IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 2, 2019, accepted June 20, 2019, date of publication June 26, 2019, date of current version July 11, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2925101

Occlusion Detection of Dense Correspondence Fields Using Motion Statistics and Triangulation

CHONG DONG[1], **ZHISHENG WANG**[1], **CHANGDA XING**[1], **JIAMING HAN**[1], **AND PINGAN HUANG**[2]

[1]College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211100, China
[2]Avic Xian Aircraft Industry Company Ltd., Xi'an 710089, China

Corresponding author: Zhisheng Wang (zshengwang@126.com)

This work was supported in part by the Natural Science Foundation of China under Grant 61473144, and in part by the Aeronautical Science Foundation of China under Grant 20162852031.

ABSTRACT Despite progress made in the accuracy and robustness of the dense matching technique in past years, efficient occlusion detection remains an open problem. In this paper, we present a two-step occlusion detection method to remove false matches in dense matching fields. First, a statistical dense matching method is developed by considering the correspondence between the grids to identify most occlusion regions. Second, to handle the potential misjudgment match in the occlusion boundary, a double-threshold filtering method is first used to reduce the noise in the grid image, which ensures that the gradient operator can accurately extract the boundary grid in the grid image; then, misjudgment matches in the boundary grid region are corrected based on the triangulation with descriptors. The results of the experiments comparing the proposed method and existing occlusion detection methods by, respectively, using the MPI-Sintel and KITTI datasets' test sequence show that the proposed method has higher accuracy and better robustness.

INDEX TERMS Dense matching, occlusion detection, motion statistics, triangulation.

I. INTRODUCTION

Dense matching aims at determining the dense correspondence between two consecutive frames. It plays an important role in image processing and has been widely used in many vision tasks, including dense stereo reconstruction [1], optical flow estimation [2], object recognition [3], object tracking [4], and image retrieval [5]. Despite an abundance of literature related to this topic, obtaining a reliable dense correspondence field remains a challenging problem mainly because of illumination change, repetitive patterns, geometric deformation, and occlusions. This paper addresses the issue of removing false matches caused by occlusions in dense correspondence fields.

To obtain dense matching fields, the patch-based methods have been the most used approach. The goal of this approach is to find one or more nearest neighbor matches between image patches. A serious challenge in this approach is the computational complexity as the graph-based [6] method or the belief propagation [7] method become very slow for many applications when the image resolution is high or the database image is large. The PatchMatch algorithm [8] and its improved versions [9]–[11] have represented a powerful breakthrough in this field. It brings almost two orders of magnitude faster than the mentioned above approaches and has similar accuracy. The core idea behind this breakthrough is to transform this large search problem into random search methods that exploit the local coherence of image patches. Although the efficiency of computing the dense correspondence field has advanced remarkably, the computed nearest neighbor field is often very noisy which is mainly caused by the matching ambiguities and occlusions.

The research of PatchMatch-based matching algorithms mostly focused on reducing the ambiguity of image patches and computation complexity. They are typically introduced advanced data structures such as propagation-assisted KD-trees [11], CSH [10] and RIANN [12], or designed efficient search strategies like hierarchical architecture [13], [14],

The associate editor coordinating the review of this manuscript and approving it for publication was Dong Wang.

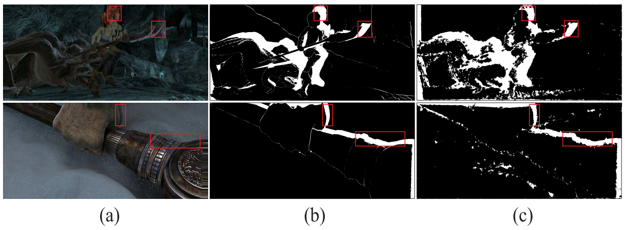


FIGURE 1. **Illustration of the false matches in the occlusion region. (a) and (b) are the means of two consecutive images and the ground truth of occlusions, respectively. (c) shows the occlusion detection results of the forward-backward consistency checking method. The regions surrounded by the red squares indicate several occluded regions. We can see that most false matches are located in the occluded region.**

to improve the efficiency of the algorithm. However, the occlusion detection of the matching field is an indispensable component of generating high-quality dense correspondence fields. Few studies have investigated methods of improving the efficiency of this step. Typically, the putative correspondence field contains a large number of false matches, in addition to a limited number of true matches. These false matches can be approximately divided into two parts, one is a small number of false matches due to the ambiguity of image patches, and the other is a large number of false matches due to occlusions. One of the most important reasons is that occlusions are often not explicitly defined in many matching algorithms, and then patches in the occluded region may randomly match that of other regions and even interfere with the correct correspondence in the neighborhood of occluded regions. Fig. 1 indicates the results of a forward-backward consistency checking method when processing dense correspondence fields with a large number of false matches. The results reveal that most false matches are obviously located in the occluded region compared to the ground truth of occlusions. In this context, most matching algorithms [14]–[22] depend on the forward-backward consistency check method to detect occlusions. If the backward and forward flow is constant, the pixel will be considered as non-occluded. However, this cross-checking method may not be reliable for some cases. For instance, the flow of mutually corresponding pixels in the occluded region may be zero, while it will be considered as true correspondences using cross-checking; another case exists in the low-texture region-multiple pixels may correspond to the same pixel-which means that there are many erroneous mutual correspondences and they are difficult to detect using cross-checking. Besides, this method must calculate the bidirectional flow, which is highly time-consuming for some advanced matching algorithms.

Aiming to solve those pre-mentioned problems, this paper combines the statistical matching constraint and triangulations of an image into a two-step occlusion detection method to remove false matches in dense matching fields. In our approach, first we propose a statistical matching framework in which the grid size is allowed to change, and a statistical dense matching method is used to identify most occlusion regions. Then, based on the observation of the boundary grid, we describe a triangulation-based misjudgment correction method to improve the accuracy of occlusion detection further. First, we propose a double-threshold filtering method to ensure the accuracy of the extraction of grid boundaries. Next, we use a descriptor-based triangulation to correct misjudgment matches in the extraction region. Although triangulation-based methods have been used in [23], our approach is different. The extraction region might follow the real occlusion boundaries more closely, which makes triangulation more accurate, and measurements based on the descriptor data can also improve the accuracy of the triangulation.

In summary, we make the following contributions: 1) We propose a two-step occlusion detection method (TSOD) that combines the statistical matching constraint and triangulations of an image. We show that TSOD is an effective approach to remove false matches that arise from occlusions in dense matching fields. 2) We propose a statistical dense matching method which extends the applicable range of the statistical matching model from robust correspondence field to dense correspondence field. Furthermore, misjudgment correction based on the triangulation with descriptors can achieve more accurate results by utilizing the extraction occlusion boundary and the measurement based on the descriptor data. 3) We experimentally show that the proposed method performs better than the forward and backward consistency check method on the challenging datasets MPI-Sintel and KITTI.

The remainder of this paper is organized as follows. Section II reviews related work. In Section III, we present a two-step occlusion detection method to remove false matches in dense matching fields. In Section IV, we present and compare results of numerical experiments conducted on test sequences of MPI-Sintel and KITTI datasets using the proposed method and other representative methods. Our conclusions are provided in Section V.

II. RELATED WORK

It is beyond the scope of this paper to review the entire literature on image matching. We refer to previous literature [24]–[26] for a detailed overview of image matching algorithms. Instead, we review the work most related to our method. In particular, we will focus on the work that addresses occlusions in the dense correspondence field.

One common approach to detect occlusions is to compare forward and backward motion estimates based on motion symmetry between image sequences. Some algorithms try to exploit this property to model occlusions in the dense stereo matching. Luo and Burkhardt [27] proposed using cooperative bidirectional matching to detect discontinuities and occlusions and incorporate them into the cost function. Sun *et al.* [28] proposed a model based on the symmetric visibility constraint to handle occlusions for more general scenes. Some algorithms propose to extended the consistency check method based on the characteristic of its algorithm.

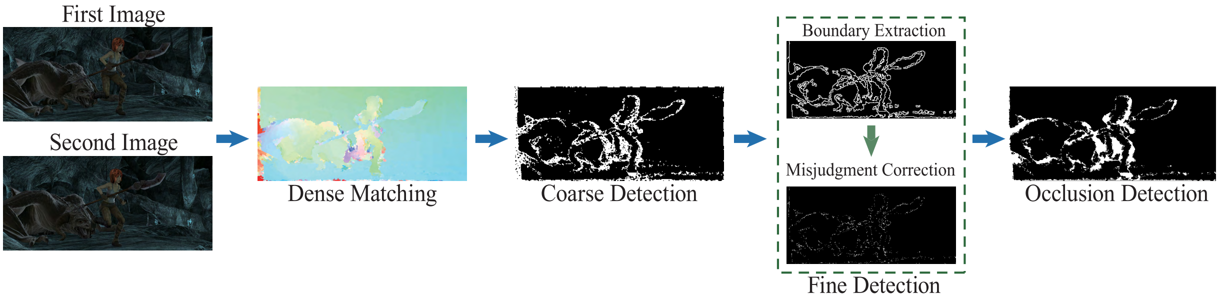


FIGURE 2. Overview of the proposed TSOD algorithm. Given two images, we compute dense matches using CPM [14]. We use the statistical dense matching method to detect most false matches (Coarse Detection) and correct misjudgment matches using a descriptor-based triangulation (Fine Detection).

Hu *et al.* [14] performed a forward-backward consistency check on each level of the pyramid based on the proposed hierarchical matching framework, which can help to detect occlusions and remove outliers. Bailer *et al.* [13] proposed to check the consistency of the forward flow via the backward flow with different patch radii, which enhances its robustness to some extent. In the optical flow community, this symmetry property [29]–[31] is also exploited to model occlusions to extrapolate flow fields in the occluded region. Some algorithms [32], [33] use the symmetry of image sequence to detect occlusion regions and inpaint these areas using surrounding intensities and gradients. Xiao *et al.* [34] proposed to use the squared image residue to enhance the robustness of the bidirectional consistency checking and explicitly introduce an occlusion term to balance the energy loss in the variational optical flow model. Although the enhanced version of the cross-checking method is effective in occlusion detection, as the analysis in Section I, it exists an inevitable problem that the motion in the occluded or the low-texture region is inconsistent.

Several algorithms do not rely on motion symmetry to detect occlusions. Zitnick and Kanade [35] proposed to identify occlusions by examining the magnitude of the converged match values in conjunction with the uniqueness constraint. Min and Sohn [36] proposed to determine candidate sets of occluded regions by geometric and photometric constraints and assign a reasonable cost function to detect occlusions further. Others infer occluded regions based on the unidirectional flow and low-level image data. Strecha *et al.* [37] proposed a probabilistic framework to estimate occlusions by using the histogram of occluded pixel intensities. Glocker *et al.* [38] proposed triangulations to exactly compute local affine warps and incorporating triangle motions into higher-order likelihood terms. Inspired by this work, Kennedy and Taylor [39] divide the image into discrete triangles to describe occlusions and use the motion information of the triangle to optimize the computation of the flow fields. Zhang *et al.* [23] proposed an improved numerical quadrature scheme that is used to handle the resulting occlusions effects via the comparison of data cost between quadrature points and triangles, but the accuracy of methods based on triangulation largely depends on the locating of the motion boundary.

Our work is related to GMS [40]. GMS has significantly improved the efficiency of removing false matches in sparse matching fields via matching statistics, but the results are often too noisy with regard to the dense correspondence field. We count the number of neighborhood matches for each grid-cell based on the correspondence between the grids, which is shown to improve the performance of the statistical matching constraint in dense correspondence fields. Our work is related to [23] in terms of using triangulation. We explicitly provide detection regions that may follow the real motion boundaries more closely than other methods [23], [38], [39], and we use the measurement based on the descriptor data as occlusion costs, which are shown to improve the performance of the triangulation in occlusion detection. In this paper, a two-step occlusion detection method combining the statistical matching constraint and triangulations is proposed, which can improve the efficiency of detecting occlusions in dense matching fields.

III. TWO-STEP OCCLUSION DETECTION

In this section, we present our occlusion detection method, TSOD, and discuss its main features. TSOD is constituted by two main procedures: coarse detection based on the statistical matching constraint and fine detection based on the triangulation with descriptors. An overview of the TSOD algorithm is given in Fig. 2. We first discuss the statistical matching model in Section III-A and then detail the coarse detection procedure for dense correspondence fields in Section III-B. Finally, we describe the fine detection procedure which is constituted by misjudgment region extraction and triangulation-based correction, in Section III-C.

A. MATCHING STATISTICS MODEL

The statistical matching model is essentially a statistical formulation for motion smoothness based on the number of neighboring matches. For a pair of images taken from different views of the same 3D scene, a descriptor matching implies a pixel (descriptor point) in one image is identified as the same point in the other image. If the motion is smooth, neighboring pixels and features share similar motion. Therefore, the assumption of motion smoothness indicates a neighborhood around a true match to view the same 3D location. Likewise, the neighborhood around a false match views geometrically different 3D location.

Matching statistics is performed under the assumption that a single false match occurring in the true match region is a small probability event. For a single match, the number of supporting matches around its neighborhood is counted using a statistical formulation, and then if this number is greater than the desired threshold, this match is judged to be correct and vice-versa. Grid technology can effectively improve the real-time and robustness of this model. According to the structure of the grid framework, descriptors in the image correspond to each grid-cell. Then, the matching statistics procedure is performed to counting the number of matches in the neighborhood of each grid-cell. Matching statistics and grid framework constitute the main components of the GMS algorithm. We refer readers to [40] for more detailed information.

GMS has shown clear advantages in the detection of false matches in robust correspondence fields. However, if the GMS algorithm is applied to the dense correspondence field, the resulting correspondence field remains a large number of misjudgment matches. In the GMS algorithm, most descriptors in a single grid-cell correspond to a single grid-cell, because robust descriptor such as the SIFT descriptor [41] typically manifest as visual characteristics of the cluster in the image. Therefore, this algorithm still works well only by considering the grid-pair with the highest number of matches. However, for the dense correspondence field, descriptors in a single grid-cell typically correspond to multiple grid-cells, and each grid-pair contains a similar number of matches. Consequently, the result with the GMS method will engender many misjudgment matches. Fig. 3 indicates the correspondence between the grids under different correspondence fields, which reveals the limitation of the GMS algorithm for the dense correspondence field. To address this issue, we introduce a statistical dense matching method, which can effectively detect most false matches in dense correspondence fields.

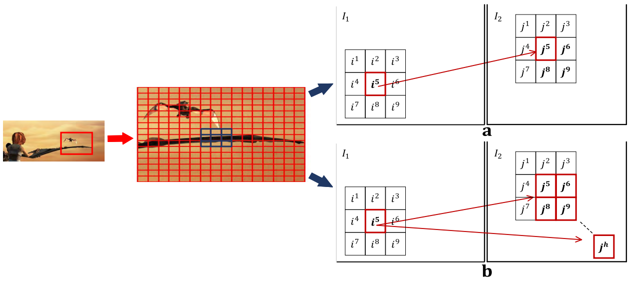


FIGURE 3. The illustration of the correspondence between the grids under different correspondence fields. Blue squares denote a 3 ∗ 3 grid-cell neighborhood and red squares in the amplified image denote uniform grid-cells with $m \times m$ descriptors. a and b denotes grid correspondences in robust and dense correspondence fields, respectively.

B. MATCHING STATISTICS FOR DENSE CORRESPONDENCE FIELDS

For a dense correspondence field, let $\mathcal{X} = \{\mathcal{X}_1, \mathcal{X}_2 \ldots \mathcal{X}_Q\}$ indicate all matches, where Q denotes the total number of image pixels, each match $\mathcal{X}_i = \{d_i, d_i'\}$ defines a correspondence between a descriptor (pixel point) $d_i \in I_1$ and a descriptor $d_i' \in I_2$. According to the analysis in Section I, we define our goal of occlusion detection as removing false matches from $\mathcal{X}$.

The first step of our approach is the construction of a grid framework for the matching field $\mathcal{X}$. The grid resolution of dense correspondence fields needs to be higher than that of robust correspondence fields, and more grid-cells can improve match localization. However, the overhigh grid resolution will reduce the number of descriptors in each grid-cell and thus separability of true and false matches. Therefore, we divide an image into discrete squares, where each square is constituted by $m \times m$ descriptors. We will show that parameter m can help balance the accuracy and running time of occlusion detection.

After the construction of the grid framework for dense matching fields, we perform the statistical matching measure for rejecting grid-cells that contain false matches. In the one-to-many grid correspondence, we first calculate the case of the one-to-one grid correspondence. To improve the robustness of this model, we selected a $k \times k$ neighborhood to count the number of matches within the grid-cell neighborhood. For the purpose of convenience, denote S_{ij} as the supporting matching score for grid-pair $\{i, j\}$:

$$S_{ij} = \sum_{n=1}^{K} \left\| \mathcal{R}(\mathcal{X}_{i^n j^n}) \right\| \tag{1}$$

where K is the number of disjoint grid-cells in the $k \times k$ neighborhood, $\|\cdot\|$ denotes the count operation on the matching grid-pair, $\mathcal{R}(\mathcal{X}_{i^n j^n})$ represents the neighborhood around the grid-pair $\{i, j\}$, and $\mathcal{X}_{i^n j^n}$ is the number of matches between grid-cells $\{i^n, j^n\}$. This statistical measure of one-to-one grid correspondence obviously provides insufficient information to judge if all the descriptors in a grid-cell are correct. Here, we define the sum of all grid-pairs scores as the final score S_f of the grid-cell:

$$S_f = \sum_{h=1}^{H} S_{ij}^h \tag{2}$$

where S_{ij}^h denotes the score of each grid-pair, which is calculated by Eq. (1), and H denotes the number of grid-pairs. Matching statistics is performed on the one-to-many grid correspondence. This is a necessary operation to ensure the validity of the neighborhood matching statistics for each grid-cell.

After counting the neighborhood scores of grid-cell, we can judge if every grid-cell is correct by comparing the total score S_f with the desired threshold. In practice, more descriptors require more neighborhood matching supports.

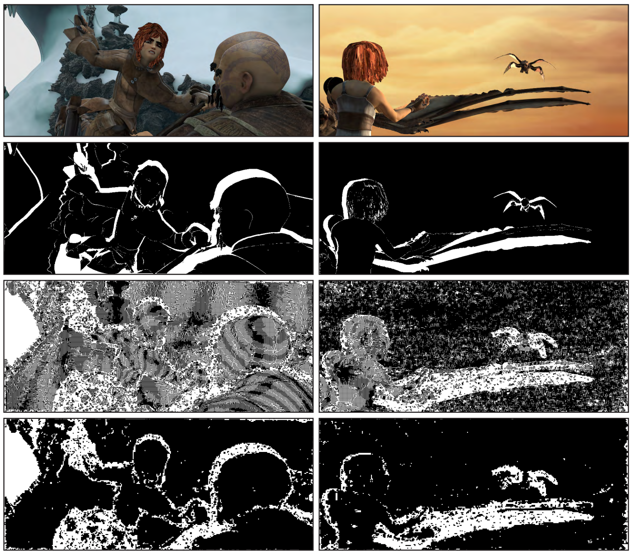


FIGURE 4. Comparison of results of occlusion detection. From left to right: The sequences of ambush_6 and temple_3 from the MPI-Sintel dataset; from top to bottom: The mean of two consecutive images, ground truth of occlusions, results of article [40] and results of the proposed method.

The threshold under the one-to-many grid correspondence is related to both the number of grid-pair and the number of descriptors in the grid-cell. Different from [40], we define the sum of every grid-pair threshold as the desired threshold τ_f, that is:

$$\tau_f = \alpha_f \cdot \sum_{h=1}^{H} \sqrt{n_h} \tag{3}$$

where n_h represents the number of matches for each grid-pair, and α_f is obtained via experimentation to ensure the confident rejection of the wrong grid-cell. In addition, executing this operation for all grid-cell, the false grid-cells are obtained for $S_f \leq \tau_f$. Fig. 4 shows results of occlusion detection on several typical sequences with obvious occlusions from the MPI-Sintel database by using the proposed method or the method presented in [40]. A visual comparison of the results indicates that the proposed method provides better performance that of the method [40].

C. MISJUDGMENT CORRECTION OF BOUNDARY GRID REGIONS

The grid framework helps improve the performance of the algorithm, but for the dense correspondence field, it also can bring misjudgment matches in some grid-cells, especially those located at the boundary of occlusion regions. Fig. 5 shows that grid-cells at the occlusion boundary typically contain a certain amount of misjudgment matches. We use the following procedure to deal with these misjudgment matches. First, we extract the boundary of the false grid region that contains misjudgment matches. Afterward, we use the descriptor-based triangulation to correct these matches.

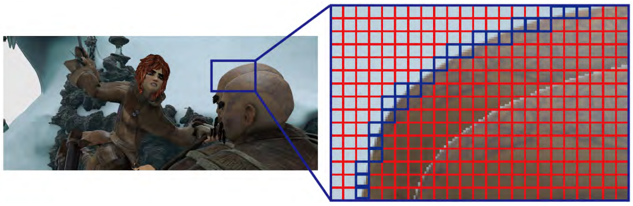


FIGURE 5. Illustration of grid-cells in the occlusion boundary. Blue squares denote the boundary of the occlusion region.

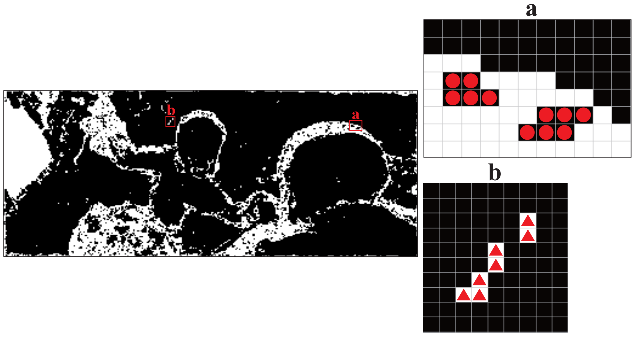


FIGURE 6. The illustration of several noises in the grid image. a and b denote the amplified images of small holes in the false grid region and isolated grid-cells in the true grid region, respectively. In the amplified images, red circles denote the erroneous holes and red triangles denote the isolated grid-cells.

1) BOUNDARY EXTRACTION OF FALSE GRID REGIONS

Let $G(x, y)$ indicate a binary image of grid judgments. $\mathcal{T}(x, y)$ and $\mathcal{F}(x, y)$ respectively denote true and false grid-cell set, where (x, y) indexes the location of grid-cell in the grid image. For the binary image, detecting changes in intensity to find boundaries can be accomplished using gradient operators, such as the Soble operator. However, there exists noise in the grid image that may result in deviations from the real boundary and the additional correction computation, especially small holes in the false grid region and isolated grid-cells in the true grid region. Fig. 6 illustrates the close-up view of the details of noises. Hence, to ensure the accuracy of the boundary grid extraction, the grid image needs to be filtered to reduce noise.

For holes in the region of the false grid-cell, we use hole area and similarity measure as a double-threshold to determine whether they are noise. One measure to detect erroneous holes is to compute a distance, $D(\bar{g}, \bar{u})$, between the average color of all the grid-cell in the hole region, $\bar{g}$, and the average color of grid-cell $\bar{u}$. In practice, $\bar{g}$ and $\bar{u}$ in an n-dimensional vector, according to the Euclidean distance, $D(\bar{g}, \bar{u})$ is defined as:

$$D(\bar{g}, \bar{u}) = \|\bar{g} - \bar{u}\| = \left[(\bar{g} - \bar{u})^T(\bar{g} - \bar{u})\right]^{\frac{1}{2}} \tag{4}$$

Another measure is the hole area $Area_4$. Since some of the small holes are at the edge of the occlusion boundary, $Area_4$ is calculated via a basic region-growing algorithm based on 4-connectivity. In the right column of Fig. 6, the area of

a shows several erroneous hole regions at the edge of the boundary. The double-threshold divides $\mathcal{T}(x, y)$ into true and false sets $\left\{\mathcal{F}_{\mathcal{T}(x,y)}, \mathcal{T}_{\mathcal{T}(x,y)}\right\}$ as follows:

$$\mathcal{T}(x, y) \in \begin{cases} \mathcal{F}_{\mathcal{T}(x,y)} & \text{if } Area_4 < \tau_a \;\; AND \;\; D(\bar{g}, \bar{u}) < \tau_s \\ \mathcal{T}_{\mathcal{T}(x,y)} & \text{otherwise} \end{cases} \tag{5}$$

where τ_a and τ_s respectively denote the area and the similarity measure threshold. By setting a proper double-threshold, the noise in the grid image can be well filtered. In our experiment, we set τ_a as follows:

$$\begin{cases} \tau_a = 2 + 100 \times Number^{-1} \\ \tau_a = min(\tau_a, 10) \end{cases} \tag{6}$$

where *Number* is the number of descriptors in the grid-cell, *min* is the minimization operation, and τ_s is experimentally determined. In practice, we first use the area threshold to detect erroneous holes and then use the similarity measure to detect remaining holes further.

The detection of isolated false grid-cells is also based on the region-growing algorithm. To preserve some the tiny and semi-continuous occlusion boundary, we select a $n \times n$ neighborhood as a growing region of each false grid-cell and count the number of all isolated grid-cells that meet the growing condition, as shown in **b** in Fig. 6 where the growing region of each red triangle is set to a $n \times n$ neighborhood. Let S_i denote the number of isolated false grid-cells in the neighborhood. By executing the filtering operation for false grid-cells, the isolated false grid-cells are obtained for $S_i < \tau_i$, where τ_i indicates the threshold of the isolated grid-cell, which is set to no more than 5 in our experiment.

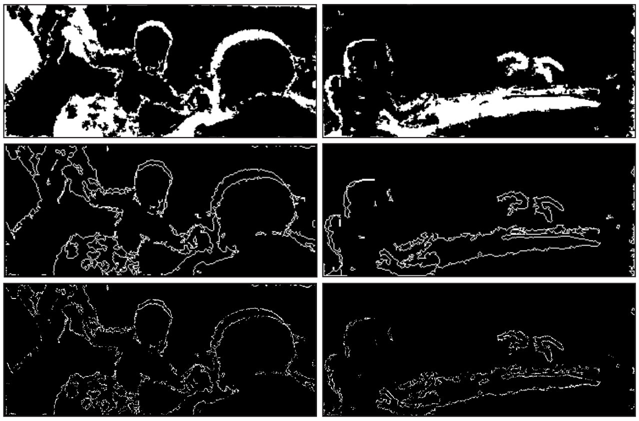


FIGURE 7. Example of misjudgment correction. From left to right: The sequences of ambush_6 and temple_3 from the MPI-Sintel dataset; from top to bottom: Results of grid filtering, results of boundary extraction, results of misjudgment correction.

After filling the wrong hole and removing the isolated false grid-cell, we perform the boundary extraction of false grid regions using the Sobel operator. As shown in Fig. 7, the double-threshold measure can effectively reduce noise in the grid image, ensuring that the gradient operator can accurately extract the occlusion boundary that contains misjudgment matches.

2) MISJUDGMENT CORRECTION BASED ON TRIANGULATION

Matches on the occlusion boundary are vital for subsequent motion estimation. In this section, we will focus on how to correct misjudgment matches in the boundary grid. Triangulations of an image is considered to be an effective method of detecting motion occlusions during the variational optical flow computation [23]. This method typically constructs a set of discrete triangles along low-level image data such as image edges or the motion boundary and then determines occlusions by comparing the data changes between triangle regions and embedded pixels. Obviously, the performance of this approach depends on the locating of the motion boundary. In fact, the occlusion region obtained by the proposed method may follow real motion boundary regions more closely. Based on these, we use triangulations to correct the misjudgment matches from the extracted boundary region.

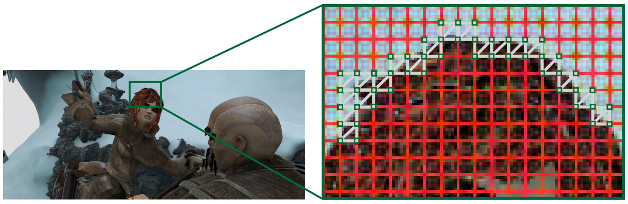


FIGURE 8. Illustration for triangulations of the occlusion boundary. The triangulation is constituted by the white lines and green squares, where green squares denote vertex points of triangles.

For the extracted region, we first construct discrete triangles to cover all descriptors in this region. Since this step is a "fine detecting" for the matches in the boundary region, we set each independent triangle to contain three descriptors to ensure that it is within the grid-cell. Fig. 8 shows the schematic of the triangulation. After the construction of triangles, we perform a data comparison between triangle regions and embedded pixels. Let ΔD_d indicate the data changes of the descriptor between consecutive frames. It is thus defined as:

$$\Delta D_d = \|D(x+u) - D(x)\|_2 \tag{7}$$

where $\|\cdot\|_2$ denotes the operation of the Euclidean distance, and $x = (x, y)^T$ represents a descriptor location and $u = (u, v)^T$ is the displacement of each descriptor in the x- and y-directions between consecutive frames. According to the definition of the descriptor data change, the data change ΔD_T of the triangle is deduced as:

$$\Delta D_T = \left\|D_{T_1} - D_{T_2}\right\|_2 \tag{8}$$

where D_{T_1} and D_{T_2} respectively denote the data of three descriptors in the corresponding triangle between I_1 and I_2.

After calculating the data change of triangles and descriptors, we perform the misjudgment correction based on these data change. We use the following procedure. First, we determine the location of each triangle in I_2 by using the correspondence field and the location of each triangle in I_1, and then judge whether these triangle regions are embedded by other pixels. Since each triangle can be considered as an independent motion unit, the triangle is deemed to be non-occlusion when no embedding occurs. For each triangle in the boundary region $\mathcal{R}$, when embedding occurs, we determine the occlusion of triangles from extracted boundary regions by the following criteria:

$$\begin{cases} \text{if } \Delta D_T > \Delta D_d \ \ AND \ \ d \in \mathcal{R}, \\ \quad \text{the descriptor embedded is a misjudgment match} \\ \text{if } \Delta D_T < \Delta D_d \ \ AND \ \ T \in \mathcal{R}, \\ \quad \text{the descriptors in } T \text{ are misjudgment matches.} \end{cases} \tag{9}$$

Fig. 9 shows an example of misjudgment correction and results of occlusion detection on two typical sequences with obvious occlusions from the MPI-Sintel database by using the proposed method or the method presented in [23]. A visual comparison of the results indicates that the proposed method provides better performance than that of the method in [23].

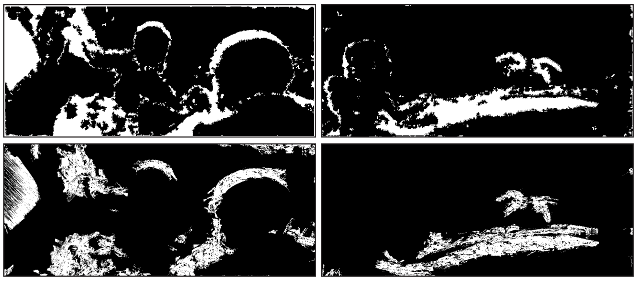


FIGURE 9. Comparison of results of occlusion detection. From left to right: The sequences of ambush_6 and temple_3 from the MPI-Sintel dataset; From top to bottom: results of the proposed method and results of the article [23].

IV. EXPERIMENTS AND ANALYSIS

This paper focuses on using the TSOD method to remove false matches that are caused by occlusions in dense matching fields. To verify the validity of the proposed method, we test the proposed method in two state-of-the-art datasets to compare with several occlusion detection methods to make a convincing evaluation. For comparison purposes, we use the same dense matching algorithm to generate input matches. CPM [14] is a PatchMatch algorithm based on a coarse-to-fine strategy, and it can quickly produce dense and accurate input matches. In the CPM method, the sample interval of the descriptor is set to 1 to produce the dense correspondence field, and the other parameters required are set to the default values provided. In the case of the proposed method, we set parameters fixed $m = 3$, $k = 3$, $\alpha_f = 6.5$ for the dense statistical matching procedure and $\tau_s = 18, n = 3$ for the misjudgment correction procedure. Fig. 10 provides an overview of the proposed method. All algorithms were run in Visual Studio 2015 on a PC with an Intel Core i5-2410M, 2.30 GHz CPU with a single-core implementation.

In the following, we first introduce a measure of performance for occlusion detection in Section IV-A. Section IV-B then runs the proposed method on two challenging datasets in comparison with several occlusion detection methods. In Section IV-C, we compare and analyze the evaluation results. Finally, we analyze the effects of inputs and parameters on the performance of the algorithm in Section IV-D.

```
Input: Dense correspondence fields
Initialization:
Divide two images by $G$ grids respectively; each grid-cell contains $m \times m$ descriptors
Compute $S_{ij}$ for each grid-pair using Eq. (1)
for $i = 1$ to $G$ do
    for $j = 1$ to $H$ do
        $S_f += S_{ij}$;
    end for
    Compute $\tau_f$;                    ▷ Eq. (3)
    if $S_f \leq \tau_f$ then
      $occlusions = occlusions \cup \mathcal{X}_{i^n j^n}$;
    end if
end for
Extract the occlusion boundary $\mathcal{R}$ and construct discrete triangles $T$;
Compute $\Delta D_T$ and $\Delta D_d$ using Eq. (7)(8)
for $i = 1$ to $T$ do
    if $T$ is embedded then
      if $\Delta D_T > \Delta D_d$ then
         $occlusions = occlusions \cup d$;
      else
         $occlusions = occlusions \cup T_i$;
      end if
    end if
end for
Output: $occlusions$
```

FIGURE 10. Overview of the TSOD method.

A. ERROR MEASURE

For a quantitative evaluation of the proposed method, we employ omission rate (OR) and false rate (FR) as the measurement of performance for occlusion detection. Wherein, OR indicates the ratio of the number of undetected occlusion points to the total number of the pixels in the occlusion regions of the image, and FR indicates the ratio of the number of the non-occlusion point that is erroneously detected as the occlusion point to the total number of the pixels in the occlusion regions of the image. The error measurement formulation is defined as follows:

$$OR = \frac{N_{omission}}{N_{occ}} \times 100\% \tag{10}$$

$$FR = \frac{N_{false}}{N_{occ}} \times 100\% \tag{11}$$

where $N_{omission}$ denotes the number of the undetected occlusion point, N_{false} denotes the number of the non-occlusion point that is erroneously detected as the occlusion point and N_{occ} denotes the total number of the pixels in the occlusion regions of the image.

B. RESULTS FROM THE MPI-SINTEL AND KITTI DATASETS

In this section, we test the proposed method using sequences from the MPI-Sintel dataset [42]. The MPI-Sintel dataset is a challenging evaluation dataset based on an animated movie and contains several typical motion forms, such as large displacement motion, motion occlusion, non-rigid motion, and complex scene. We thus run our method on the training set of the MPI-Sintel dataset for comparison with several

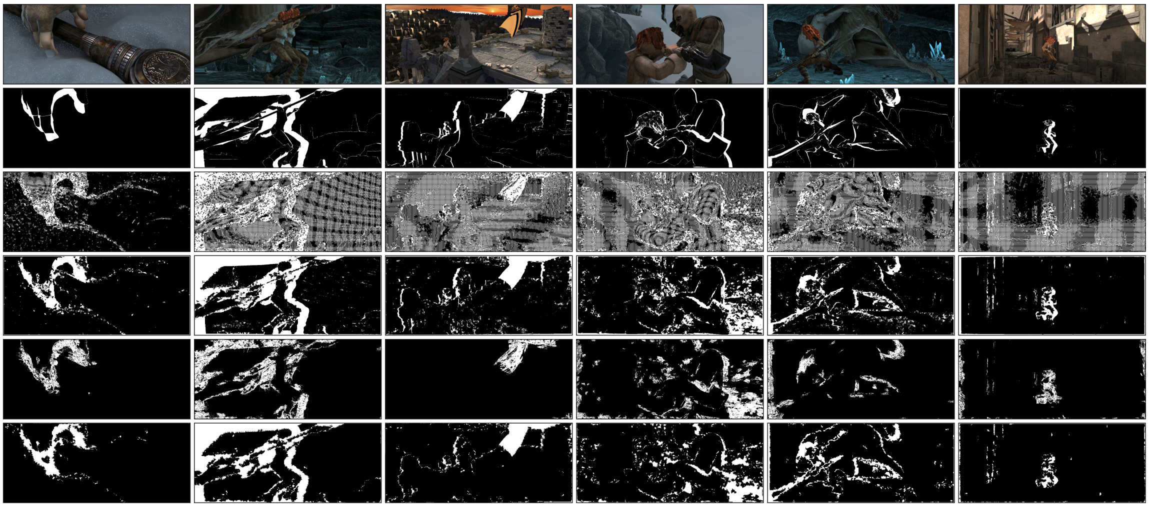


FIGURE 11. Comparison of results with several occlusion detection methods. From left to right: The sequences of ambush_7, cave_2, temple_2, ambush_5, cave_4, and alley_2 from the MPI-Sintel dataset; from top to bottom: The mean of two consecutive images, ground truth of occlusions, results of GMS, results of FBCC, results of TBOD, and results of TSOD.

TABLE 1. OR/FR results of eight training sequences from the MPI-Sintel dataset.

Method	alley_2	temple_2	temple_3	ambush_5	ambush_6	ambush_7	cave_2	cave_4
GMS	33.3/1800	32.6/527	25.2/302	41.7/919	32.1/344	30.4/231	22.5/173	39.7/610
FBCC	38.9/320	32.3/104	20.4/170	53.8/332	29.3/129	15.9/191	10.8/49.3	41.1/157
TBOD	38.6/220	60.7/45.2	43.4/71.5	67.7/247	46.2/88.9	32.3/105	43.4/45.6	67.2/65.0
TSOD	30.3/195	33.3/42.9	19.8/81.5	42.9/251	28.7/92.3	12.3/94.2	12.7/37.5	40.8/110

methods: GMS [40], triangulation-based occlusion detection (TBOD) [23] and the forward-backward consistency check (FBCC). Fig. 11 respectively shows the mean of two consecutive images, ground truth of occlusions, and the visual results of the compared methods for the evaluation sequences of the MPI-Sintel dataset. As can be seen from Fig. 11, the GMS method can identify most occlusion regions, but at the same time, results also contain a large number of false detections. The FBCC method is useful in occlusion detection due to the bidirectional checking, but in the low-texture region, there still exists some false detections. In contrast, the result of the TBOD method contains less false detections, but there exists an apparent omission detection for the motion occlusion of the tiny image structure, such as the results of the ambush_3 and temple_3 training sequences. The proposed method shows better results than the GMS and TBOD method, and the result of the proposed method has fewer false detections in low-texture regions than the FBCC method.

For a quantitative comparison, Table 1 summarizes results of OR and FR of occlusion detection of eight training sequences from the MPI-Sintel dataset. In Table 1, the FR of the GMS method is high above that of other methods, which indicates there exist many errors in the detection result. The OR of the TBOD method is high above that of other methods for most evaluation sequences, which indicates that the TBOD method cannot effectively identify the motion occlusion region in the image sequences. The FR of the FBCC method is much lower than the GMS method, and the OR of FBCC is lower than TBOD, which indicate that the FBCC method has certain robustness. The TSOD method shows better performance than the FBCC method on both OR and FR, especially that the FR of TSOD is approximate twice as low as that of FBCC, which indicates there exist fewer errors in the detection results. In conclusion, the performance of TSOD better than the above three occlusion detection methods, which indicates the proposed method can detect occlusions of the image sequence more completely and ensure the accuracy of the detection results.

Although the MPI-Sintel dataset enables a fair evaluation for the comparison method of occlusion detection, we additionally apply the proposed method on other evaluation sequences in the KITTI dataset [43], [44] to ensure a persuasive comparison with the other compared methods. The KITTI dataset was created from a platform on a driving car and contains images of city streets. We choose sequences of the KITTI dataset that contain several typical motion forms, such as motion occlusion, large displacement motion, and complex scene.

Fig. 12 respectively shows the mean of two consecutive images, ground truth of dense occlusions (computed with [23]), and the visual results of the compared methods for the evaluation sequences of the KITTI dataset. The results

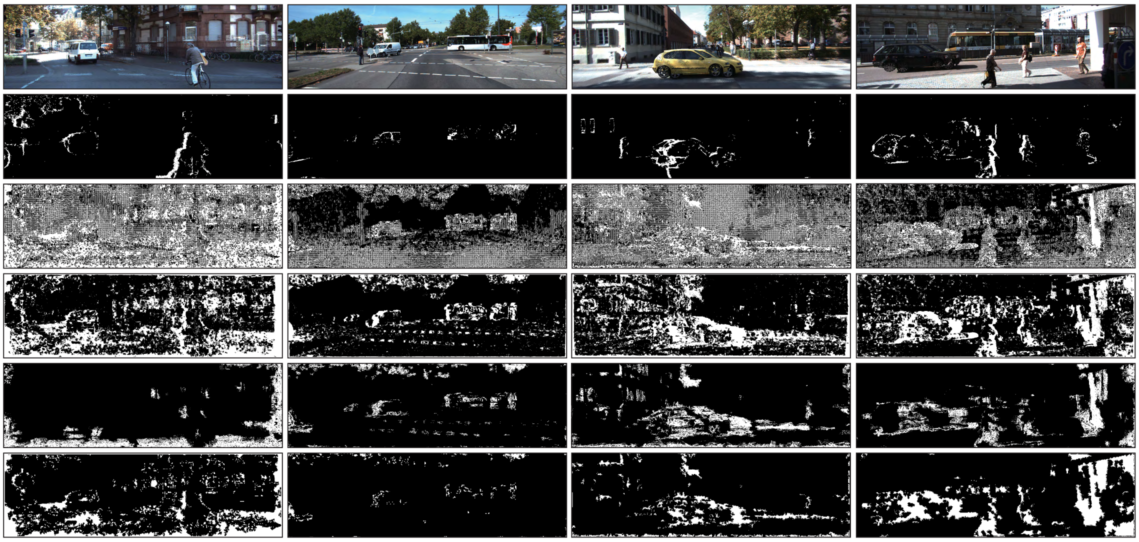


FIGURE 12. Comparison of results with several occlusion detection methods. From left to right: The sequences of 000002, 000016, 000167, 000169 from the KITTI-2015 dataset; from top to bottom: The mean of two consecutive images, ground truth of occlusions, results of GMS, results of FBCC, results of TBOD, results of TSOD.

of the GMS method contain a large number of detection errors, which indicates this method is not suitable for the dense matching fields. The TBOD method exists a significant omission detection for some motion occlusion regions, such as the cyclist and car in the 000002 sequence. The FBCC method can identify most motion occlusion regions, but there exist many errors in the low-texture region. The results of the proposed method show good visual performance for the detection of the motion and low-texture regions.

TABLE 2. OR/FR results of four training sequences from the KITTI-2015 dataset.

Method	000002	000016	000167	000169
GMS	48.9/1538	55.8/3600	46.5/2262	50.3/1031
FBCC	52.6/1015	55.1/1893	50.9/1196	55.5/838
TBOD	80.5/853	74.1/746	75.9/554	72.3/642
TSOD	47.2/578	53.8/300	48.5/383	51.3/485

Table 2 exhibits detailed comparative results of OR and FR of several occlusion detection methods for the KITTI-2015 [44] evaluation sequences. As can be seen, the FR of the GMS method and the OR of the TBOD method are highest among the four occlusion detection methods, and the FR of the FBCC method is much higher than that of the proposed method. Evidence shows that the proposed method can effectively detect motion occlusion and has certain robustness for the real scenario sequence.

C. COMPREHENSIVE EVALUATION AND ANALYSIS

To comprehensively evaluate the proposed method, we respectively calculate the average omission rate (AOR) and average false rate (AFR) of GMS, TBOD, FBCC, and the proposed method for 20 training sequences from MPI-Sintel [42], KITTI-2012 [43], and KITTI-2105 [44] datasets.

TABLE 3. AOR/AFR results of compared methods for three training datasets.

Method	MPI-Sintel		KITTI-2012		KITTI-2015	
	AOR(%)	AFR(%)	AOR(%)	AFR(%)	AOR(%)	AFR(%)
GMS	71.8	618.2	55.3	2120.4	51.3	2111.7
FBCC	33.3	177.5	54.4	1244.3	52.5	1237.4
TBOD	72.9	120.5	81.4	703.3	74.8	672.6
TSOD	28.6	108.5	52.3	451.9	50.6	441.5

From Table 3, we can see that the omission rate of the proposed method is lower than other occlusion detection methods. More importantly, the proposed method leads to a significant degradation in the false rate, especially that the FR of TSOD is approximately two and a half times lower than the FR of FBCC. Evidence shows that the proposed method can detect motion occlusion regions more completely for image sequences in different scenes, and has higher accuracy and robustness than GMS [40], TBOD [23], and FBCC.

D. PERFORMANCE ANALYSIS

To get a better understanding of the performance of the proposed method, we evaluate the impact of different input matches, the procedure of misjudgment correction and the parameters of grid size m on the results of occlusion detection of the proposed method. To facilitate quantitative evaluation of the proposed method, we used the performance measure of optical flow algorithms [45], the average angular error (AAE) and the average endpoint error (AEE), to evaluate the occlusion detection performance of the proposed method. Simultaneously, for fair comparison, we use EpicFlow [17] to interpolate matching correspondences that are handled by using the proposed method to a dense optical flow, which is the same as the interpolate procedure in the state-of-the-art optical flow algorithm. The performance analysis of the proposed method is tested on a subset of the clean-version training set of the MPI-Sintel dataset.

To evaluate the robustness of the proposed method, we generate input matches using three matching methods, where of all are tailored to the problem of optical flow estimation. The first choice is Deep Matching approach (DM) [21], which has shown excellent performance for large displacement motion in optical flow computation. The second choice is DiscreteFlow (DF) [22], which uses explicit regularization to obtain pixel-accurate optical flow fields. The last choice is the recent CPM approach [14], which can quickly produce dense accurate matching correspondences. We perform occlusion detection with the proposed method on three input matches and use EpicFlow [17] to generate a dense flow field. To further test the robustness of the proposed method, we add various levels of matching noise to input matches and evaluate the performance of the proposed method. Fig. 13 clearly shows the robustness of the proposed method under several dense input matches with different levels of matching noise. Not only that, the proposed method improves the accuracy of the flow field for the three optical flow algorithms, which shows the effectiveness of the proposed method.

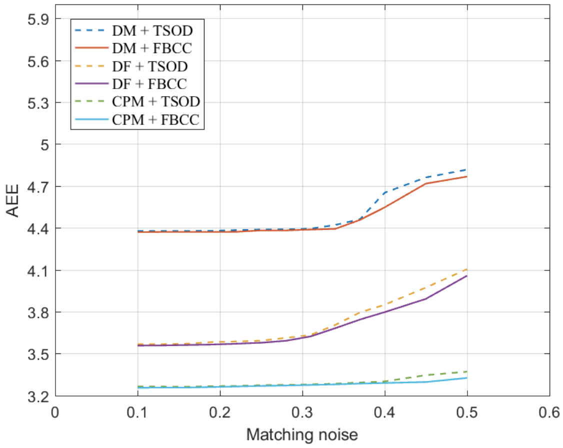


FIGURE 13. The robustness of the TSOD method.

TABLE 4. Comparison of the proposed method with different operations: "MC", "BGR", and "DBT".

Method	ambush_4	cave_2	cave_4	market_2	alley_2
TSOD	7.518	2.014	5.208	5.914	1.692
−MC	7.852	2.036	5.361	5.982	1.781
−BGR	7.828	2.035	5.349	5.972	1.773
−DBT	7.723	2.108	5.339	5.930	1.712

TABLE 5. Comparison of the proposed method with different descriptors.

Descriptors	ambush_4	cave_2	cave_4	market_2	alley_2
SIFT-F	7.518	2.014	5.208	5.914	1.692
CRT	7.639	2.021	5.311	5.955	1.699
CT	7.754	2.049	5.621	6.013	1.711
RGB	7.883	2.155	6.153	6.211	1.721

Next, we evaluate the validity of misjudgment correction (MC), boundary grid extraction (BGE), and descriptor-based triangulation (DBT) on the MPI-Sintel dataset. To demonstrate the benefits of these procedures in the proposed method, we utilized these procedures as additional operations during misjudgment correction. Table 4 and Table 5 report the comparison of AAE results on sequences ambush_4, cave_2, cave_4, market_2, and alley_2 from the MPI-Sintel dataset thoroughly, where − denotes that these operations are not performed. As shown in Table 4, misjudgment correction improves the accuracy of the results of occlusion detection significantly, which indicates MC corrects a certain amount of misjudgment matches. Misjudgment correction with BGE leads to an enhancement in performance because BGE also removes a certain amount of noise while extracting regions of boundary grids. DBT improves the accuracy of the triangulation, which indicates that descriptor-based triangles effectively represent the data change of the occluded region in the image. In addition, we compare the performance of the proposed method for the triangulation based on four different descriptors: SIFT flow (SIFT-F) [46], Complete Rank Transform (CRT) [47], Census Transform (CT) [48] and RGB. We choose the Euclidean distance between descriptors as the data change. Table 5 shows the performance of each descriptor for the five training sequences from the MPI-Sintel dataset. We can see that SIFT-F is the best performer.

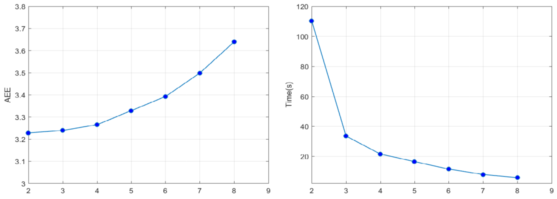


FIGURE 14. Effect of different grid sizes on the proposed method. Note that $m = 3$ hardly impairs accuracy but leads to acceptable running time.

We also report the average running time and AEE of the proposed method with different grid sizes m. As shown in Fig. 14, the small grid size improves the performance because it can more efficiently represent the motion occlusion region, which makes the procedure of matching statistics more accurate. However, a small grid size leads to a high computation complexity, especially at $m = 2$. We find that $m = 3$ hardly impairs accuracy but leads to acceptable running time.

V. CONCLUSION

In this paper, we present an efficient two-step occlusion detection method to remove false matches in dense matching fields. It contains two major steps: the coarse detection based on the statistical matching constraint and the fine detection based on triangulations of an image. In the first step, most occluded regions are accurately identified by using a statistical dense matching method. This method is developed by considering the correspondence between the grids. Second, to improve the accuracy of occlusions detection, we describe a misjudgment correction method based on the triangulation with descriptors. In this method, the region that may contain misjudgment matches is extracted by using a double-threshold filtering and a gradient operate. Then misjudgment matches in the extraction region are corrected based on the triangulation with descriptors. We conduct comparative experiments using the test sequences of the MPI-Sintel and KITTI datasets. The results show that the proposed method provides higher accuracy and robustness than typical existing occlusion detection methods.

REFERENCES

[1] P. Fua and Y. G. Leclerc, "Object-centered surface reconstruction: Combining multi-image stereo and shading," *Int. J. Comput. Vis.*, vol. 16, no. 1, pp. 35–56, Sep. 1995.

[2] T. Brox and J. Malik, "Large displacement optical flow: Descriptor matching in variational motion estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 3, pp. 500–513, Mar. 2011.

[3] A. C. Berg, T. L. Berg, and J. Malik, "Shape matching and object recognition using low distortion correspondences," in *Proc. CVPR*, Jun. 2005, pp. 26–33.

[4] C. Robin and S. Lacroix, "Multi-robot target detection and tracking: Taxonomy and survey," *Auton. Robots*, vol. 40, no. 4, pp. 729–760, 2016.

[5] M. Jian, J. Dong, and J. Ma, "Image retrieval using wavelet-based salient regions," *Imag. Sci. J.*, vol. 59, no. 4, pp. 219–231, 2011.

[6] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in *Proc. 8th IEEE Int. Conf. Comput. Vis.*, Jul. 2001, pp. 508–515.

[7] J. Sun, N.-N. Zheng, and H.-Y. Shum, "Stereo matching using belief propagation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 1, no. 1, pp. 787–800, Jul. 2003.

[8] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman, "PatchMatch: A randomized correspondence algorithm for structural image editing," *ACM Trans. Graph.*, vol. 28, no. 3, p. 24, 2009.

[9] C. Barnes, E. Shechtman, D. B. Goldman, and A. Finkelstein, "The generalized patchmatch correspondence algorithm," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2010, pp. 29–43.

[10] S. Korman and S. Avidan, "Coherency sensitive hashing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 6, pp. 1099–1112, Jun. 2016.

[11] J. Sun and K. He, "Computing nearest-neighbor fields via propagation-assisted KD-trees," in *Proc. IEEE CVPR*, Jun. 2012, pp. 111–118.

[12] N. Ben-Zrihem and L. Zelnik-Manor, "Approximate nearest neighbor fields in video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5233–5242.

[13] C. Bailer, B. Taetz, and D. Stricker, "Flow fields: Dense correspondence fields for highly accurate large displacement optical flow estimation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 4015–4023.

[14] Y. Hu, R. Song, and Y. Li, "Efficient coarse-to-fine patchmatch for large displacement optical flow," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 5704–5712.

[15] J. Lu, Y. Li, H. Yang, D. Min, W. Eng, and M. N. Do, "PatchMatch filter: Edge-aware filtering meets randomized search for visual correspondence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 9, pp. 1866–1879, Sep. 2017.

[16] A. Hosni, C. Rhemann, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 2, pp. 504–511, Feb. 2013.

[17] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid, "EpicFlow: Edge-preserving interpolation of correspondences for optical Flow," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1164–1172.

[18] Z. Chen, H. Jin, Z. Lin, S. Cohen, and Y. Wu, "Large displacement optical flow from nearest neighbor fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2443–2450.

[19] L. Bao, Q. Yang, and H. Jin, "Fast edge-preserving patchmatch for large displacement optical flow," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3534–3541.

[20] R. Timofte and L. V. Gool, "Sparse flow: Sparse matching for small to large displacement optical flow," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Jan. 2015, pp. 1100–1106.

[21] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "DeepFlow: Large displacement optical flow with deep matching," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1385–1392.

[22] M. Menze, C. Heipke, and A. Geiger, "Discrete optimization for optical flow," in *Proc. German Conf. Pattern Recognit.*, Nov. 2015, pp. 16–28.

[23] C. Zhang, Z. Chen, M. Wang, M. Li, and S. Jiang, "Robust non-local TV-optical flow estimation with occlusion detection," *IEEE Trans. Image Process.*, vol. 26, no. 8, pp. 4055–4067, Aug. 2017. doi: 10.1109/TIP.2017.2712279.

[24] M. Lemmens, "A survey on stereo matching techniques," *Int. Arch. Photogramm. Remote Sens.*, vol. 27, no. 8, pp. 11–23, Jul. 1988.

[25] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, nos. 1–3, pp. 7–42, Apr. 2002.

[26] B. Zitová and J. Flusser, "Image registration methods: A survey," *Image Vis. Comput.*, vol. 21, pp. 977–1000, Oct. 2003.

[27] A. Luo and H. Burkhardt, "An intensity-based cooperative bidirectional stereo matching with simultaneous detection of discontinuities and occlusions," *Int. J. Comput. Vis.*, vol. 15, no. 3, pp. 171–188, Jul. 1995.

[28] J. Sun, Y. Li, S. B. Kang, and H. Shum, "Symmetric stereo matching for occlusion handling," in *Proc. CVPR*, Jun. 2005, pp. 399–406.

[29] L. Alvarez, R. Deriche, T. Papadopoulo, and J. Sánchez, "Symmetrical dense optical flow estimation with occlusions detection," *Int. J. Comput. Vis.*, vol. 75, no. 3, pp. 371–385, Dec. 2007.

[30] A. Ayvaci, M. Raptis, and S. Soatto, "Sparse occlusion detection with optical flow," *Int. J. Comput. Vis.*, vol. 97, no. 3, pp. 322–338, 2012.

[31] C. Ballester, L. Garrido, V. Lazcano, and V. Caselles, "A TV-L1 optical flow method with occlusion detection," in *Proc. German Assoc. Pattern Recognit. Symp. OAGM*, Aug. 2012, pp. 31–40. doi: 10.1007/978-3-642-32717-9_4.

[32] S. Ince and J. Konrad, "Occlusion-aware optical flow estimation," *IEEE Trans. Image Process.*, vol. 17, no. 8, pp. 1443–1451, Aug. 2008.

[33] R. Ben-Ari, and N. Sochen, "Variational stereo vision with sharp discontinuities and occlusion handling," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–7. doi: 10.1109/ICCV.2007.4408996.

[34] J. Xiao, H. Cheng, H. S. Sawhney, C. Rao, and M. A. Isnardi, "Bilateral filtering-based optical flow estimation with occlusion detection," in *Proc. Eur. Conf. Comput. Vis.*, May 2006, pp. 211–224. doi: 10.1007/11744023_17.

[35] C. L. Zitnick and T. Kanade, "A cooperative algorithm for stereo matching and occlusion detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 7, pp. 675–684, Jul. 2000.

[36] D. Min and K. Sohn, "Cost aggregation and occlusion handling with WLS in stereo matching," *IEEE Trans. Image Process.*, vol. 17, no. 8, pp. 1431–1442, Aug. 2008.

[37] C. Strecha, R. Fransens, and L. J. V. Gool, "A probabilistic approach to large displacement optical flow and occlusion detection," in *Proc. Int. Workshop Stat. Methods Video Process.*, May 2004, pp. 71–82. doi: 10.1007/978-3-540-30212-4_7.

[38] B. Glocker, T. H. Heibel, N. Navab, P. Kohli, and C. Rother, "Triangle-Flow: Optical flow with triangulation-based higher-order likelihoods," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2010, pp. 272–285. doi: 10.1007/978-3-540-30212-4_7.

[39] R. Kennedy and C. J. Taylor, "Optical flow with geometric occlusion estimation and fusion of multiple frames," in *Proc. Int. Workshop Energy Minimization Methods Comput. Vis. Pattern Recognit.*, Jan. 2015, pp. 364–377. doi: 10.1007/978-3-319-14612-6_27.

[40] J. Bian, W.-Y. Lin, Y. Matsushita, S.-K. Yeung, T.-D. Nguyen, and M.-M. Cheng, "GMS: Grid-based motion statistics for fast, ultra-robust feature correspondence," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4181–4190.

[41] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sep. 1999, pp. 1150–1157.

[42] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2012, pp. 611–625.

[43] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. Int. Conf. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[44] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3061–3070. doi: 10.1109/CVPR.2015.7298925.

[45] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," *Int. J. Comput. Vis.*, vol. 92, no. 1, pp. 1–31, 2011.

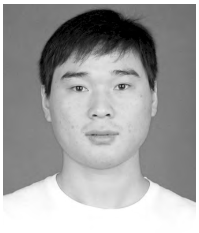
[46] C. Liu, J. Yuen, and A. Torralba, "SIFT flow: Dense correspondence across scenes and its applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 978–994, May 2011.

[47] O. Demetz, D. Hafner, and J. Weickert, "The complete rank transform: A tool for accurate and morphologically invariant matching of structures," in *Proc. BMVC*, Sep. 2013, pp. 1–12.

[48] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," in *Proc. Eur. Conf. Comput. Vis.*, May 1994, pp. 151–158.

CHONG DONG received the M.S. degree from the Anhui University of Technology, China, in 2015. He is currently pursuing the Ph.D. degree with the College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, China. His research interests include optical flow estimation, deep learning, and UAV visual navigation.

ZHISHENG WANG received the M.S. and Ph.D. degrees from Northwestern Polytechnical University, China, in 1999 and 2002, respectively. In 2002, he joined the College of Automation Engineering, Nanjing University of Aeronautics and Astronautics (NUAA), China, where he is currently a Full Professor. From 2002 to 2004, he was a Postdoctoral Fellow with the College of Automation Engineering, NUAA. From 2008 to 2010, he was a Postdoctoral Fellow with the School of Mechanical Engineering and Automation, National University of Defense Technology, China. From 2010 to 2011, he was a Visiting Professor with Stockholm University, Sweden. He has authored or coauthored more than 60 referred papers. His professional interests include information fusion, fusion control, and computer vision.



CHANGDA XING received the B.S. degree from the Qindao College, Qingdao University of Technology, China, in 2013, and the M.S. degree from the Anhui University of Science and Technology, China, in 2016. He is currently pursuing the Ph.D. degree with the Nanjing University of Aeronautics and Astronautics. His main research interests include image fusion, pattern recognition, and hyperspectral super-resolution.

JIAMING HAN is currently pursuing the Ph.D. degree with the College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, China. His current research interests include pattern recognition, image processing, robot vision/control, and deep learning.

PINGAN HUANG received the M.S. degree from Northwestern Polytechnical University, China, in 1999. He is currently the Deputy Chief Engineer of the aviation industry Xi'an Aircraft Industry, a Researcher, and an Aviation Industry Senior Expert. His current research interests include aircraft system integrated design and system integration measurement technology, aircraft research and production technology management, and robot visual navigation.