, P. Depth Completion in Autonomous Driving: Adaptive Spatial Feature Fusion and Semi-Quantitative Visualization. *Appl. Sci.* **2023**, *13*, 9804. https://doi.org/10.3390/app13179804

Academic Editor: Dimitris Mourtzis

Received: 9 July 2023
Revised: 13 August 2023
Accepted: 29 August 2023
Published: 30 August 2023

1. Introduction

With the continuous development of technology, autonomous driving has become one of the most compelling problems in the field of computer vision. The implementation of autonomous driving cannot be achieved without the perception and understanding of the environment [1]. In autonomous driving systems, depth information is crucial to real-time perception and decision processes. Depth information can provide important information about object localization, distance and shape, which can help autonomous driving systems make accurate judgments and decisions. How to obtain accurate depth information has become an important problem in the research of autonomous driving.

Currently, LiDAR is a widely used hardware device among the depth information collection methods. LiDAR calculates the distance of an object by emitting a laser beam and measuring its return time. Compared with other sensors, LiDAR has high precision and can provide reliable depth information. However, LiDAR has some limitations, the most prominent of which is the sparsity of the data due to the sampling nature of this method. The point cloud of the LiDAR, which emits 64 beams, are projected into the camera's coordinate system, and the depth information accounts for about 5% of the total pixels [2]. In practical applications, the point cloud data collected using LiDAR are usually very limited and do not provide a complete pixel-level depth map.

To overcome the sparsity of LiDAR data, depth completion has become an important algorithm. Depth completion is a method that fills a sparse depth map into a dense depth map. It estimates the missing depth information by using a multi-sensor fusion approach.

Therefore, depth completion has a wide range of promising applications in autonomous driving systems and has an important impact on the safety of the system.

To improve the performance of depth completion, researchers have proposed an impressive array of methods, including sparse invariant convolution [2] and multimodal fusion strategies [3–8]. However, it is difficult to fill the depth map by relying on only 5% of the depth information. To overcome this challenge, significant progress has been made in recent years in the use of RGB images to guide depth completion. These methods use RGB images as an additional information source to improve the performance of the depth completion. The main methods on RGB images include supervised learning [9–17], reconstructing stereo images [18–22], temporal relations [23–25], and other related sequence relations [26]. In a related study, Schneider et al. [27] provided powerful guidance information for depth completion by using RGB information to generate sharp depth boundaries. They improved the performance of the network by using geodesic distance as a constraint. In addition, S2D [28] used a self-supervised deep neural network based on residual blocks to fuse LiDAR data and RGB images in the same feature space, thus effectively improving the prediction accuracy. In addition, DeepLiDAR [29] provides a better guide for depth completion by predicting surface normal vectors using RGB data.

These encouraging research works show that depth completion using image guidance has great potential to improve the accuracy of prediction results and has attracted increasing research interest. However, despite the progress made, there are still some challenges and problems to be addressed. For example, existing depth completion methods still have performance limitations in small objects. Therefore, this paper proposes a depth completion method based on adaptive feature fusion, aiming to improve the performance of existing methods, especially for small objects.

Inspired by [30], the core idea of the method in this paper is to propose an end-to-end architecture with adaptive spatial feature fusion by encoder–decoder (ASFF-ED) module, and this architecture is able to achieve adaptive adjustment during feature extraction by optimizing pixel-level weights. Specifically, when extracting features on feature maps of different scales, the adaptive feature fusion module is able to extract detailed information of small objects on large-scale feature maps, while extracting the overall features on large scales on small-scale feature maps. This strategy effectively alleviates the problem of depth information loss in the downsampling process, thus improving the performance of the depth completion.

In addition, this paper also proposes a semi-quantitative visualization to solve the problem that the existing depth map visualization cannot be analyzed quantitatively to compare the experimental results. Based on the consideration of human color sensitivity, we design a semi-quantitative visualization, which is a non-uniform quantizing method, to achieve better visualization. This method helps to present the depth information intuitively and is especially suitable for practical application scenarios such as the visual monitoring of autonomous driving systems. In summary, the method presented in this paper has the following contributions:

(1) Devised a novel depth completion architecture with ASFF-ED module, which achieves satisfactory results in solving the depth completion of small objects and improving performance. Through combining the network proposed in this paper with ASFF-ED, the method in this paper can better handle the depth information of small-sized objects and improve the overall algorithm robustness and accuracy.
(2) Proposed a depth map visualization method called semi-quantitative visualization. We also focus on some problems of current depth map visualization methods, such as the inability to perform quantitative analysis and the difficulty of horizontal comparison. This method can display the depth information in a more intuitive way, so that the observer can understand and analyze the depth map more clearly.
(3) The proposed algorithm in this paper was proven to be superior in depth completion through comparative experiments and ablation experiments and provides a new solution for depth completion in the field of autonomous driving systems.

2. Adaptive Spatial Feature Fusion Depth Completion

This chapter will provide a comprehensive introduction to our depth completion architecture, including the network architecture, the adaptive feature fusion module, and the loss function. In addition, we will introduce a visualization method of the depth map called semi-quantitative visualization and demonstrate its validity.

2.1. Network Architecture

This section presents an encoder–decoder network called DepthNet. The architecture of this network is shown in Figure 1.

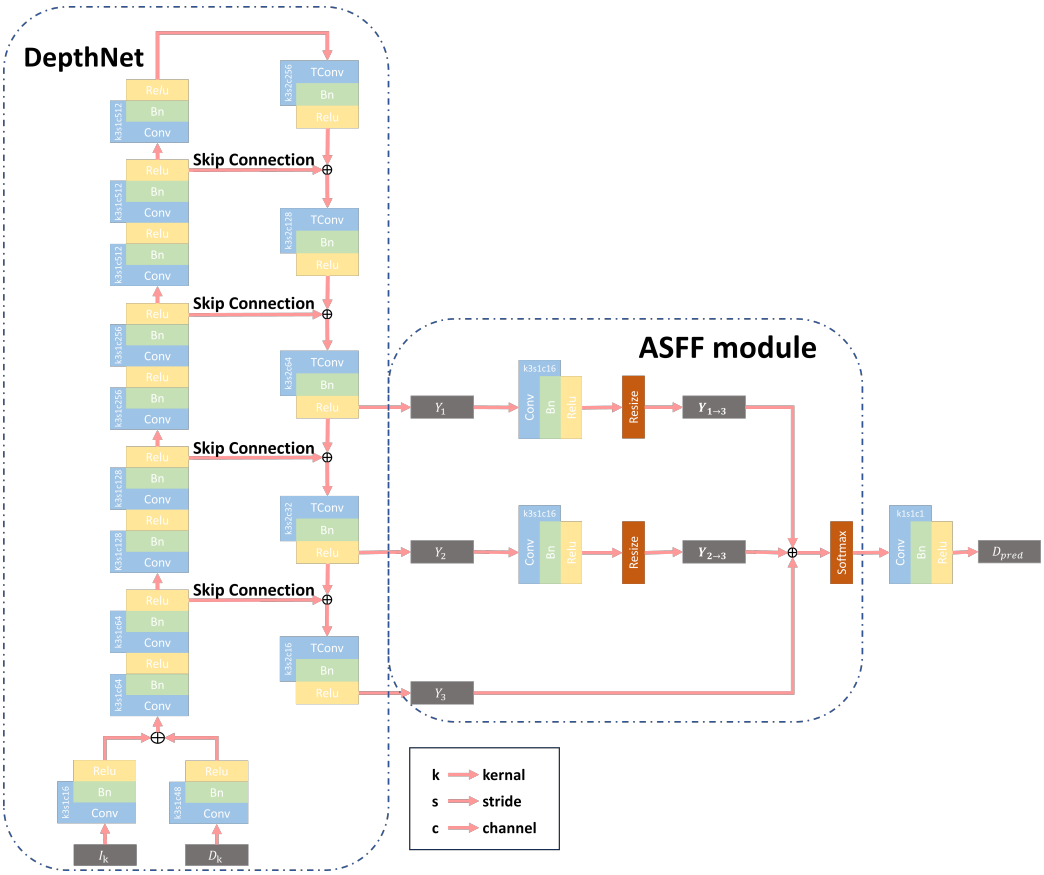

Figure 1. The network architecture of DepthNet, where Conv means convolution, Tconv means transpose convolution, Bn means batch normalization, Relu means rectified linear unit activation function, k3s1c16 means convolution operation with a step size of 1 using 16 convolution kernels of size 3×3, and other similar representations and so on. The output of DepthNet is passed through the ASFF-ED module to obtain the D_{pred}.

The input of this network consists of a sparse depth map D_k and an RGB image I_k in the same coordinate system, and the output is the predicted dense depth map D_{pred}. Before processing, I_k and D_k are first fed to a convolution with a kernel size of 3×3 for simple feature extraction, respectively, and then the two feature maps are concatenated and input to the encoder. A block module similar to resnet34 [31] is used at the encoder to downsample the image by 16 times. This block module is able to extract deep features in the feature map and avoid the problem of gradient disappearance.

The decoder upsamples the feature map that has been downsampled by 16 times through the transpose convolution to obtain a feature map of the same size as the input original image. At the same time, the feature information corresponding to the scale at the encoder is fed to the decoder through a skip connection, which helps to decode the image

information better and prevents the gradient from disappearing. Finally, the output results Y_1, Y_2 and Y_3 are three kinds of feature maps of different scales.

2.2. Adaptive Spatial Feature Fusion Module

In order to solve the accuracy problem of the classical algorithm Faster R-CNN [32] for small object detection in object detection, FPN [33] proposed a feature extraction method based on a feature pyramid network. Instead of limiting the extraction to feature maps at a single scale, the method divides the feature map into multiple scales and generates anchors of different sizes at each scale. Smaller objects on small-sized feature maps result in blurred features due to downsampling, when the receptive field can more easily focus on larger-sized objects. Conversely, on large-size feature maps, smaller objects have sharper textures, while larger objects tend to lose sight of the overall features due to excessive attention to detail, thus affecting their robustness.

Feature pyramid networks work by corresponding the size of an object to the scale of the feature map. When an object of a certain size is associated with a feature map of a certain scale, the object is considered to be a positive sample; while on feature maps of other scales, the object is considered to be a negative sample. The reason for this is that if the object is detected in every scale of the feature map, it can easily lead to conflicting feature information between different scales. By building a feature pyramid network based on the above idea, the size of the object is effectively correlated with the scale of the feature map, thus improving the accuracy of small object detection. This method brings a new breakthrough in the research and application in the field of object detection, and shows good results in practice.

Although we did not use a feature pyramid network to extract features, this study uses an encoder–decoder-type network and generates multi-scale feature maps at the decoder. Unlike the other depth completion methods mentioned in this paper, those methods only use the last layer of the feature map to generate the dense depth map, ignoring the potential value of feature information at other scales. Depth completion is different from object detection, but we can borrow ideas from objection detection. It is well known that objects at a distance are relatively small, while objects at close range are relatively large. In an autonomous driving scene, the top half of the depth map usually represents places far away from the camera and relatively small object contours, while the bottom half represents places closer to the camera and relatively large object contours.

Inspired by it, we have introduced ASFF-ED module into DepthNet. This method extracts feature map information near the top pixels in larger-sized feature maps, and near the bottom pixels in smaller-sized feature maps. By adaptively fusing multi-scale features, we are able to enhance the completion of the contours when predicting depth, better understand the differences in features between near and far objects and improve the accuracy of depth information for small objects.

For the above theory, we have three feature maps of different sizes, denoted Y_1, Y_2 and Y_3, according to Figure 1. These feature maps have different resolutions and number of channels. To facilitate the computation, the ASFF-ED uses a 1×1 convolution kernel to modify the number of channels of Y_1 and Y_2, and uses the upsampling method to align the resolution of Y_1 and Y_2 with Y_3. This results in the unified feature maps $Y_{1\to3}$ and $Y_{2\to3}$ after the channels and resolution modifications, representing the result of unifying the channels and resolution size with Y_3.

ASFF-ED is a method that obtains the final prediction by assigning weights to the pixel values of each input image and summing them together. The output of the adaptive feature module is set to be a dense depth map, where the i-th pixel is denoted as Y^i_{pred}. Y^i_{pred} is calculated as shown below:

$$Y^i_{pred} = \alpha_i P^i_1 + \beta_i P^i_2 + \gamma_i P^i_3 \quad (1)$$

where P_1^i, P_2^i and P_3^i represent the value of the i-th pixel in images $Y_{1\to3}$, $Y_{2\to3}$ and Y_3, respectively, while α_i, β_i and γ_i represent the weight parameters of the corresponding pixel points. In order to achieve a more convergent training process, we require that these three weight values satisfy the following relationship:

$$\alpha_i + \beta_i + \gamma_i = 1 \tag{2}$$

where, based on the above constraint relationship, we can obtain using softmax:

$$\alpha_i = \frac{e^{\mu_{\alpha_i}}}{e^{\mu_{\alpha_i}} + e^{\mu_{\beta_i}} + e^{\mu_{\gamma_i}}} \tag{3}$$

$$\beta_i = \frac{e^{\mu_{\beta_i}}}{e^{\mu_{\alpha_i}} + e^{\mu_{\beta_i}} + e^{\mu_{\gamma_i}}} \tag{4}$$

$$\gamma_i = \frac{e^{\mu_{\gamma_i}}}{e^{\mu_{\alpha_i}} + e^{\mu_{\beta_i}} + e^{\mu_{\gamma_i}}} \tag{5}$$

where μ_{α_i}, μ_{β_i} and μ_{γ_i} are the model weight parameters, respectively.

2.3. Loss Function

The loss function in this paper consists of two components: Sparse Loss and Smoothness Loss.

Sparse Loss: It aims to ensure that the predicted dense depth map has the same depth value on pixels containing depth information as in the sparse depth map, as shown in the following equation:

$$L_{dense} = \left\| D_{gt} - D_{pred>0} \right\|^2 \tag{6}$$

where D_{gt} is the ground truth of the depth map and D_{pred} is the dense depth map output using the DepthNet network. In the depth map, a pixel value greater than 0 indicates a valid value.

Smoothness Loss: Considering that the depth information predicted by the network is independent of each other, this may lead to large abrupt changes in depth information between consecutive pixels, resulting in discontinuities between image pixel values. To solve this problem, Smoothness Loss [18] is introduced in this paper to increase the overall smoothness of the predicted image. The calculation formula is:

$$L_{smooth} = \left\| D_{pred} \right\|^2 \tag{7}$$

In summary, the overall loss function in this paper can be expressed as:

$$L = \alpha L_{dense} + \beta L_{smooth} \tag{8}$$

where α and β are the weight parameters of the loss function to balance the contributions of Sparse Loss and Smoothness Loss.

2.4. Semi-Quantitative Visualization

Depth maps are stored on the computer as 16-bit grey-scale images, with the value of each pixel representing depth information. On the KITTI dataset, the farthest distance of a depth map typically only reaches to about 80 m, as the sparse depth map contains information in only 5% of the pixels, and the ground truth increases to 30% of the pixels containing information. When the image is opened directly for observation, the whole map is black and it is difficult to observe the information in the depth map. In order to better observe the information in the depth map with the naked eye, the depth map needs

to be quantified and colored. A common approach today is to quantize the depth map uniformly, using:

$$P_c = \frac{d - d_{min}}{d_{max} - d_{min}} \tag{9}$$

where P_c denotes the normalized pixel value, d denotes the depth information in the depth map, d_{max} denotes the point with the farthest distance in the depth map, and d_{min} denotes the value with the nearest distance depth in the depth map. By normalizing the depth map, it can be converted by the color map for visualization, as shown in Figure 2.

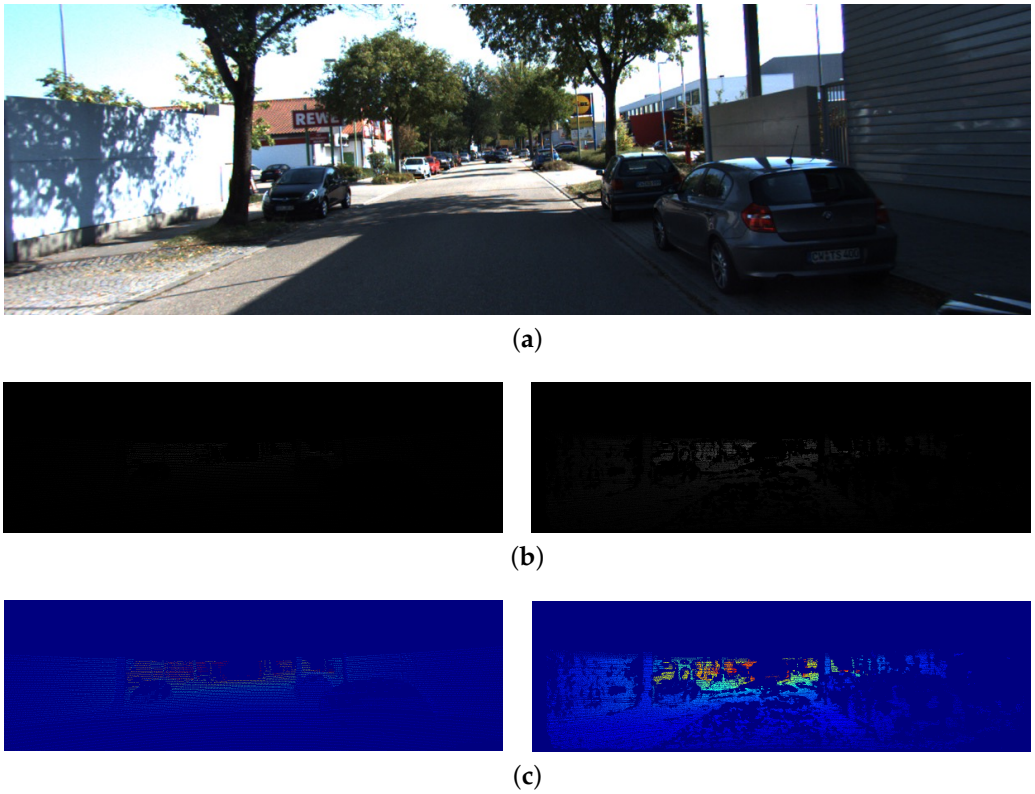

Figure 2. Comparison of original depth map and visualization. (**a**) RGB images; (**b**) original sparse depth map (**left**) and ground truth (**right**); (**c**) sparse depth map (**left**) and ground truth (**right**) after visualization.

The right of Figure 2b shows a 30% dense ground truth with a slightly visible outline. In contrast, the original sparse depth map in Figure 2b is invisible to the naked eye as only 5% of the pixels have depth information; it is simply dark. To see the depth map clearly, it needs to undergo pseudo-color coloring, as shown in Figure 2c. However, the current mean-quantization coloring methods are only applicable for observing the internal relationship of each pixel. As different images have different depth maxima and minima, this approach does not allow for a horizontal comparison of the advantages and disadvantages of each algorithm. The mean quantization method divides the pseudo-color range evenly according to the range of depth values. However, in an autonomous driving scene, the driver is more concerned with near objects than far objects.

Because of this, when assigning a color map range, more color should be given to the depth range with smaller values in order to convey changes in depth information to the driver through visual effects. In this section, a novel method of non-mean quantization for depth map visualization, a semi-quantitative visualization, is proposed.

As mentioned earlier, humans are very sensitive to changes in distance of one meter, but beyond ten meters, the perception of distance becomes blurred. In the images captured by the camera, the driver is not concerned with the movement of vehicles and pedestrians

at a distance, but is more concerned with the risk of collision with vehicles and pedestrians at a closer distance. Based on this, the quantification of depth values should be guided by the idea of assigning more color to the depth range with smaller values and less color to the depth range with larger values. As a result, each quantization range can be divided proportionally using a fixed constant scale k, as follows:

$$\frac{f(1)}{f_{max}} = \frac{f(2)}{f(1)} = \frac{f(3)}{f(2)} = \cdots = \frac{f(n)}{f(n-1)} = \frac{f_{min}}{f(n)} = k \tag{10}$$

where n is the total number of quantization range, $f(n)$ is the maximum depth value of the n-th depth value's quantization range, f_{max} is the maximum depth value of the range, and the minimum depth value $f_{min} > 0$, then we have:

$$f(1) = kf_{max} \tag{11}$$

$$f(2) = k^2 f_{max} \tag{12}$$

$$f(3) = k^3 f_{max} \tag{13}$$

$$\cdots$$

$$f_{min} = k^{n+1} f_{max} \tag{14}$$

Therefore, the following relationships exist when starting to assign a quantified range:

$$\frac{f_{min}}{f_{max}} = k^{n+1} \tag{15}$$

i.e.,

$$k = \frac{f_{min}}{f_{max}}^{\frac{1}{n+1}} \tag{16}$$

In summary, given a quantization range when the maximum depth value of the dataset is known, the range split ratio k can be calculated to unify the color map of the depth map.

3. Experiments

To validate the effectiveness of the depth completion approach based on adaptive spatial feature fusion, we selected depth completion data from the KITTI dataset and conducted experiments on KITTI video sequences containing complex scenes such as city streets, urban roads and highways. To evaluate the performance of our method, we performed a quantitative performance comparison with other depth completion algorithms and have provided a visualization of the depth completion results.

The experiments were running on a computer configured with an I9-900K CPU and a Tesla P100 GPU. Relevant software versions include CUDA 10.1, CUDNN 7.6 and Pytorch 1.11.0. We used the 64-bit operating system Windows Server 2012 R2 and ran subsequent experiments on this platform.The parameters α and β in Equation (8) are set to 1 and 0.1, respectively.

3.1. KITTI Dataset

Generating depth completion ground truth in real-world video sequences is a rather challenging computer vision task. Compared to other tasks, it requires significant time investment and is difficult to obtain 100% dense depth maps as ground truth. To address this problem, the KITTI dataset [34] provides a dataset for depth completion, which contains stereo images, sparse depth maps and ground truth. To generate the ground truth, the 3D LiDAR point cloud is first projected onto the image coordinate to obtain the sparse depth map. The ground truth is then generated using 11 consecutive frames of point cloud data.

The resulting ground truth is a semi-dense depth map with valid depth pixels present only at the bottom of the image. The dataset contains 42949 pairs of training images and 1000 validation sets for selecting models. During training, the input fed to the network had a resolution size of 1216 × 362.

3.2. Evaluation Metric

This section introduces the commonly used evaluation metrics in depth completion, mainly including root mean square error (RMSE) that quantifies the overall level of difference between predicted and true values. The physical significance of RMSE can be understood as the average distance between each predicted value and its corresponding true value, mean absolute error (MAE), that quantifies the average level of difference between predicted and true values. The physical significance of MAE can be understood as the average absolute distance between each predicted value and its corresponding true value, inverse-depth root-mean-squared error (iRMSE) and inverse-depth mean absolute error (iMAE). These metrics are used to measure the error between the predicted depth map and the ground truth, and thus to evaluate the performance of the depth completion algorithm.

Among them, the KITTI benchmark is using RMSE as the main basis for the ranking of depth completion. Under the circumstances of calculating model error sensitivities or data assimilation applications, MAEs are definitely not preferred over RMSEs. The values of iRMSE and iMAE are affected by the average of the true and predicted values, so there may be a risk of exaggerating or minimizing the actual performance of the model.

RMSE is one of the main evaluation metrics and is calculated as shown in the following equation:

$$RMSE(d - \hat{d}) = \sqrt{\frac{1}{m}\sum_{i-1}^{m}(d_i - \hat{d}_i)} \tag{17}$$

where d denotes the predicted depth map, $\hat{d}$ the ground truth of the depth map, i denotes the depth of the i-th pixel and m denotes the total number of pixels. A lower value of RMSE where the error between the predicted depth and the ground truth is smaller and the performance of depth completion is better.

In addition to the RMSE, there are other evaluation metrics that can be used to assess the depth completion algorithm. The MAE is one of them and is calculated as shown below:

$$MAE(d - \hat{d}) = \frac{1}{m}\sum_{i-1}^{m}\left|d_i - \hat{d}_i\right| \tag{18}$$

The iRMSE and the inverse-depth mean absolute error iMAE are metrics that are evaluated against the inverse depth and are calculated as follows, respectively:

$$iRMSE(d - \hat{d}) = \sqrt{\frac{1}{m}\sum_{i-1}^{m}(\frac{1}{d_i} - \frac{1}{\hat{d}_i})} \tag{19}$$

$$iMAE(d - \hat{d}) = \frac{1}{m}\sum_{i-1}^{m}\left|\frac{1}{d_i} - \frac{1}{\hat{d}_i}\right| \tag{20}$$

For these metrics, lower values indicate better performance results for depth completion.

3.3. Ablation Study

As shown in Table 1,we conducted ablation experiments to verify the performance effectiveness of the innovations proposed in this paper. We conducted two sets of experiments on the KITTI validation set for our methods, Experiments 1 and 2. Experiment 1 used only the DepthNet network for depth completion, while Experiment 2 introduced the DepthNet with ASFF-ED module. The experimental results show that all four performance metrics improve with the ASFF-ED module. This indicates that the error in the prediction results is reduced, which fully demonstrates the effectiveness of our method. Considering

the characteristics of the KITTI dataset, the ASFF-ED module can make full use of the information of different size depth maps to obtain more accurate dense depth maps. These findings further validate the feasibility and superiority of the method in this paper in practical applications. Through these ablation studies, we validate the effectiveness of our method in improving performance and provide strong support for further research.

Table 1. Quantitative evaluation on KITTI test set.

Name	Method	iRMSE (1/km)	iMAE (1/km)	RMSE (mm)	MAE (mm)
Experiment 1	DepthNet	897.587	368.446	3.212	2.021
Experiment 2	DepthNet+ASFF-ED	**857.191**	**254.503**	**2.817**	**1.203**

3.4. Depth Completion Result

The performance of the method in this paper, which has been experimentally validated on the KITTI test set, is shown in Table 2. Compared with deep completion algorithms in recent years, the method in this paper performs better in terms of the RMSE metric. Compared with methods that use multiple network models for optimizing prediction results, such as SSGP [35], the method in this paper is relatively simple to build, using a single-network architecture that only requires RGB images and depth map as input. This lightweight architectural model is more suitable for the application of autonomous driving.

The method adopts an adaptive spatial feature fusion design, which improves the depth completion performance while effectively alleviating the phenomenon of inaccurate depth prediction of small objects due to the information loss problem arising from the downsampling process. We compare the effectiveness of this paper's method with other depth completion methods in recent years, as shown in Figure 3. As can be seen from the first column of results, in the area marked by the yellow rectangular box, most algorithms are unable to accurately outline the contour shape of the car, let alone the surface texture, because the vehicle is very far away from the location. In contrast, the method in this paper is able to clearly show the edge contours of the car and can clearly observe the change in depth of the front windscreen position. In contrast, the method of pNCNN (d) [6] has completely lost information about the vehicle, while the methods of TWISE [36] and ScaffFusion-SSL [37] have distorted vehicle contours and slightly rough surface textures.

Table 2. Quantitative evaluation on KITTI test set.

Method	iRMSE (1/km)	iMAE (1/km)	RMSE (mm)	MAE (mm)
NonLearning Complete [38]	3.63	1.23	1222.00	303.82
VOICED [39]	3.56	1.20	1169.97	299.41
AdaFrame-VGG8 [40]	3.32	1.16	1125.67	291.62
ScaffFusion [37]	3.32	1.17	1121.89	282.86
SynthProj [41]	3.53	1.19	1095.26	280.42
VLW-DepthNet [42]	3.43	1.21	1077.22	282.02
KBNet [43]	2.95	1.02	1069.47	256.76
SynthProjV [41]	3.12	1.13	1062.48	268.37
LW-DepthNet [42]	2.99	1.09	991.88	261.67
pNCNN (d) [6]	3.37	1.05	960.05	251.77
IR_L2 [44]	4.92	1.35	901.43	292.36
ScaffFusion-SSL [37]	3.24	0.88	847.22	205.75
TWISE [36]	**2.08**	**0.82**	840.20	**195.58**
SSGP [35]	2.51	1.09	838.22	244.70
Ours	2.73	1.20	**818.42**	252.48

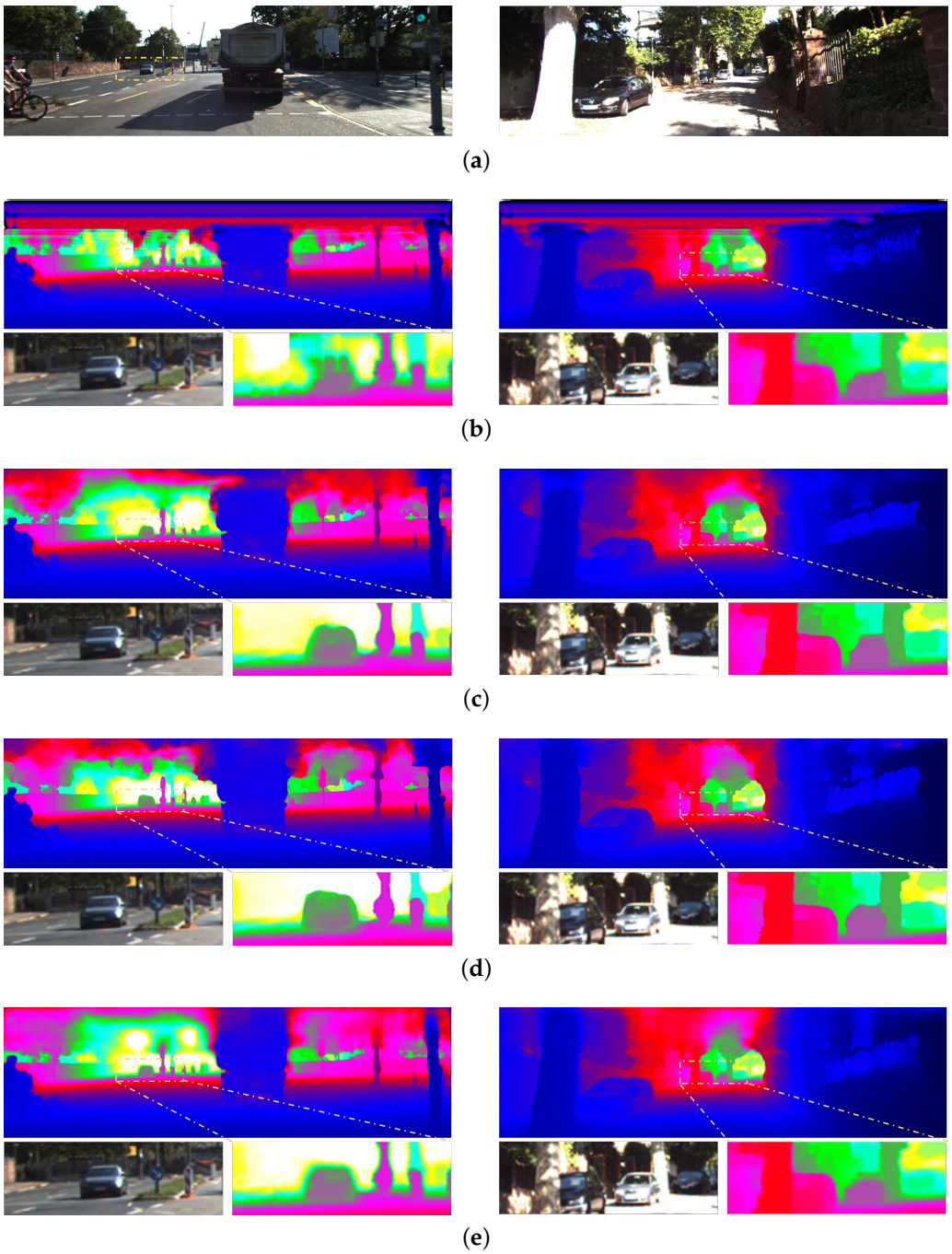

Figure 3. The comparation between ours and other depth completion methods on KITTI test set. (**a**) RGB images; (**b**) pNCNN; (**c**) ScaffFusion-SSL; (**d**) TWISE; (**e**) ours.

Further comparing the image in the second column, the strong reflections at the location of the yellow rectangular box lose much of the image texture information while producing shadows, which causes the algorithm to produce jagged dithering information during depth map estimation. However, this does not affect the algorithm's detection of small object targets. It can be observed that although the edge parts are not smooth, the vehicle and tree contours in the depth map of this paper are more similar to the original image compared to other algorithms.

Through experimental comparisons, it is verified that the method in this paper has certain advantages in small object, and the effectiveness of the ASFF-ED module for the small objects of depth completion is demonstrated. In terms of the quantitative performance metrics of the algorithm, the algorithm achieves a relatively low RMSE due to the rich

information provided by the multi-scale feature map and the combination of the flexible information fusion approach of the ASFF-ED module, which reflects the effectiveness and rationality of the algorithm.

3.5. Visualization

In this section, we set the parameters of Equation (16) to $f_{min} = 0.001$, $f_{max} = 80$ and $n = 255$. Semi-quantitative visualization is a non-mean quantization method that assigns color range based on distance. This method is more suitable for human attentional habits for distance and assigns a unique color to each depth range.

As KITTI does not make the official visual color map available, we were unable to show the visual colors in Figure 4 for comparison purposes. Therefore, we used a self-supervised depth completion from a previous study [45] and the algorithm in this paper to compare the visual differences between the mean-quantization visualization commonly used for depth completion and the semi-quantitative visualization proposed in this paper, as shown in Figure 4c.

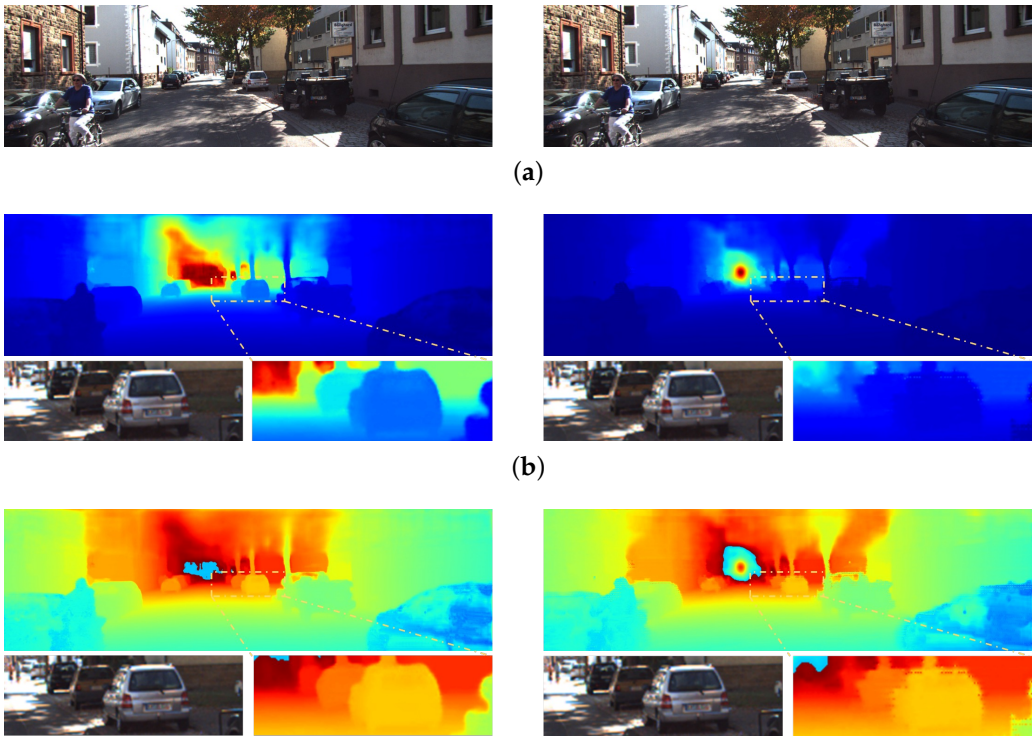

Figure 4. Visual comparison between ours (**left**) and [45] (**right**) algorithms using different visualization methods. (**a**) RGB images; (**b**) mean-quantization visualization; (**c**) semi-quantitative visualization.

The second row of the figure shows the depth maps obtained by the two algorithms using the mean-quantization visualization. This method only generates information on the relative distances in a single image. However, as the maximum and minimum depth values do not coincide from image to image, the color of each pixel obtained according to Equation (9) will also vary. The third row of the figure shows the depth maps obtained by the two algorithms using the semi-quantitative visualization. As can be seen from Figure 4, the two algorithms do not differ much in their depth predictions, but they produce a significant visual difference compared to the mean quantization visualization. The observation of the whole image shows that the mean quantization visualization assigns the color map equally to the entire depth range, which may result in a wasted color map, i.e., producing the visual effect of an all-blue color in the image. In contrast, the depth map

visualized by the semi-quantitative visualization proposed in this paper is smoother in terms of color distribution, resulting in a better color range to the nearer parts, which is in line with human visual observation habits.

The car marked in Figure 4 is about 22 metres away. If mean-quantization visualization is used, it can be obtained from Equation (9) that 0–22 m can be assigned approximately 71 bits of color in an image of 256 bits. If semi-quantitative visualization is used, we can obtain a color map according to Equations (10) and (16). By referring to the color map, we can determine that 0–22 m months is assigned to 181 bits of the color system. That is about 1.5 times more colors.

These results show that the semi-quantitative visualization has advantages in the visualization of depth completion, and is able to produce depth maps that are more in line with human visual observation habits than the traditional mean quantization visualization. This provides a useful reference for further research and application of depth completion algorithms.

4. Conclusions

The contribution of this paper is to address the safety problem of depth completion in autonomous driving by proposing an encoder–decoder network model with an ASFF-ED module embedded in it. By combining the rich information of multi-scale feature maps and enabling the network to adaptively fuse feature information from different scales, we effectively improve the information fusion capability of the model and alleviate the problem of the inaccurate detection of small-object depth information, thus reducing prediction errors. By evaluating this method on the KITTI dataset and comparing the results with other depth completion methods, we have produced satisfactory results. The experimental results show that the method proposed in this paper is lightweight, efficient and has a lower RMSE compared to other depth completion methods.

To address the problem of deep graph visualization in the field of deep completion, we propose a new algorithm for depth map visualization that is more intuitive and useful for the quantitative analysis of algorithms and horizontal comparison between algorithms. The training process of our method is easy, requires no additional information, only the annotation of depth information, and is applicable to a variety of scenes.

The research results in this paper are of great practical importance for promoting the development of autonomous driving technology and improving road safety. Future research directions can consider the further optimization of the network model and algorithms to improve the accuracy and robustness of depth completion in order to meet the demand for highly reliable depth information for autonomous driving systems.

The accuracy of depth completion is closely related to the safety of autonomous driving. Although our accuracy is better than other methods for small objects, we still have a long way to go. We are not perfect in restoring the boundary of small objects in the distance. In addition, there is still room for improvement in the accuracy of our model. In the future, we will still focus on this subject area.

Author Contributions: Conceptualization, H.W. and E.G.; methodology, H.W.; software, H.W.; validation, H.W. and E.G.; formal analysis, H.W.; investigation, H.W. and E.G.; resources, H.W.; data curation, H.W.; writing—original draft preparation, H.W.; writing—review and editing, E.G., P.C. and F.C.; visualization, H.W.; supervision, E.G.; project administration, E.G.; funding acquisition, E.G. and P.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (61871132), Education Research Project for Young and Middle aged Teachers of Fujian Provincial Department of Education (JAT220310) and Minjiang University Scientific Research Promotion Fund (MJY22025).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

Abbreviations

The following abbreviations are used in this manuscript:

ASFF-ED	Adaptive Spatial Feature Fusion by Encoder–Decoder
RMSE	Root-Mean-Squared Error
MAE	Mean Absolute Error
iRMSE	Inverse Root Mean Square Error
iRAE	Inverse Mean Absolute Error
LiDAR	Light Detection And Ranging
FPN	Feature Pyramid Network

References

1. Rehman, A.U.; Mushtaq, Z.; Qamar, M.A. Fuzzy logic based automatic vehicle collision prevention system. In Proceedings of the 2015 IEEE Conference on Systems, Process and Control (ICSPC), Bandar Sunway, Malaysia, 18–20 December 2015; IEEE: Piscataway Township, NJ, USA, 2015; pp. 55–60.
2. Uhrig, J.; Schneider, N.; Schneider, L.; Franke, U.; Brox, T.; Geiger, A. Sparsity invariant cnns. In Proceedings of the 2017 International Conference on 3D Vision (3DV), Qingdao, China, 10–12 October 2017; IEEE: Piscataway Township, NJ, USA, 2017; pp. 11–20.
3. Hua, J.; Gong, X. A normalized convolutional neural network for guided sparse depth upsampling. In Proceedings of the IJCAI, Stockholm, Sweden, 13–19 July 2018; pp. 2283–2290.
4. Eldesokey, A.; Felsberg, M.; Khan, F.S. Propagating confidences through cnns for sparse data regression. *arXiv* **2018**, arXiv:1805.11913
5. Huang, Z.; Fan, J.; Cheng, S.; Yi, S.; Wang, X.; Li, H. Hms-net: Hierarchical multi-scale sparsity-invariant network for sparse depth completion. *IEEE Trans. Image Process.* **2019**, *29*, 3429–3441. [CrossRef] [PubMed]
6. Eldesokey, A.; Felsberg, M.; Holmquist, K.; Persson, M. Uncertainty-aware cnns for depth completion: Uncertainty from beginning to end. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–18 June 2020; pp. 12014–12023.
7. Van Gansbeke, W.; Neven, D.; De Brabandere, B.; Van Gool, L. Sparse and noisy lidar completion with rgb guidance and uncertainty. In Proceedings of the 2019 16th International Conference on Machine Vision Applications (MVA), Tokyo, Japan, 27–31 May 2019; IEEE: Piscataway Township, NJ, USA, 2019; pp. 1–6.
8. Tang, J.; Tian, F.P.; Feng, W.; Li, J.; Tan, P. Learning guided convolutional network for depth completion. *IEEE Trans. Image Process.* **2020**, *30*, 1116–1129. [CrossRef]
9. Saxena, A.; Chung, S.; Ng, A. Learning depth from single monocular images. *Adv. Neural Inf. Process. Syst.* **2005**, *18* , 1161–1168.
10. Eigen, D.; Fergus, R. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 2650–2658.
11. Li, B.; Shen, C.; Dai, Y.; Van Den Hengel, A.; He, M. Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1119–1127.
12. Cao, Y.; Wu, Z.; Shen, C. Estimating depth from monocular images as classification using deep fully convolutional residual networks. *IEEE Trans. Circuits Syst. Video Technol.* **2017**, *28*, 3174–3182. [CrossRef]
13. Xu, D.; Ricci, E.; Ouyang, W.; Wang, X.; Sebe, N. Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5354–5362.
14. Fu, H.; Gong, M.; Wang, C.; Batmanghelich, K.; Tao, D. Deep ordinal regression network for monocular depth estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 8–23 June 2018; pp. 2002–2011.
15. Gan, Y.; Xu, X.; Sun, W.; Lin, L. Monocular depth estimation with affinity, vertical pooling, and label enhancement. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 224–239.
16. Lee, J.H.; Han, M.K.; Ko, D.W.; Suh, I.H. From big to small: Multi-scale local planar guidance for monocular depth estimation. *arXiv* **2019**, arXiv:1907.10326.
17. Yin, W.; Liu, Y.; Shen, C.; Yan, Y. Enforcing geometric constraints of virtual normal for depth prediction. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 5684–5693.
18. Garg, R.; Bg, V.K.; Carneiro, G.; Reid, I. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *Computer Vision–ECCV 2016, Proceedings of the 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016*; Proceedings Part VIII 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 740–756.

19. Xie, J.; Girshick, R.; Farhadi, A. Deep3d: Fully automatic 2d-to-3d video conversion with deep convolutional neural networks. In *Computer Vision–ECCV 2016, Proceedings of the 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016*; Proceedings, Part IV 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 842–857.
20. Godard, C.; Mac Aodha, O.; Brostow, G.J. Unsupervised monocular depth estimation with left-right consistency. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 270–279.
21. Kuznietsov, Y.; Stuckler, J.; Leibe, B. Semi-supervised deep learning for monocular depth map prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6647–6655.
22. Godard, C.; Mac Aodha, O.; Firman, M.; Brostow, G.J. Digging into self-supervised monocular depth estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3828–3838.
23. Mahjourian, R.; Wicke, M.; Angelova, A. Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5667–5675.
24. Wang, C.; Buenaposada, J.M.; Zhu, R.; Lucey, S. Learning depth from monocular videos using direct methods. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2022–2030.
25. Yin, Z.; Shi, J. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1983–1992.
26. Chen, W.; Fu, Z.; Yang, D.; Deng, J. Single-image depth perception in the wild. *Adv. Neural Inf. Process. Syst.* **2016**, *29* , 730–738.
27. Schneider, N.; Schneider, L.; Pinggera, P.; Franke, U.; Pollefeys, M.; Stiller, C. Semantically guided depth upsampling. In *Pattern Recognition, Proceedings of the 38th German Conference, GCPR 2016, Hannover, Germany, 12–15 September 2016*; Proceedings 38; Springer: Berlin/Heidelberg, Germany, 2016; pp. 37–48.
28. Ma, F.; Cavalheiro, G.V.; Karaman, S. Self-supervised sparse-to-dense: Self-supervised depth completion from lidar and monocular camera. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; IEEE: Piscataway Township, NJ, USA, 2019; pp. 3288–3295.
29. Qiu, J.; Cui, Z.; Zhang, Y.; Zhang, X.; Liu, S.; Zeng, B.; Pollefeys, M. Deeplidar: Deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3313–3322.
30. Liu, S.; Huang, D.; Wang, Y. Learning spatial fusion for single-shot object detection. *arXiv* **2019**, arXiv:1911.09516.
31. 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, 27–30 June 2016; pp. 770–778.
32. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28* , 91–99. [CrossRef]
33. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
34. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? the kitti vision benchmark suite. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; IEEE: Piscataway Township, NJ, USA, 2012; pp. 3354–3361.
35. Schuster, R.; Wasenmuller, O.; Unger, C.; Stricker, D. Ssgp: Sparse spatial guided propagation for robust and generic interpolation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2021; pp. 197–206.
36. Imran, S.; Liu, X.; Morris, D. Depth completion with twin surface extrapolation at occlusion boundaries. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 2583–2592.
37. Wong, A.; Cicek, S.; Soatto, S. Learning topology from synthetic data for unsupervised depth completion. *IEEE Robot. Autom. Lett.* **2021**, *6*, 1495–1502. [CrossRef]
38. Krauss, B.; Schroeder, G.; Gustke, M.; Hussein, A. Deterministic guided lidar depth map completion. In Proceedings of the 2021 IEEE Intelligent Vehicles Symposium (IV), Nagoya, Japan, 11–17 July 2021; IEEE: Piscataway Township, NJ, USA, 2021; pp. 824–831.
39. Wong, A.; Fei, X.; Tsuei, S.; Soatto, S. Unsupervised depth completion from visual inertial odometry. *IEEE Robot. Autom. Lett.* **2020**, *5*, 1899–1906. [CrossRef]
40. Wong, A.; Fei, X.; Hong, B.W.; Soatto, S. An adaptive framework for learning unsupervised depth completion. *IEEE Robot. Autom. Lett.* **2021**, *6*, 3120–3127. [CrossRef]
41. Lopez-Rodriguez, A.; Busam, B.; Mikolajczyk, K. Project to adapt: Domain adaptation for depth completion from noisy and sparse sensor data. In Proceedings of the Asian Conference on Computer Vision, Kyoto, Japan, 30 November–4 December 2020.
42. Bai, L.; Zhao, Y.; Elhousni, M.; Huang, X. DepthNet: Real-time LiDAR point cloud depth completion for autonomous vehicles. *IEEE Access* **2020**, *8*, 227825–227833. [CrossRef]
43. Wong, A.; Soatto, S. Unsupervised depth completion with calibrated backprojection layers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 12747–12756.

44. Lu, K.; Barnes, N.; Anwar, S.; Zheng, L. From depth what can you see? Depth completion via auxiliary image reconstruction. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11306–11315.
45. Chen, Z.; Wang, H.; Wu, L.; Zhou, Y.; Wu, D. Spatiotemporal guided self-supervised depth completion from lidar and monocular camera. In Proceedings of the 2020 IEEE International Conference on Visual Communications and Image Processing (VCIP), Virtual, 1–4 December 2020; IEEE: Piscataway Township, NJ, USA, 2020; pp. 54–57.