Moving Object Detection Based on Fusion of Depth Information and RGB Features

Xin Bi, Shichao Yang and Panpan Tong

School of Automotive Studies, Tongji University, Shanghai 201804, China; bixin@tongji.edu.cn (X.B.); yangshichao@tongji.edu.cn (S.Y.)
* Correspondence: tongpanpan@tongji.edu.cn
† These authors contributed equally to this work.

Abstract: The detection of moving objects is one of the key problems in the field of computer vision. It is very important to detect moving objects accurately and rapidly for automatic driving. In this paper, we propose an improved moving object detection method to overcome the disadvantages of the RGB information-only-based method in detecting moving objects that are susceptible to shadow interference and illumination changes by adding depth information. Firstly, a convolutional neural network (CNN) based on the color edge-guided super-resolution reconstruction of depth maps is proposed to perform super-resolution reconstruction of low-resolution depth images obtained by depth cameras. Secondly, the RGB-D moving object detection algorithm is based on fusing the depth information of the same scene with RGB features for detection. Finally, in order to evaluate the effectiveness of the algorithm proposed in this paper, the Middlebury 2005 dataset and the SBM-RGBD dataset are successively used for testing. The experimental results show that our super-resolution reconstruction algorithm achieves the best results among the six commonly used algorithms, and our moving object detection algorithm improves the detection accuracy by up to 18.2%, 9.87% and 40.2% in three scenes, respectively, compared with the original algorithm, and it achieves the best results compared with the other three recent RGB-D-based methods. The algorithm proposed in this paper can better overcome the interference caused by shadow or illumination changes and detect moving objects more accurately.

Keywords: moving object detection; RGB-D; CNN; super-resolution reconstruction

1. Introduction

In recent years, with the rapid development of computer vision and machine learning technologies, moving target detection based on video sequence images has become an important research topic in the field of computer vision [1]. Moving target detection refers to the segmentation of moving objects and stationary regions in an image sequence or video, and the objects with spatial position changes are extracted as the foreground [2]. It involves computer vision, machine learning, video processing and other fields. The accurate and rapid detection of moving objects is of great significance to intelligent driving, intelligent video monitoring, medical treatment and even military aspects. For example, in the traffic system, moving object detection technology is used to obtain the current traffic flow on the road; in the military field, moving target detection can be used to achieve accurate guidance.

Traditional moving object detection methods are based only on the information of RGB images taken by 2D cameras to extract objects, which can reflect the information of object color and light intensity to a certain extent, but it cannot comprehensively consider the object’s spatial location and three-dimensional information such as size and shape. In addition, the use of color images to detect objects is susceptible to the effects of shadows, occlusion and illumination, mainly for the following reasons: the target and its shadow...
mostly have the same motion properties, and both have large differences from the background, so it is easy to misjudge the shadow as the target; occlusion will lead to the loss of tracking targets; illumination changes will lead to significant changes in the image, which is not conducive to the separation of background and foreground.

This paper focuses on the above problem and proposes an innovative moving target detection algorithm that fuses depth and RGB information based on the codebook algorithm [3]. The depth information of the same scene is used to compensate for the inaccuracy of recognition due to shadow interference, light change or color interference when only color images are used for detection. The two main aspects are super-resolution reconstruction of depth images and fusion of depth and color information for the detection of moving targets. Our contribution can be summarized as follows:

- In this paper, we propose a color edge-guided depth map super-resolution CNN, which uses the correspondence between the color map and the depth map to input both the low-resolution depth map and the high-resolution color-optimized edge map of the same scene into the network, and it uses the edge map to guide the depth map reconstruction. The final experimental results show that the proposed model effectively improves the quality of the depth map after super-resolution.
- We propose an RGB-D based codebook moving object detection algorithm, which adds the depth information of the scene as an auxiliary dimension based on the original codebook algorithm, including two parts of background modeling and foreground detection, and it takes advantage of the depth information not being disturbed by shadows, illumination and occlusion to better segment moving objects.

The rest of the paper is organized as follows. Section 2 describes existing studies related to super-resolution image reconstruction and moving object detection for depth images. Section 3 presents our super-resolution reconstruction model and moving object detection algorithm. Section 4 shows the comparison of our method with existing methods. Finally, Section 5 concludes the paper and outlines future research directions.

2. Related Works

2.1. Depth Image Super-Resolution Reconstruction

In terms of super-resolution reconstruction using only a single image, Freeman et al. [4] formulated the super-resolution problem into a multi-class Markov Random Field (MRF) model, where each hidden node represents the label of a high-resolution block. However, the reconstruction process is highly dependent on the available training samples, and it may not provide reliable results when correspondence cannot be established. Dong et al. [5] integrated the traditional sparsely coded image super-resolution method process into a CNN to learn the mapping between low and high-resolution images directly during the training process, which provides higher accuracy while achieving simplification and speediness purposes. However, this method does not utilize any prior knowledge, and it is sometimes difficult to learn such many-to-one mappings, resulting in problems such as blurring or artifacts. Based on the example depth map, Mandal et al. [6] constructed the sub-dictionary of the example to restore the high-resolution depth map. Lei et al. [7] proposed a trilateral depth mapping up-sampling method based on viewpoint synthesis quality, which considered the depth smoothness, texture similarity and viewpoint synthesis quality in the up-sampling filter.

With the development of deep learning, super-resolution based on deep convolutional networks is a rapidly growing field [8], and several classes of super-resolution CNNs have emerged. The linear Super-Resolution Convolutional Neural Network (SRCNN) [5], which was first used for super-resolution, is regarded as a pioneering work in super-resolution based on deep learning and has had a great impact in the field of super-resolution. Later, the deeper network Very Deep Super-Resolution (VDSR) [9] was proposed, and the accuracy after reconstruction was further improved as the network deepened. In addition to these, there are the super-resolution residual network Cascading Residual Network (CARN) [10], recursive network Super-Resolution Feedback Network (SRFBN) [11], and the attention-based
network Holistic Attention Network (HAN) [12]; the success of these models demonstrates the usability of convolutional neural networks in the super-resolution domain.

In terms of reconstruction using multiple depth images, Rajagopalan et al. [13] modeled the original high-resolution image into MRF according to the Bayesian framework, and they used the edge adaptive MRF to deal with discontinuity, which achieved good results. Wetlz et al. [14] used the pseudo-Huber loss function as the regularization term of the objective function, and they used the quasi-Newton optimizer to calculate the minimization. With the help of Graphics Processing Unit (GPU) acceleration, high-resolution images can be restored at an interactive frame rate. Izadi et al. [15] only use depth data from Kinect to track the 3D attitude of the sensor and reconstruct the geometric accurate 3D model of the physical scene in real time.

From the aspect of color image-guided depth map super-resolution reconstruction, Ferstl et al. [16] developed a second-order TGV model based on the assumption that RGB images correspond to depth image edges, using the raw RGB intensity as input to the anisotropic diffusion tensor to enhance image detail as well as edge retention, and solving it numerically using a primal-pairing algorithm. Zhang [17] introduced the wavelet transform-extracted edge indicator function into the color-constrained canonical term on this basis to make the details clearer at the edges as well as in the images. Yang et al. [18] proposed an adaptive color-guided autoregressive model, which is constructed by the local correlation in the initial depth map and the nonlocal similarity in the color image. Song [19] proposed a progressive deep CNN framework and further optimized the reconstructed depth images by combining the constraint of having local consistency between color and depth images, and the constraint of statistical information of depth images. Wen et al. [20] proposed a coarse-to-fine CNN and constructed a color guidance strategy for depth image up-sampling based on color difference and spatial distance.

2.2. Moving Target Detection

In terms of a moving object detection algorithm, the traditional methods mainly include the inter-frame difference method [21,22], optical flow method [23,24] and background difference method [25,26]. The inter-frame difference method is generally used to extract moving objects from adjacent frames in video, and the background difference method [27] is used to detect the target by performing the difference operation between the current video frame and the background model. In recent years, with the progress of depth image acquisition technology, there are many moving target detection algorithms based on RGB-D information. On the basis of a 2D moving target detection algorithm, the depth information of the object is added to eliminate the interference of shadow, illumination change and occlusion. Parvizi et al. [28] defined the splitter by analyzing the local minimum of the probability density function for depth information, reducing over-segmentation or under-segmentation to some extent. Ottonelli et al. [29] used the ViBE (visual background extractor) background modeling algorithm to segment human contours with high reliability, and they added the compensation factor obtained by color and depth information, but it still has the problems of edge noise and missing detection. Hu et al. [30] established a Bayesian classifier at each pixel based on color and depth information, and then, they fused it to detect moving objects. Hu et al. [31] improved the reliability of a color map and depth map with different weights under three different illumination conditions by integrating the feature information of color space and depth space in the background model updating stage of the ViBE algorithm. Li et al. [32] proposed a moving object detection method combining a sequential frame difference method and optical flow algorithm with higher accuracy and efficiency. They first divided the pre-processed images in pairs of sequences and calculated the sum of the differential results, and then, they processed the differential video sequences using the optical flow method to accurately detect and identify moving targets. However, it has a large amount of calculation.

It can be seen that detection based on RGB information alone is susceptible to illumination changes and shadow interference, resulting in low detection accuracy and noise, while
some other algorithms have high detection accuracy but are computationally intensive. On
the other hand, it also shows that depth information has a strong application value in the
field of moving object detection. Therefore, this paper first studies the super-resolution
reconstruction of a depth map, proposes a CNN model framework for the super-resolution
reconstruction of a depth map based on color edge guidance, and obtains a high-quality
depth map. Then, the codebook target detection algorithm is improved by using depth
information. A moving object detection algorithm based on depth information and RGB
features fusion is proposed, which can show excellent performance in scenes such as illu-
mination change and shadow interference, and it provides a reference for target detection
methods in complex scenes.

3. Proposed Method

3.1. Depth Map Super-Resolution Reconstruction Based on Color Edge Guidance

Image super-resolution reconstruction technology has become one of the hotspots of
image research in recent years. Due to the imperfect manufacturing process of a depth
camera, it is difficult to obtain high-resolution depth maps. In addition, the traditional
deep image super-resolution technology often has the defects of blurred image edges and
increased noise after reconstruction. With the development of deep learning technology,
researchers have gradually applied deep learning technology to the field of image super-
resolution reconstruction and achieved good effects. Inspired by the SRCNN proposed by
Dong [5] and the dual-flow network described by Liu [33], and considering the guiding
effect of the edge information of high-resolution color images on the super-resolution recon-
struction of a depth map, this section proposes a convolution neural network framework
with dual-input sources. The depth image and the optimized edge map extracted from
the color image of the same scene are input into the network at the same time. The color
information is used to assist the depth map to complete the reconstruction and improve
the quality of the reconstructed image.

To facilitate the description below, the meaning of the variable symbols used in this
section is explained as follows. $I$ represents the high-resolution color map, $D_L$ represents
the low-resolution depth map, $D_H$ represents the high-resolution depth true value map,
$I_{bw}$ represents the optimized edge map, and $D_L$ represents the low-resolution depth map
of the same size as $I$.

3.1.1. Edge Graph Preprocessing

The proposed method needs to be preprocessed before inputting the image into the
CNN, including obtaining the low-resolution depth map $D_L$ with the same size as the color
map $I$ and the edge map $I_{bw}$ of color and depth fusion. $D_L$ is obtained by using the bicubic
up-sampling method for $D_L$, and the method for obtaining $I_{bw}$ refers to Reference [34].
Firstly, canny edge detection operator is performed on $D_L$ and $I$ to obtain the depth edge
map $D_e$ and color edge map $I_e$, respectively. Then, the bicubic interpolation method is used
to sample the depth edge map, and the up-sampling depth edge map $D_{ue}$ with the same
size as the color edge map is obtained. In addition, in order to prevent the misalignment of
the RGB image and the depth image, the morphological expansion operation is performed
on the color edge map $I_e$ to obtain the expanded color edge map $I_{me}$, and then, the logical
“AND” operation is performed between the expanded color edge map $I_{me}$ and the up-
sampled depth edge map $D_{ue}$. Finally, to prevent noise at the edges, the edge map is
smoothed using a Guided Image Filter [35], which effectively preserves the image edges,
and $I_{bw}$ is obtained. The process of obtaining the optimized color edge map $I_{bw}$ is shown in
Figure 1.
3.1.2. Network Model Framework

The CNN framework designed in this paper is shown in Figure 2. The input low-resolution depth map $D_L$ and the optimized color edge map $I_{bw}$ are firstly extracted from the features of the two maps through the upper and lower convolutional layers conv1 and conv2, respectively. The operations of these two convolutional layers can be expressed as follows:

$$
F_1(D_L) = \max(0, W_1 \otimes D_L + b_1) + a_1 \min(0, W_1 \otimes D_L + b_1) 
$$

(1)

$$
F_2(I_{bw}) = \max(0, W_2 \otimes I_{bw} + b_2) + a_2 \min(0, W_2 \otimes D_L + b_2) 
$$

(2)

Here, $W_1$, $W_2$ and $b_1$, $b_2$ represent the weight and bias coefficient, respectively, symbol $\otimes$ represents convolution, $W_1$ corresponds to a number of $n_1$ convolution kernels of size $c \times f_1 \times f_1$, where $c$ represents the number of channels of the input image, and $f_1 \times f_1$ is the size of the convolution kernels. The number of output feature maps is $n_1$, and $b_1$ is the $n_1$-dimensional vector. $W_2$ and $b_2$ are similar to the above. Finally, PRelu is used as the activation function, while $a_1$ and $a_2$ are $n_1$-dimensional and $n_2$-dimensional vectors, respectively, which are adaptive coefficients in PRelu.
The concat layer splices $F_1(D_L)$ and $F_2(I_{bw})$ with the channel as the axis to integrate the depth feature and color feature. This step can be expressed as:

$$\text{concat} = F_1(D_L) \oplus F_2(I_{bw})$$  \hspace{1cm} (3)$$

where the symbol $\oplus$ indicates the stitching with the channel as the axis.

The third convolution layer realizes the mapping from a low-resolution map to a high-resolution map. Finally, the fourth convolution layer is reconstructed to obtain a high-resolution depth map. These two steps can be expressed as:

$$F_3(\text{concat}) = \max(0, W_3 \otimes \text{concat} + b_3) + a_3 \min(0, W_3 \otimes \text{concat} + b_3)$$  \hspace{1cm} (4)$$

$$F(D) = W_4 \otimes F_3(\text{concat}) + b_4$$  \hspace{1cm} (5)$$

where $W_3$ corresponds to a number of $n_3$ convolution kernels of size $(n_1 + n_2) \times f_3 \times f_3$, $W_4$ corresponds to a number of $c$ convolution kernels of size $n_3 \times f_4 \times f_4$, and $b_3$ and $b_4$ are $n_1$-dimensional and $n_2$-dimensional vectors, respectively.

### 3.2. Codebook Moving Target Detection Algorithm Based on RGB-D

#### 3.2.1. Algorithm Overview

Traditional moving object detection algorithms typically use RGB color information to detect the moving objects in the video, but its detection effect is poor due to the influence of shadow, occlusion and illumination mutation. In addition, there are also problems such as a large number of dense noise points or blurred edges in the detection results. On the contrary, the depth information is mostly unaffected by shadows and illuminations. When the depth information is added to the background modeling, the target detection algorithm with more accurate detection results and better robustness can be theoretically obtained. Therefore, the moving object detection algorithm proposed in this paper is based on the codebook algorithm and adds the depth information as an extension. The color and depth information of the pixel are combined to comprehensively judge whether the pixel belongs to the foreground or background. Finally, redundant noise points are removed, and the detection results are optimized by filling holes, morphological erosion and morphological reconstruction. The algorithm is implemented by MATLAB, and the algorithm framework is shown in Figure 3.

![Figure 3. The improved RGB-D based codebook moving object detection algorithm framework in this paper.](image-url)
3.2.2. Specific Algorithm Flow

The codebook algorithm proposed by Kim et al. firstly inputs the video frame sequence; then, it uses the idea of clustering to classify each pixel, adds codewords one by one, and updates the codebook regularly to obtain the final background model. It occupies small memory, and the intuitive principle of the method is simple, including background modeling and foreground detection. The following will briefly introduce them, respectively:

- Background modeling of the original codebook algorithm

The codebook algorithm firstly uses the input $N$ video frames for training. $\chi = \{x_1, x_2, ..., x_N\}$ is used to represent that each pixel has an $N$-dimensional RGB vector, and $I = \{c_1, c_2, ..., c_L\}$ is used to represent that each pixel has a codebook with $L$ codewords. The codebook model of each pixel contains different numbers of codewords according to the change of the sample. Each codeword $c_i$ consists of an RGB vector $v_i$ and a cell $aux_i$, with six elements, where:

$$\begin{align*}
\{v_i &= (R_i, G_i, B_i) \\
\text{aux}_i &= \langle \hat{I}_i, \bar{I}_i, f_i, \lambda_i, p_i, q_i \rangle
\end{align*}$$

(6)

where $\bar{R}_i, \bar{G}_i,$ and $\bar{B}_i$, respectively, represent the average value of the color components of the pixels belonging to the codeword $c_i$. $\bar{I}_i$ and $\hat{I}_i$ represent the minimum and maximum light intensity values of the pixels belonging to the code $c_i$, respectively. $f_i$ represents the frequency of occurrence of the codeword $c_i$. $\lambda_i$ represents the maximum interval in which the code word $c_i$ is not reproduced during the training phase, and $p_i$ and $q_i$ represent the first and last time of the codeword $c_i$, respectively.

In the training phase, each pixel is compared with all the codewords in the current codebook to determine whether they can be matched to the codewords. The specific content of the algorithm is shown in Algorithm 1:

Algorithm 1 Codebook background modeling algorithm

**Input:** A three-channel video frames, $\epsilon_1, a, \beta, T_M$

**Output:** a background model

1. Initialize $L = 0$, set $I$ to empty set
2. for $t = 1$ to $N$ do
3. $\chi_t = (R, G, B)$, $I = \sqrt{R^2 + G^2 + B^2}$
4. for each $c_m \in I$ do
5. if $\text{colordist}(x_i, v_m) \leq \epsilon_1$ and $\text{brightness}(I, \langle \hat{I}_m, \bar{I}_m \rangle) = \text{true}$ then
6. $v_m = (\frac{f_m R + R}{f_m + 1}, \frac{f_m G + G}{f_m + 1}, \frac{f_m B + B}{f_m + 1})$
7. $\text{aux}_m = < \min\{I, \bar{I}_m\}, \max\{I, \hat{I}_m\}, f_m + 1, \max\{\lambda_m, t - q_m\}, p_m, t >$
8. else
9. $L = L + 1$
10. create a new $c_L$ by setting
11. $v_L = (R, G, B)$
12. $\text{aux}_L = < I, I, 1, t - 1, t, t >$
13. end if
14. end for
15. end for
16. for each $i = 1$ to $L$ do
17. $\lambda_i = \max\{\lambda_i, (N - q_i + p_i - 1)\}$
18. end for

$\epsilon_1$ in the algorithm is a manually given threshold, and the two judgment conditions represent the color distortion and brightness limits, respectively. For the input pixel $x_i$ and codeword $c_i$, there are:
\[
\begin{align*}
\|\mathbf{x}\|^2 &= R^2 + G^2 + B^2, \\
\|\mathbf{v}\|^2 &= R_i^2 + G_i^2 + B_i^2, \\
\langle \mathbf{x}, \mathbf{v} \rangle^2 &= (R_i R + G_i G + B_i B)^2
\end{align*}
\]

The color distortion can be calculated as:

\[
\begin{align*}
\rho_2^2 &= \|\mathbf{x}\|^2 \cos^2 \theta = \langle \mathbf{x}, \mathbf{v} \rangle^2 / \|\mathbf{v}\|^2, \\
\text{colordist}(\mathbf{x}, \mathbf{v}) &= \sqrt{\|\mathbf{x}\|^2 - \rho_2^2}
\end{align*}
\]

In order to allow the change of brightness during detection, the upper and lower limits of brightness have been given:

\[
\begin{align*}
I_{\text{low}} &= \alpha \hat{I}, \\
I_{\text{h}} &= \min \{ \beta \hat{I}, \frac{\hat{I}}{2} \}
\end{align*}
\]

where \( \alpha \) is usually between 0.4 and 0.7, and \( \beta \) is usually between 1.1 and 1.5. Then:

\[
\text{brightness}(I, \langle \hat{I}, \hat{I} \rangle) = \begin{cases} 
\text{true}, & \text{if } I_{\text{low}} \leq \|\mathbf{x}\| \leq I_{\text{h}}, \\
\text{false}, & \text{otherwise}
\end{cases}
\]

In the background update stage, according to the maximum interval time to remove the code words that have not appeared for a long time, the improved background model \( M \) is obtained:

\[
M = \{ \mathbf{c}_m \mid \mathbf{c}_m \in l \cap \lambda_m \leq T_M \}
\]

where \( T_M \) is the threshold, set to half the number of training frames, that is \( N/2 \).

- Foreground detection of original codebook algorithm

When the background model in the video scene is obtained, the codebook foreground detection algorithm can perform background subtraction \( BGS(x) \) operation on the new pixel input in the test set, which is fast and intuitive, as shown in Algorithm 2.

**Algorithm 2 Codebook foreground detection algorithm**

**Input:** pixel containing (R, G, B) components, \( \varepsilon_t, M \)  

**Output:** type of pixel

1. \( \mathbf{x} = (R, G, B), \hat{I} = \sqrt{R^2 + G^2 + B^2} \)
2. for each \( \mathbf{c}_m \in M \) do
3. \hspace{1em} if \( \text{colordist}(\mathbf{x}, \mathbf{v}_m) \leq \varepsilon_t \) and \( \text{brightness}(I, \langle \hat{I}_m, \hat{I}_m \rangle) = \text{true} \) then
4. \hspace{2em} \( \mathbf{v}_m = (\frac{f_m R_m + R}{f_m + 1}, \frac{f_m G_m + G}{f_m + 1}, \frac{f_m B_m + B}{f_m + 1}) \)
5. \hspace{2em} aux_m = \min \{I, \hat{I}_m\}, \max \{I, \hat{I}_m\}, f_m + 1, \max \{\lambda_m, t - q_m\}, p_m, t >
6. \hspace{2em} return false(background)
7. \hspace{1em} else
8. \hspace{2em} return true(foreground)
9. \hspace{1em} end if
10. end for

\( \varepsilon_t \) is the threshold for detection; if the pixel is not matched to the code word in the background model, the pixel is defined as foreground; otherwise, it is background.

Although the detection speed of the original codebook algorithm is fast, color information is taken into account, and the upper and lower limits of light intensity are set to reduce the interference caused by light changes, the experimental results show that the effect is average. As shown in Figure 4, the highway video frame sequence in the baseline category
in the CDNet2014 dataset [36] is used to evaluate the target detection effect of the original codebook algorithm.

**Figure 4.** Comparison of the detection results of the original codebook algorithm: (a) Color image; (b) True value diagram; (c) Detection result diagram of the original codebook algorithm.

It can be seen from Figure 4 that the original codebook algorithm can obtain the detection results quickly and can basically detect the approximate contour of the moving object. However, due to factors such as the leaves on the roadside swaying with the wind and changes in light intensity in the video, the original algorithm shows a large number of false detections at the trees beside the road on the upper left, and there are many missed detections inside the car. In addition, the shadows of the cars have the same trajectory as the cars and lead to a certain degree of adhesion between the shadows and the cars in the detection results, and more noise points appear elsewhere.

Considering that the shooting of the depth image is independent of brightness and color, this paper adds depth information to the codeword to expand the dimension of the codeword, and it optimizes the output result graph to reduce noise. Therefore, this paper proposes a codebook algorithm based on RGB-D to improve the performance and robustness of the algorithm. The algorithm includes background modeling and foreground detection.

- **Codebook moving target detection algorithm based on RGB-D**

The algorithm proposed in this paper adds one-dimensional depth information into the RGB component $v_i$ of the original codeword to convert into an RGBD vector, namely $v_i = (R_i, G_i, B_i, D_i)$. Then, we add the minimum and maximum depth values $\hat{D}_i$ and $\hat{D}_i$ of all the pixels belonging to codeword $c_i$ to the original six-dimensional tuple to make $\text{aux}_i$, an eight-dimensional tuple, namely $\text{aux}_i = < I_i, \hat{I}_i, \hat{D}_i, \hat{D}_i, f_i, \lambda_i, p_i, q_i >$. In addition, based on the two judgment conditions of background modeling and foreground detection algorithm, the depth deviation limit is added, and its definition refers to the light intensity limit, that is:

$$
\text{depth}(D, < \hat{D}, \hat{D} >) = \begin{cases} 
\text{true}, & \text{if } D_{\text{low}} \leq D \leq D_{\text{h}} \cup D = 0, \\
\text{false}, & \text{otherwise}
\end{cases}
$$

(12)

where

$$
\begin{align*}
D_{\text{low}} &= \alpha_d \hat{D}_i, \\
D_{\text{h}} &= \min\{\beta_d \hat{D}_i, \frac{D}{\alpha_d}\}
\end{align*}
$$

(13)

where $\alpha_d$ is usually between 0.4 and 0.7, $\beta_d$ is usually between 1.1 and 1.5. If the boundary range is too large, some foreground points will not be detected because the depth value changes little; if the range is too small, due to the quality problem of the depth map of the data set, the noise points whose depth values change are mistakenly detected as moving objects. Adding $D = 0$ to the formula takes into account that the depth of the moving object may be empty. In the judgment, the algorithm proposed in this paper not only takes the intersection after comparing color and depth changes separately but also fuses the two in order to reduce the interference of shadow and light intensity changes with
color changes as the dominant one. The judgment conditions are set as \( \text{color}(x_t, v_i) \) and \( \text{bright}(I, D, < I, \hat{I}, \hat{D}, \hat{D}>) \) as follows:

\[
\text{color}(x_t, v_i) = \begin{cases} 
\text{true}, & \text{if } \text{colordist}(x_t, v_i) \leq \varepsilon_1 \cup (\varepsilon_1 < \text{colordist}(x_t, v_i) \leq \varepsilon_2 \cap \text{depth}(D, < \hat{D}, \hat{D} >)) \\
\text{false}, & \text{otherwise}
\end{cases}
\]

\[
\text{bright}(I, D, < I, \hat{I}, \hat{D}, \hat{D}>) = \begin{cases} 
\text{true}, & \text{if } \text{brightness}(I, < I, \hat{I} >) \cup (\text{brightness}_2(I, < I, \hat{I} >) \cap \text{depth}(D, < \hat{D}, \hat{D} >)) \\
\text{false}, & \text{otherwise}
\end{cases}
\]

where \( \varepsilon_2 = k\varepsilon_1, k > 1 \) is another threshold value greater than \( \varepsilon_1 \), \( \text{brightness}_2(I, < I, \hat{I} >) \) is the value when \( \text{brightness}(I, < I, \hat{I} >) \) takes another set of \( \alpha_2 \) and \( \beta_2 \), making the upper and lower limits larger. The movement of shadow and the change of illumination will inevitably change the color and light intensity of the corresponding pixel, but the depth value of the pixel is not affected. Therefore, these two equations show whether the change of depth value exceeds the threshold as an auxiliary. Although the color and brightness of the pixel change, the depth change is still within the set range, which largely excludes the possibility that the pixel is a moving object.

\( D_{i}, \hat{D}_{i}, \) and \( \hat{D}_{i} \) is updated in the same way as \( R_{i}, I_{i}, \) and \( I_{i} \), where three conditions are required to determine whether a pixel \( x_t \) has a matching code \( c_m \), namely:

\[
\begin{align*}
\text{color}(x_t, v_m) = & \text{true} \\
\text{bright}(I_t, D_t, < I_m, \hat{I}_m, \hat{D}_m, \hat{D}_m >) = & \text{true} \\
\text{depth}(D_t, < \hat{D}_m, \hat{D}_m >) = & \text{true}
\end{align*}
\]

It can be seen from the detection results of Figure 4 that there are some noise points in the original codebook algorithm in addition to the detected foreground objects. Therefore, this paper uses morphological methods to remove the redundant noise points. Firstly, the erosion operation in the basic morphological operation of the binary image is used to remove the generated noise points. The erosion operation ablates the boundaries of objects according to the size of the structural elements, so that objects smaller than the structural elements disappear, and therefore, all noise points that do not fully contain structural elements can be removed. However, the disadvantage of erosion operation is that it will also change the shape of the foreground object to a certain extent. As shown in Figure 5b after erosion, although the sporadic noise spots are completely eliminated, the detected hand part is also eroded and thinned. Therefore, the morphological reconstruction is used to restore the image. The reconstruction involves two images, one of which is used as a marker and the other as a mask. The reconstruction method can accurately recover the image before the target area is eroded and achieve the purpose of removing the noise without changing the detection result of the foreground object. The results are shown in Figure 5c. In this paper, the morphological erosion and morphological reconstruction functions of MATLAB are used.
Figure 5. The detection result of the algorithm proposed in this paper: (a) Initial image of algorithm detection; (b) Morphological erosion image; (c) Morphological reconstruction image.

4. Results and Discussion

4.1. Depth Map Super-Resolution Reconstruction Based on Color Edge Guidance

In order to evaluate the effectiveness of the proposed network model framework for super-resolution reconstruction of depth maps based on color edge guidance, this paper first used the New Tsukuba Dataset [37] to train this network and then used the trained network to reconstruct the images in the Middlebury 2005 dataset [38] for qualitative and quantitative analysis. The New Tsukuba dataset is a one-minute video sequence with a frame sequence resolution of 640 × 480, including 1800 stereo matching pairs. It contains a color map, real disparity map, occlusion map and discontinuous map. Since this dataset is a scene generated by complete computer graphics (CG), the position of each coordinate can be determined, so a 100% correct depth map can be obtained. The Middlebury dataset is one of the datasets widely used in the field of depth map super-resolution reconstruction. In addition, this paper also made a comparative analysis with several other depth map super-resolution reconstruction methods, including the traditional bicubic, bilinear interpolation, reconstruction method based on second-order TGV [16], method based on nonlocal mean filtering [39] and SRCNN [5].

4.1.1. Experimental Setup

The purpose of training the network is to make the network gradually learn the mapping relationship between low-resolution and high-resolution depth maps, namely, learning parameter \( \theta = \{W_i, b_i, a_k\}, i = 1, 2, 3; k = 1, 2, 3 \). The data set used for training was denoted as \( \{D_{L_1}, I_{bwi}, D_{H_1}\} \), and 32 × 32 sub-images were cut by sliding step 24 on the low-resolution depth map \( D_L \) after bicubic up-sampling and the optimized edge map \( I_{bwi} \), respectively. Since the image was not filled, the final output image size of the CNN was 20 × 20, so the 20 × 20 sub-images were intercepted at the corresponding position of \( D_H \) as the label value for comparison in the loss function. In this paper, the mean square error (MSE) function was used as the loss function of the network, and its expression is as follows:

\[
L(\theta) = \frac{1}{n} \sum_{i=1}^{n} ||F(D_{L_i}, I_{bwi}; \theta) - D_{H_i}||^2
\]  

(17)

where \( n \) represents the number of training sample sets. The use of MSE as a loss function increases the signal-to-noise ratio (PSNR) of the output, which is a commonly used objective evaluation metric for image quality. The loss function is minimized by combining the stochastic gradient descent method and the error backpropagation. The update of the weight coefficient can be expressed as:

\[
\begin{align*}
\Delta_{i+1} &= 0.9\Delta_i - \eta \frac{\partial L}{\partial W_i} \\
W_{i+1} &= W_i + \Delta_{i+1}
\end{align*}
\]  

(18)

where \( l \in \{1, 2, 3, 4\} \) represents the \( l \)th bias coefficient, \( i \) represents the number of iterations, and \( \eta \) is the learning rate. The network parameter settings were as follows: \( c = 1, n_1 = n_2 = 64, n_3 = 64, f_1 = f_2 = 9, f_4 = 5 \), the learning rate of the first three convolution layers
was 0.001, and the learning rate of the fourth convolution layer was 0.0001. The weight was randomly initialized by Gaussian distribution with expectation of 0 and variance of 0.01, and the biases are all initialized to 0.

In this paper, the CNN was designed and implemented based on Caffe and MATLAB. In total, 111,150 sets of sub-images obtained from 225 sets of images in New Tsukuba Dataset were selected as the training set, and 37050 sets of sub-images obtained from 75 sets of images were selected as the test set. The low-resolution images used for training and testing were replaced by the sub-images by double-cubic down-sampling according to two and four times. The CPU of the training platform was Intel Core i5-4200H, and the GPU was Nvidia GeForce GTX 950M, 8GB memory.

4.1.2. Qualitative Analysis

This section provides a qualitative evaluation of the experimental results. Art, Books and Moebius in the data set compare the specific details of the four times up-sampling by the above different methods, as shown in Figure 6. It can be seen from the figure that the algorithm proposed in this paper has achieved the best results in the comparison of several methods. Compared with the traditional bicubic interpolation method or the reconstruction method based on second-order TGV, the obtained image is more clear and can retain more fine structural information in the image without blurring or introducing noise. Compared with the single-image SRCNN reconstruction method, RGB edge information is added to the training network as auxiliary information, which makes the reconstructed image edge more sharp and smooth.

![Figure 6. Comparison of the results of up-sampling depth maps with several methods: (a) Ground Truth; (b) Bicubic; (c) TVG; (d) SRCNN; (e) Method proposed in this article.](image)

4.1.3. Quantitative Analysis

In order to quantitatively analyze the experiment in this paper, the super-resolution reconstruction of six images in the Middlebury 2005 dataset was carried out by using the proposed algorithm and other comparison algorithms when the up-sampling coefficients were two and four in this section. The root mean square error (RMSE) was used as the evaluation index, and its expression is as follows:

\[
\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}
\]
where $y_i$ represents the pixel value (the depth value) on the depth ground truth map, $\hat{y}_i$ represents the pixel value on the result map obtained by the algorithm after sampling the low-resolution depth map, and $n$ represents the total number of pixels in the depth image. The RMSE index can well reflect the difference between the image reconstruction results and the real results.

Tables 1 and 2 show the experimental results when the samples were up-sampling to two and four. Park’s results were from Reference [40], while other results were obtained by using the source code provided by the authors. The best reconstruction results in the table have been boldly displayed. It can be seen from the table that the proposed algorithm obtains the best results in the up-sampling of two and four times of the depth map. On the one hand, compared with the traditional machine learning method, the convolution-based neural network can achieve better results even than SRCNN or the shallow neural network proposed in this paper, mainly because the convolution layer obtains abundant features, which plays an important role in depth map reconstruction. On the other hand, compared with the SRCNN method based on single image reconstruction, the proposed method adds another input, namely, the color image edge information of the same scene, which is conducive to maintaining the sharpness and smoothness of the edge in the reconstructed image and achieves better experimental results.

Table 1. Comparison of RMSE results of several methods (Art/Books/Moebius).

| Method     | Art $2\times$ | Books $2\times$ | Moebius $2\times$ | Art $4\times$ | Books $4\times$ | Moebius $4\times$ |
|------------|---------------|-----------------|-------------------|---------------|-----------------|-------------------|
| Bicubic    | 2.630         | 3.872           | 1.045             | 1.604         | 0.870           | 1.329             |
| Bilinear   | 2.875         | 4.150           | 1.132             | 1.698         | 0.955           | 1.431             |
| Park et al.| 2.833         | 3.498           | 1.088             | 1.530         | 1.064           | 1.349             |
| Ferstl et al.| 3.164         | 3.761           | 1.370             | 1.770         | 1.153           | 1.497             |
| SRCNN      | 1.924         | 2.651           | 0.793             | 1.150         | 0.726           | 1.049             |
| The proposed model | **1.051** | **2.222** | **0.437** | **0.873** | **0.505** | **0.905** |

Table 2. Comparison of RMSE results of several methods (Dolls/Laundry/Reindeer).

| Method     | Dolls $2\times$ | Laundry $2\times$ | Reindeer $2\times$ | Dolls $4\times$ | Laundry $4\times$ | Reindeer $4\times$ |
|------------|-----------------|-------------------|--------------------|-----------------|-------------------|--------------------|
| Bicubic    | 0.910           | 1.309             | 1.601              | 2.396           | 1.928             | 2.814              |
| Bilinear   | 0.987           | 1.400             | 1.738              | 2.548           | 2.096             | 3.014              |
| Park et al.| 0.963           | 1.301             | 1.552              | 2.132           | 1.834             | 2.407              |
| Ferstl et al.| 1.164         | 1.435             | 1.891              | 2.681           | 2.546             | 3.165              |
| SRCNN      | 0.8             | 1.124             | 1.095              | 1.742           | 1.378             | 2.013              |
| The proposed model | **0.679** | **1.063** | **0.663** | **1.321** | **0.803** | **1.687** |

4.2. Codebook Moving Target Detection Algorithm Based on RGB-D

In this paper, the SBM-RGBD dataset [41] was used to evaluate and compare the background modeling methods of moving object detection scenes in RGBD videos. The dataset provides a set of diversified true color and depth sequences obtained by Microsoft Kinect. It consists of 33 videos (about 15,000 frames), which represent typical indoor visual data captured in video surveillance and intelligent environment scenes. The resolution of the video is 640 × 480, and the depth map is 16 or 8 bits. The length of the video varies from 70 to 1400 frames. In order to verify the effectiveness of the proposed algorithm in this paper, we compare it with the original codebook algorithm in the following three scenarios set up: (1) the target is close to the background; (2) the target is similar to the background color; (3) illumination changes. Three RGB-D based moving object detection methods proposed in the last 5 years are also selected for comparison in these three scenarios, which
are SRPCA [42], CwisarDH+ [43], and BSABU [44]. The results of these three methods are
directly obtained from the results submitted by their authors on the SBM-RGBD dataset
website. The experiment was divided into qualitative and quantitative analysis. The
algorithm framework was built on MATLAB, and the CPU of the operating environment
was Intel Core i5-4200H, 8G memory. The experimental parameters were set as shown in
Table 3.

Table 3. Experimental parameters settings of moving target detection algorithm.

| Parameters | Values | Parameters | Values | Parameters | Values |
|------------|--------|------------|--------|------------|--------|
| $\alpha$   | 0.75   | $\beta$    | 1.3    | $\epsilon$ | 5      |
| $\alpha_d$ | 0.97   | $\beta_d$  | 1.05   | $\epsilon_t$ | 25     |
| $\alpha_2$ | 0.45   | $\beta_2$  | 1.4    | $k$        | 8      |

The evaluation indexes used in quantitative analysis were accuracy $P$, recall $R$ and
index $F$. The specific formulas are as follows:

$$\begin{align*}
  P &= \frac{TP}{TP + FP} \\
  R &= \frac{TP}{TP + FN} \\
  F &= 2 \times \frac{P \times R}{P + R}
\end{align*}$$

(20)

where $TP$ (True Positive) represents the number of pixels that are correctly detected as
moving objects, $FP$ (False Positive) represents the number of pixels that are falsely detected
as moving objects, and $FN$ (False Negative) represents the number of pixels that are falsely
detected as background. Index $F$ is the combination of $P$ and $R$, and the larger the value,
the better the performance of the algorithm.

4.2.1. Qualitative Analysis

- **Experimental Scenario One: the target is close to the background**

  The content of the test frame sequence “Wall” in the data set is that the book was
  gradually close to and away from the lens by hand. In addition, due to the shooting light, the
  wall covered by the book had a large number of shadows. In this section, we selected the 74th
  frame, which was the nearest to the wall, the 94th frame, which was the nearest to the shot,
  and the 134th frame, which was the deepest shadow in the middle of the frame sequence. The
color images are shown in Figure 7, and the corresponding depth images of these three frames
are shown in Figure 8. From the depth map, it can be seen that the area of depth change in the
scene corresponds to the hand and the book that should be detected, which also shows that
the depth information is of great value in detecting objects to some extent.

![Figure 7](image_url)

**Figure 7.** Depth sampling map of Experimental Scenario One: (a) the 74th frame; (b) the 94th frame;
(c) the 134th frame.
The detection results are shown in Figure 9. It can be seen from the results that when the target is close to the background, the original codebook algorithm is prone to missed detection, and the detection of the hand is also prone to false detection. When the shadow of the object is deep, due to the obvious difference in light intensity and brightness between the appearance of the shadow and the background, the original algorithm is easy to misjudge the shadow as the foreground, leading to the adhesion between the moving object and the shadow. In addition, there are also some noises in the figure, and the subjective effect is poor. The detection results of the SRPCA method do not have many noise points, but some background areas are detected as the object, and the contours differ significantly from the ground truth. The contours of the CwisarDH+ method detection results are close to the truth, but large hole areas appear inside the object. The BSABU method has excellent detection results in the 94th and 134th frames, but when the object is closest to the background in the 74th frame, the object is almost not detected. The detection segmentation effect of the proposed algorithm is clear and free of holes and noise, which is significantly better than the original algorithm and the other three methods.
Figure 9. Comparison of the detection results of the two algorithms in Experimental Scenario One: (a) Ground truth; (b) The detection result of the original codebook algorithm; (c) The detection result of the original codebook algorithm with noise points removed using morphological operations; (d) The detection result of the algorithm proposed in this article; (e) The detection result of the SRPCA method; (f) The detection result of the CwisarDH$^+$ method; (g) The detection result of the BSABU method.

- Experimental Scenario Two: the target is similar to the background color

According to experience, when the moving target is similar to the background color, it is easy to lead to missed or false detection of target pixels. Therefore, this paper selected the “Hallway” frame sequence in the data set to compare the effectiveness of the two algorithms. The man in the video walked with the same box as the white wall color, and the pants color was similar to the barrel color. In the experiment, the 258th frame was selected. The sampling images of the experiment are shown in Figure 10, and the detection results are shown in Figure 11.
Figure 10. Sampling map of Experimental Scenario Two: (a) Background image; (b) Color image of the 258th frame; (c) Depth image of the 258th frame.

Figure 11. Comparison of the detection results of the two algorithms in Experimental Scenario Two: (a) Ground truth; (b) The detection result of the original codebook algorithm; (c) The detection result of the original codebook algorithm with noise points removed using morphological operations; (d) The detection result of the algorithm proposed in this article; (e) The detection result of the SRPCA method; (f) The detection result of the CwisarDH+ method; (g) The detection result of the BSABU method.

It can be seen from the test results that due to the similar color between the wall and the box, and between the pants and the bucket, the original algorithm has many false detections and objects are not coherent. In addition to some details such as the gap between the arms, the algorithm proposed in this paper maintains a relatively complete shape of the box and the person. The detection result of the SRPCA method is incomplete in the legs of the characters with more holes, while the detection results of both the CwisarDH+ method and the BSABU method in this scenario are closer to the ground truth, and the gaps between the person and the box can be detected better. The result of BSABU is slightly better than the algorithm proposed in this paper, which is the best result among these methods.

• Experimental Scenario Three: sudden illumination changes

When using the background modeling algorithm, it often occurs after the background model is established in the training process, and the illumination conditions of the test frame change, resulting in many false detections of the test results. In this paper, we select the “Shelves” frame sequence from the test set, in which a man enters a room and turns on the light, causing a sudden change in illumination. In the experiment, the 390th frame was selected. The sampling images of the experiment are shown in Figure 12, and the truth map and the detection results of the two algorithms are shown in Figure 13.
Figure 12. Sampling map of Experimental Scenario Three: (a) Background image; (b) Color image of the 390th frame; (c) Depth image of the 390th frame.

Figure 13. Comparison of the detection results of the two algorithms in Experimental Scenario Three: (a) Ground truth; (b) The detection result of the original codebook algorithm; (c) The detection result of the original codebook algorithm with noise points removed using morphological operations; (d) The detection result of the algorithm proposed in this article; (e) The detection result of the SRPCA method; (f) The detection result of the CwisarDH$^+$ method; (g) The detection result of the BSABU method.

It can be seen from the test results that due to the difference between the illumination brightness in the detection and the background modeling, in the original algorithm, a large number of pixels are mistaken for the foreground due to the brightness change. The algorithm proposed in this paper is able to detect the foreground more completely due to the addition of depth information as an auxiliary judgment, and the detection is greatly improved because the depth is not affected by the change of illumination. The detection result of the SRPCA method is much different from the ground truth, and there are more false detections. Although the detection results of both the CwisarDH$^+$ and the BSABU methods are closer to the ground truth and can clearly and completely detect the human form, the detection results have spikes on the edges and generate a little noise on the left side of the figure compared to the algorithm proposed in this paper. Therefore, the algorithm in this paper subjectively achieves the best results in this scenario.

4.2.2. Quantitative Analysis

Table 4 lists the value of the quantitative index $F$ used to evaluate the detection effect of the several methods in the above three scenarios, and the best result for each experiment has been shown in bold. From the data in the table, it can be seen that the algorithm proposed in this paper achieves the best results in both the “Wall” and the “Shelves” experiments, and its F-value is only 2.27% lower than the best result in the “Hallway” experiment, although
it does not achieve the best result. Although the result of the CwisarDH+ and the BSABU methods in the “Hallway” experiment is slightly better than the proposed algorithm, the result of the CwisarDH+ method in the “Wall” experiment at the 74th frame is only 0.2736, which is much lower than the 0.9916 of the proposed algorithm, and the result of the BSABU method in the “Wall” experiment at the 94th and 134th frames is only 0.5737 and 0.4932, which is much lower than the 0.9953 and 0.9889 of the proposed algorithm. Therefore, combining all the experimental results, the algorithm proposed in this paper achieves the best result among all the methods compared, and it has significantly improved compared with the original algorithm, and the detection results in the experiments of each scenario are excellent.

Table 4. Comparison of the F-values of the proposed method with several RGB-D based methods in recent years for 3 scenarios. The results with the best performance in each scenario have been bolded.

|                      | 74th Frame | 94th Frame | 134th Frame | 258th Frame | 390th Frame |
|----------------------|------------|------------|-------------|-------------|-------------|
| SRPCA [42]           | 0.8192     | 0.7846     | 0.8433      | 0.7835      | 0.7598      |
| CwisarDH+ [43]       | 0.8966     | 0.5737     | 0.4932      | 0.9736      | 0.9945      |
| BSABU [44]           | 0.2736     | 0.9892     | 0.9770      | 0.9843      | 0.9912      |
| Original algorithm   | 0.9182     | 0.8642     | 0.8367      | 0.8756      | 0.7102      |
| Original algorithm after morphological operations | 0.9374 | 0.8704 | 0.8636 | 0.9199 | 0.7748 |
| Proposed algorithm   | **0.9916** | **0.9953** | **0.9889** | **0.9620** | **0.9960** |

The codebook algorithm requires only a small amount of memory to save the model and has a small amount of computation, so it can run for a long time and the detection speed is also fast. In this paper, the original codebook algorithm is improved by adding depth information, and the stability of depth information largely improves the defect that the original algorithm has difficulty in accurately detecting foreground targets when shadow and illumination changes affect it. Finally, the detection experiments were carried out using the proposed algorithm in three scenes: dark shadows, similar color of objects and background, and illumination changes. The detection results are compared with the other three RGB-D based methods, which further qualitatively and quantitatively verify the effectiveness of the improved algorithm.

5. Conclusions

In this paper, we propose a color edge-guided depth map super-resolution convolutional neural network framework and RGB-D-based codebook moving object detection algorithm. Our work improves the super-resolution reconstruction and moving object detection, and the validity of the model framework and algorithm is verified on different datasets. Firstly, according to the corresponding relationship between the color map and the depth map, the low-resolution depth map and the high-resolution optimized color edge map of the same scene are simultaneously input to the network, with the aim of using the color edge map to guide the reconstruction of the depth map. The reconstructed results are then compared with several traditional super-resolution algorithms, and the results show that the proposed model framework effectively improves the quality of the depth map after super-resolution. Secondly, we add the depth information of the scene as an auxiliary dimension based on the original codebook algorithm to better segment moving objects by using the property that depth information is almost immune to shadows, light changes, and color interference. Finally, the post-processing of morphological erosion and morphological reconstruction is used to reduce noise and optimize the detection results. We compare the proposed algorithm with three other recent RGB-D based methods, and the results show that the proposed algorithm significantly improves the effectiveness of the original algorithm in detecting objects in three scenarios, and it achieves the best results overall among all methods, proving the effectiveness of the proposed method in this paper.
In future research, we consider adding deconvolution layers to replace the up-sampling in pre-processing, increasing the number of network layers, increasing the nonlinearity, and improving the network fitting ability to make the reconstructed accuracy higher. In addition, from the aspect of moving object detection, the proposed algorithm sets the threshold, boundary constant and other hyperparameters to a fixed value, which is difficult to guarantee good detection results in all scenarios, and it may cause problems such as large amounts of noise and holes in detection. Therefore, the subsequent research considers changing these constants to adaptive to expand the applicability of the algorithm in different scenes.

**Author Contributions:** Conceptualization, X.B. and S.Y.; methodology, X.B. and S.Y.; software, S.Y.; validation, X.B., S.Y. and P.T.; formal analysis, X.B. and P.T.; investigation, X.B. and S.Y.; resources, X.B.; data curation, S.Y.; writing—original draft preparation, S.Y.; writing—review and editing, X.B. and P.T.; supervision, X.B. and P.T.; funding acquisition, X.B. and P.T. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Key R&D Program of China (No. 2018AAA0102801) and the Major Science and Technology Innovation Project of the Chengdu Science and Technology Bureau, China (No. 2021-YF08-00140-GX).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** CDNet2014, New Tsukuba, Middlebury 2005, SBM-RGBD datasets are available in the public domain.

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**

1. Zuo, J.; Jia, Z.; Yang, J.; Kasabov, N. Moving target detection based on improved Gaussian mixture background subtraction in video images. *IEEE Access.* 2019, 7, 152612–152623.
2. Yazdi, M.; Bouwmans, T. New trends on moving object detection in video images captured by a moving camera: A survey. *Comput. Sci. Rev.* 2018, 28, 157–177.
3. Kim, K.; Chalidabhongse, T.H.; Harwood, D.; Davis, L. Real-time foreground–background segmentation using codebook model. *Real-Time Imaging* 2005, 11, 172–185.
4. Freeman, W.T.; Jones, T.R.; Pasztor, E.C. Example-based super-resolution. *IEEE Comput. Graph. Appl.* 2002, 22, 56–65. [CrossRef]
5. Dong, C.; Loy, C.C.; He, K.; Tang, X. Learning a deep convolutional network for image super-resolution. In Proceedings of the 13th European Conference on Computer Vision (ECCV2014), Zurich, Switzerland, 6–12 September 2014; pp. 184–199. [CrossRef]
6. Mandal, S.; Bhavsar, A.; Sao, A.K. Depth map restoration from undersampled data. *IEEE Trans. Image Process.* 2016, 26, 119–134. [CrossRef]
7. Lei, J.; Li, L.; Yue, H.; Wu, F.; Ling, N. Depth map super-resolution considering view synthesis quality. *IEEE Trans. Image Process.* 2017, 26, 1732–1745. [CrossRef]
8. Anwar, S.; Khan, S.; Barnes, N. A deep journey into super-resolution: A survey. *ACM Comput. Surv. (CSUR)* 2020, 53, 1–34. [CrossRef]
9. Kim, J.; Lee, J.K.; Lee, K.M. Accurate image super-resolution using very deep convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 1646–1654. [CrossRef]
10. Ahn, N.; Kang, B.; Sohn, K.A. Fast, accurate, and lightweight super-resolution with cascading residual network. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 252–268. [CrossRef]
11. Li, Z.; Yang, J.; Liu, Z.; Yang, X.; Jeon, G.; Wu, W. Feedback network for image super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 3867–3876. [CrossRef]
12. Niu, B.; Wen, W.; Ren, W.; Zhang, X.; Yang, L.; Wang, S.; Zhang, K.; Cao, X.; Shen, H. (2020, August). Single image super-resolution via a holistic attention network. In European conference on computer vision (ECCV), Glasgow, UK, 23–28 August 2020; pp. 191–207. [CrossRef]
13. Rajagopalan, A.N.; Bhavsar, A.; Wallhoff, F.; Rigoll, G. Resolution enhancement of pmd range maps. In Proceedings of the Dagm Symposium on Pattern Recognition, Munich, Germany, 10–13 June 2008; pp. 304–313.
14. Wetzl, J.; Taubmann, O.; Haase, S.; Köhler, T.; Kraus, M. Hornegger, J. GPU-accelerated time-of-flight super-resolution for image-guided surgery. In Bildverarbeitung für die Medizin, Meinner, H.P., Deserno, T., Handels, H., Toksdorf, T., Eds.; Springer: Berlin, Heidelberg, 2013; pp. 21–26. [CrossRef] [PubMed]

15. Izadi, S.; Kim, D.; Hilliges, O.; Molyneaux, D.; Newcombe, R.; Kohli, P.; Fitzgibbon, A. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, Santa Barbara, CA, USA, 16–19 October 2011; pp. 559–568. [CrossRef] [PubMed]

16. Ferszl, D.; Reinbacher, C.; Ranftl, R.; Rüther, M.; Bischof, H. Image guided depth upsampling using anisotropic total generalized variation. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Sydney, Australia, 1–8 December 2013; pp. 993–1000. [CrossRef]

17. Zhang, T. Research on Moving Target Detection Technology Based on RGB-D Data. Master’s Thesis, Nanjing University of Science and Technology, Nanjing, China, January 2017.

18. Yang, J.; Ye, X.; Li, K.; Hou, C.; Wang, Y. Color-guided depth recovery from RGB-D data using an adaptive autoregressive model. IEEE Trans. Image Process. 2014, 23, 3443–3458.

19. Song, X. Research on Depth Image Enhancement Based on RGB-D Information. Doctoral Dissertations, Shan Dong University, Jinan, China, November 2017.

20. Wen, Y.; Sheng, B.; Li, P.; Lin, W.; Feng, D.D. Deep color guided coarse-to-fine convolutional network cascade for depth image super-resolution. IEEE Trans. Image Process. 2018, 28, 994–1006.

21. Xiong, Y. Moving object extraction based on background difference and frame difference method. Computer Era. 2014, 3, 38–41.

22. Weng, M.; Huang, G.; Da, X. A new interframe difference algorithm for moving target detection. In Proceedings of the 2010 3rd International Congress on Image and Signal Processing, Yantai, China, 16–18 October 2010; pp. 285–289.

23. Hu, J.; Li, Y.; Pan, X. An Improved Optical Flow Algorithm in Vehicle Identification and Tracking. Sci. Technol. Eng. 2010, 10, 5814–5817.

24. Zhang, Y. Research of Moving Target Detection Based on Optical Flow Algorithm. Master’s Thesis, Xi’an Shiyou University, Xi’an, China, June 2018.

25. Cui, P. Research on Moving Targets Detection and Tracking Algorithms in Complex Interference Scenes. Master’s Thesis, Jiangnan University, Wuxi, China, June 2021.

26. Qiu, S.; Li, X. Moving target extraction and background reconstruction algorithm. J. Ambient. Intell. Humaniz. Comput. 2020, 1–9. https://doi.org/10.1007/s12652-020-02619-2.

27. Sun, T.; Qi, Y.; Geng, M. Moving object detection algorithm based on frame difference and background subtraction. J. Jilin Univ. (Eng. Technol. Ed.) 2016, 4, 1325–1329.

28. Parvizi, E.; Wu, Q.J. Multiple object tracking based on adaptive depth segmentation. In Proceedings of the 2008 Canadian Conference on Computer and Robot Vision (CRV), Windsor, ON, Canada, 28–30 May 2008; pp. 273–277.

29. Ottonelli, S.; Spagnolo, P.; Mazzeo, P.L.; Leo, M. Improved video segmentation with color and depth using a stereo camera. In Proceedings of the 2013 IEEE International Conference on Industrial Technology (ICIT), Cape Town, South Africa, 25–28 February 2013; pp. 1134–1139. [CrossRef] [PubMed]

30. Sun, T.; Qi, Y.; Geng, M. Moving object detection algorithm based on frame difference and background subtraction. J. Ambient. Intell. Humaniz. Comput. 2020, 1–11. https://doi.org/10.1007/s12652-020-02619-2.

31. Hu, T.; Zhu, X.; Guo, W.; Zhang, X.; Yang, J. Moving Object Detection Based on the Fusion of Color and Depth Information. J. Electron. Inf. Technol. 2014, 9, 2047–2052.

32. Hu, T.; Zhu, X.; Guo, W.; Zhang, F. A Moving Object Detection Method Combining Color and Depth data. Geomat. Inf. Sci. Wuhan Univ. 2019, 44, 276–282. [CrossRef] [PubMed]

33. Li, D.; Guan, Z.W.; Chen, Q.; Shi, H.P.; Wang, T.; Yue, H.H. Research on Road Traffic Moving Target Detection Method Based on Sequential Inter Frame Difference and Optical Flow Method. In Artificial Intelligence in China; Liang, Q., Wang, W., Mu, J., Liu, X., Na, Z., Eds.; Springer: Singapore, 2022; pp. 376–383.

34. Liu, F.; Liu, P.; Zhang, J.; Xu, B. Joint Detection of RGB-D Images Based on Double Flow Convolutional Neural Network. Laser Optoelectron. Prog. 2017, 55, 386–394.

35. Ni, M. Research on the Key Technologies of RGB-D Image Processing Based on Deep Learning. Master’s Thesis, Tianjin University, Tianjin, China, December 2017.

36. He, K.; Sun, J.; Tang, X. Guided image filtering. IEEE Trans. Pattern Anal. Mach. Intell. 2012, 35, 1397–1409.

37. Goyette, N.; Jodoin, P.M.; Porikli, F.; Konrad, J.; Ishwar, P. Changeddetection. net: A new change detection benchmark dataset. In Proceedings of the 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Providence, RI, USA, 16–21 June 2012; pp. 1–8.

38. Martull, S.; Peris, M.; Fukui, K. Realistic CG stereo image dataset with ground truth disparity maps. IEICE Tech. Rep. Speech 2012, 111, 117–118.

39. Scharrstein, D.; Pal, C. Learning conditional random fields for stereo. In Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 17–22 June 2007; pp. 1–8. [CrossRef]

40. Park, J.; Kim, H.; Tai, Y.W.; Brown, M.S.; Kweon, I. High quality depth map upsampling for 3d-tof cameras. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 1623–1630.

41. Hui, T.W.; Loy, C.C.; Tang, X. Depth map super-resolution by deep multi-scale guidance. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016; pp. 353–369.
41. Camplani, M.; Maddalena, L.; Moyá Alcover, G.; Petrosino, A.; Salgado, L. A benchmarking framework for background subtraction in RGBD videos. In Proceedings of the International Conference on Image Analysis and Processing (ICIAP), Catania, Italy, 11–15 September 2017; pp. 219–229.

42. Javed, S.; Bouwmans, T.; Sultana, M.; Jung, S. K. Moving object detection on RGB-D videos using graph regularized spatiotemporal RPCA. In Proceedings of the International Conference on Image Analysis and Processing (ICIAP), Catania, Italy, 11–15 September 2017; pp. 230–241.

43. De Gregorio, M.; Giordano, M. CwisarDH+: Background Detection in RGBD Videos by Learning of Weightless Neural Networks. In Proceedings of the International Conference on Image Analysis and Processing (ICIAP), Catania, Italy, 11–15 September 2017; pp. 242–253.

44. Dorudian, N.; Lauria, S.; Swift, S. Moving object detection using adaptive blind update and RGB-D camera. *IEEE Sens. J.* 2019, 19, 8191–8201.