A pixel pair–based encoding pattern for stereo matching via an adaptively weighted cost

Yuli Fu | Kaimin Lai | Weixiang Chen | Youjun Xiang

School of Electronic and Information Engineering, South China University of Technology, Guangzhou, China

Correspondence
Yuli Fu, School of Electronic and Information Engineering, South China University of Technology, Guangzhou, China.
Email: fuyuli@scut.edu.cn

Funding information
Guangzhou Technical Project, Grant/Award Number: 201902020008; National Science Foundation of Guangdong Province, Grant/Award Numbers: 2019A1515010861, 2018A030313669; National Natural Science Foundation of China, Grant/Award Number: 61471174

Abstract
Stereo matching, which is a key problem in computer vision, faces the challenge of radiometric distortions. Most of the existing stereo matching methods are based on simple matching cost algorithms and appear the problem of mismatch under radiometric distortions. It is necessary to improve the robustness and accuracy of matching cost algorithms. A novel encoding pattern is proposed for stereo matching. In the proposed encoding pattern, each of the matching windows in the grey image and gradient images is divided into several isoline-like sets with different radii. Then, pixel pairs are defined in the isoline-like sets. An encoding function is used to decide the relative order between the two pixels in each pixel pair. To apply the pattern for matching cost computation and enhance the matching accuracy, an adaptively weighted cost is designed that is related to the isoline-like sets. Experiments are conducted on the Middlebury and KITTI data sets to show the validity of the proposed method under severe radiometric distortions. Also, the comparisons with some widely used methods are made in the experiments to illustrate the advantage of the proposed method.

1 | INTRODUCTION

Stereo matching aims to obtain stereo disparity. It is one of the most important and challenging topics in computer vision. The disparity map plays a key role in many applications, such as robotics, self-driving cars and 3D reconstruction. Although stereo matching has been studied for many years, accurate dense disparity map is still difficult to obtain because of the existence of occluded regions, non-texture regions, object boundaries, radiometric distortions, etc. Traditional stereo matching algorithms can be classified as local stereo algorithms and global stereo algorithms [1]. Local stereo algorithms [2–4] generally have the following four steps: matching cost computation, cost aggregation, disparity computation and disparity refinement. It only considers the local aggregation regions and the information of surrounding pixels are probably used. Global stereo algorithms [5] typically consist of matching cost computation, disparity optimisation and disparity refinement steps. It constructs a global cost function and gets the disparity result through optimising the cost function. Matching cost computation is the key step of local and global algorithms. It is indispensable for obtaining accurate disparity. The simplest matching cost such as Absolute Difference (AD), Square Differences (SD), and Sum of the Absolute Differences (SAD) [6] are based on the colour-consistent assumption that each pixel in the left image and its corresponding pixel in the right image have similar intensity value. However, in real scenario this assumption is not true because of the influence of radiometric distortions. In order to tackle the problem of radiometric distortions, Census transform [7] was introduced to establish the relative orders between the centre pixel and neighbouring pixels in the matching windows. But Census is not robust because it relies heavily on the intensity of centre pixel and ignores the relationship between neighbouring pixels themselves. Normalized Cross-correlation (NCC) [8], which is a window-based matching algorithm, performs well for dealing with Gaussian noise, but it tends to blur depth discontinuities. Based on the assumption of the Lambertian world, adaptive NCC (ANCC) [9] robustly handles various radiometric distortions. It performs well under the existence of lighting geometry and camera parameter changes. But it fails to obtain accurate disparity map while facing the severe illumination changes, large brightness differences, or a non-Lambertian...
Proposed Method

Pixel pair–based encoding pattern

Robust soft rank transform (RSRT) [13] establishes the relative orders in the same way as FEP. The difference between RSRT and FEP is the way of encoding. RSRT can be considered as a fast version of FEP. By the way, some works, such as [14], integrate the denoising process into the stereo matching to address the problem of the existences of multiple types of noise. However, they do not specifically focus on the situation where the image pairs are taken under different lighting or illumination conditions. The images of these methods are obtained under almost the same conditions, even though the existence of noises in the surrounding environment.

Instead of these hand-crafted cost computation schemes, Zbontar [15] uses the convolutional neural networks (CNNs) to learn a more comprehensive cost measurement. Through comparing the similarity of a pair of patches and using the similarity score as the matching cost, [15] gets robust result on the Middlebury [16] and KITTI [17] stereo benchmark. [15] is used to compute matching costs by many stereo matching algorithms, such as [18–21]. Also, based on [17], many CNN-based stereo matching cost computation methods [22–25] are proposed by researchers. Luo [22] proposes a matching network with fast computation. The network outputs 64-dimensional features as the representation of each patch. [22] has better matching performance by using the inner product of features as the similarity between patches. Park [23] proposes a CNN network to get matching cost with a large-sized window. It can avoid the problem of fattening effect while extracting information from a large area. Yang and Lv [24] learn a Euclidean embedding for each image using CNN with a triplet-based loss function and a smoothness constraint. They generalise the learning result to SGM [10] and get comparable performance. In addition, Dong [26] combines SAD, GRAD, and CNN [15] methods to construct a new stereo matching cost volume. They get better performance than original stereo matching costs. CNN methods tend to enhance the weak textures and exhibit imaginary textures in occlusion areas, but they require much time to train network on the specific data sets. The hand-crafted cost computation methods do not need to train and could be directly applied to real scenarios.

Our proposed encoding pattern focus on the problem of radiometric distortions. Like other encoding patterns, such as Census and FEP, the matching windows are considered for original grey images in our proposed encoding pattern. The way of dividing the matching windows is inspired from traditional image feature extraction method: histogram of oriented gradients (HOG) [27]. HOG computes on uniformly spaced cells and maintains good invariance to both geometric and photometric deformations of the images. Radiometric distortions, which involve the deformations, could be avoided to some extent since the influences of distortions would be smaller in the small local areas. In FEP, the two non-adjacent pixels are considered. Then, the way of establishing the relative orders in FEP may be not robust enough. To obtain more robust results, a novel encoding pattern is proposed. Unlike other encoding patterns such as Census and FEP, the proposed encoding pattern focuses on the pixels which are neighbouring. On the other hand, a new encoding function is designed to obtain the accurate relative orders. For the relative orders obtained by the proposed encoding method, the stereo matching cost should be considered. Unlike the costs used in Census and FEP, the differences between the relative orders will be considered via some weights. As in [28], the gestalt grouping by proximity will be used to establish our adaptive weights in the cost. The main contributions of this paper are summarised as follows:

1. A novel encoding pattern is proposed. It is consistent with HOG’s methodology and maintains good invariance to the distortions since the distortions appear rarely in small area. Thus, it can improve robustness under the radiometric distortions.

2. A discretisation encoding function, which is easy to tune the size of the interval, is proposed. Comparing with Census’s encoding function, the proposed function considers the intensity difference between two pixels. Comparing with FEP’s encoding function, the proposed function is more flexible.

3. After encoding, in order to guarantee accuracy of stereo matching, an adaptively weighted cost related to the isoline-like sets is designed. It is consistent with the gestalt grouping by proximity.

The experiments will be given later to show the strength of the proposed method. Comparing with the traditional methods, it could be robust against the radiometric distortions and practical for real-world applications. Also, comparing with the CNN-based methods, our method can be used without training. In fact, selecting a specific and effective training data set for a neural network could be a difficulty in practice. The rest of the paper is organised as follows. Section 2 presents the proposed method in detail. In Section 3, comprehensive experiments on the Middlebury and KITTI data sets are made to verify that the proposed method can improve some previous method for stereo matching in the case of the radiometric distortions. Finally, we conclude our work in Section 4.
the matching windows. These methods ignore the relationship between neighbouring pixels themselves and fail to get robust results. Our encoding pattern does not encode the relative orders between centre pixel and its neighbouring pixels. It is inspired by the traditional image feature extraction method HOG. It extracts the relative orders on small local areas.

For every pixel in the original images, the matching windows centred at it will be constructed with radius \( R \) (an odd and positive integer). If the matching windows is uncompleted (e.g. the centres are near the borders), the rest pixels will be filled with zeros to make the windows complete. For example, for the pixels near the upper border of the image in Figure 1, zeros are filled to get the complete matching window. Then, in the matching window, pixels with the same radius are considered together. We define two adjacent pixels as pixel pairs

\[
\mathbf{S}_{(x_c, y_c)} \leftarrow \left\{ p \mid \max(|x - x_i|, |y - y_i|) = r \right\}, \tag{1}
\]

where \( x \) and \( y \) are \( p \)'s coordinates.

In the proposed encoding pattern, pixels in the matching window are indexed around the centre. For the matching window, the centre pixel is indexed as 0. The other pixels are indexed along the isoline-like sets. The pixels in \( S_{(x_c, y_c)}^1 \) are indexed as 1–8. Similarly, the pixels in \( S_{(x_c, y_c)}^2 \) are indexed as 9–24. These pixels are expressed as \( p_1, p_2, \ldots, p_{24} \) according to their indexes. For example, Figure 2 shows the division of the isoline-like sets in the 5 × 5 matching window. In Figure 2, \( S_{(x_c, y_c)}^1 \) and \( S_{(x_c, y_c)}^2 \) are shown in the mid image and right image, respectively. To be consistent with HOG's methodology, the small local areas are considered in the proposed encoding pattern. The isoline-like sets are further divided into some local areas. It would be helpful for avoiding the distortions since the severe distortions would often appear on big spaced area. Since it is not easy to decide the size of the local area to avoid radiometric distortion, the proposed encoding pattern considers the local relationship extremely, that is, only two neighbouring pixels are considered. For each of the isoline-like sets, the two pixels with adjacent indexes are considered together. By the way, the indexes in the isoline-like sets are considered as rings. Then pixels indexed as 8 and 1 are also adjacent and \( p_8 \) and \( p_1 \) in the \( S_{(x_c, y_c)}^1 \) are also considered together. We define two adjacent pixels as pixel pairs in the isoline-like sets. For example, \( p_1 \) and \( p_2, p_3, p_4, \ldots, p_{24} \) are the pixel pairs. The isoline-like sets are further divided into several pixel pairs. The pixel pairs of \( S_{(x_c, y_c)}^1 \) and \( S_{(x_c, y_c)}^2 \) are shown in Figure 3. They are indexed as 1–24. Our encoding pattern is strongly related to our defined pixel pair. Then we name it as pixel pair based encoding pattern (PPEP). A pixel pair could be considered as an extreme small local area. By the way, we considered that the design of our pixel pair could get more compact encoding information and reduce the complexity of total relative orders since a pixel pair only obtains one relative order.

For the pixel pairs, the relative order is encoded between the centre pixel and its neighbouring pixels.

\[
f(q, q') = \text{sign}(p_q - p_{q'}) \frac{|p_q - p_{q'}|}{m}, \tag{2}
\]
where $\sgn(\cdot)$ is the symbolic function and $q$ and $q'$ are the indexes of two pixels in the pixel pairs. $\lceil \cdot \rceil$ is the function of getting the smallest integer not less than itself. $m$ is a scale factor, which is used to tune the interval of the function. In some extent, the function (2) can represent the information of intensity difference between two pixels. For example, by Census, the order is determined as $1$ for the two pixels with intensity values $60$ and $25$, and the same order is determined for the two pixels with intensity values $28$ and $25$. Obviously, this ambiguity can be an obstacle. By encoding function (2), the ambiguity will be avoided. In addition, the proposed encoding function (2) does not have the limitations of upper and lower bounds, which is different from the fuzzy encoding function in FEP. When (2) is used to encode, the scale factor $m$, which means the size of the interval, is flexible to tune.

For the isoline-like set $S_{(xc,yc)}^r (r \in \{1,2,...,R\})$, the minimum and maximum indexes of the pixel pairs are expressed as $MIN_r$ and $MAX_r$, respectively. The relative order between the two pixels in the $kth$ pixel pair is expressed as $\varphi^k_{(xc,yc),I}$, which is an integer calculated by the function (2). The subscript $I$ represents establishing the relative order in the original grey image.

In addition, the proposed method also considers gradient images in $x$ direction and $y$ direction. The image gradient information can reflect the change trend between local pixel values. It can bring the information of edge to some extent since the gradient values in other areas. We consider that more robust and distinguishable relative orders could be established. Let $S_X$ and $S_Y$ be the $3 \times 3$ Sobel matrices for computing gradient images in $x$ direction and $y$ direction, respectively. Gradient images can be computed as

$$G_X = S_X \ast I,$$

and

$$G_Y = S_Y \ast I,$$

where $\ast$ is the image convolution operator. Like the original grey image, the relative orders of gradient images are obtained and are expressed as $\varphi^k_{(xc,yc),G_X}$ and $\varphi^k_{(xc,yc),G_Y}$, respectively. Then, the relative orders are concatenated and expressed as

$$\varphi^k_{(xc,yc),r} = \varphi^k_{(xc,yc),G_X} \otimes \varphi^k_{(xc,yc),G_Y},$$

where $\otimes$ is the concatenation operator.

Then, every $\varphi_{(xc,yc),r}$ are concatenated. The total relative orders of the isoline-like sets with radius $r$ in the original grey image and the gradient images are expressed as

$$\eta^1_{(xc,yc),r} = \varphi_{(xc,yc),r}^{MIN_r} \otimes \varphi_{(xc,yc),r}^{MIN_r+1} \otimes \varphi_{(xc,yc),r}^{MAX_r},$$

Take $m = 15$ and $r = 1$ as an example, in Figure 4, the relative orders of the isoline-like set $\eta^1_{(xc,yc),r}$ in the original grey image and the gradient images are obtained, respectively. Then, the total $24$ relative orders are concatenated according to the function (5) and (6). $\eta^1_{(xc,yc),I} \otimes \varphi_{(xc,yc),r}$ is consisted of $24$ relative orders. Let $\eta^1_{(xc,yc),i}$ be the values at index $i$ in $\eta^1_{(xc,yc),I}$. It is showed that $\eta^1_{(xc,yc),1} = 1, \eta^1_{(xc,yc),2} = -4, \eta^1_{(xc,yc),3} = -1$, etc.

Then, PPEP is defined as an array

$$\xi_{(xc,yc),r} = [\eta^1_{(xc,yc),I} \otimes \varphi_{(xc,yc),r}^2 \otimes \varphi_{(xc,yc),r}^3 ... \varphi_{(xc,yc),r}^R].$$

Every value of $\xi_{(xc,yc),r}$ is the relative orders of the corresponding isoline-like sets in the original image and the gradient images. Clearly, $\eta^1_{(xc,yc),I}$ consists of $24$ relative orders, $\eta^2_{(xc,yc),I}$ consists of $48$ relative orders and $\eta^R_{(xc,yc),I}$ consists of $24r$ relative orders. Thus, $\xi_{(xc,yc),r}$ consists of $12r(1 + r)$ relative orders. Take $9 \times 9$ matching window with $R = 4$ as an example, $\xi_{(xc,yc),r}$ consists of $240$ relative orders.

### 2.2 Adaptively weighted cost

The relative orders are obtained from the different isoline-like sets with different radii. In order to get better stereo matching results, the influences of radii should be considered. An adaptively weighted cost for stereo matching is proposed. Let $I_{\text{left}}$ and $I_{\text{right}}$ be the left and right images of a stereo image pair, $d_{\text{max}}$ be the maximum disparity and $d$ be a disparity where $d \in \{0,1,...,d_{\text{max}}\}$. Let $p_i$ be the corresponding pixel of $p_i$ in $I_{\text{right}}$ with a disparity $d$ and $x_i$ and $y_i$ are its coordinates. Then, the adaptively weighted cost is defined as

$$C(x_i,y_i,d) = \sum_r \exp\left(-\frac{r}{R}\right) \sum_i |\eta^r_{(xc,yc),i} - \eta^r_{(xc,-d,yc),i}|,$$

where $r \in \{1,2,...,R\}$ and $\eta^r_{(xc,yc),i}$ and $\eta^r_{(xc,-d,yc),i}$ are the pixel $p_i$ and $p_{i'}$‘s values at index $i$ in $\eta^r_{(xc,yc),I}$ and $\eta^r_{(xc,-d,yc),I}$ respectively. $\eta^r_{(xc,yc),I}$ is a part of $\xi_{(xc,yc),r}$. In (8), $|\eta^r_{(xc,yc),i} - \eta^r_{(xc,-d,yc),i}|$ computes the dissimilarity between the relative orders of the pixel in the left image and its corresponding pixel in the right image. The adaptive weights $\exp\left(-\frac{r}{R}\right)$, which are related to the radii $r$ of the isoline-like sets, are used to provide different contributions of the dissimilarities. The gestalt grouping by proximity [28] means that the point adjacent to the centre would get more attention in reality. The design of the adaptive weights follows the rule of the gestalt grouping by proximity.

The procedure of constructing costs $C$ of our algorithm is shown in Algorithm 1.

### 3 EXPERIMENTS

Except the CNN-based methods, all of our experiments are made by a PC platform with an Intel core i5, 3.00-GHz CPU, and 8.00 GB of memory. The experiments of CNN-based
methods are made by a GPU platform with GTX Titan X. For comparisons, the size of the matching windows is set as $9 \times 9$ in the experiments like FEP \cite{12}. The methods are evaluated using the bad pixel error rate of each image in non-occluded areas. It is computed as

$$
\varepsilon = \frac{100}{|I_{\text{noccl}}|} \sum_{j \in I_{\text{mic}}} E(p_j),
$$

(9)


**Algorithm 1** Procedure of PPEP for Stereo Matching via Adaptively Weighted Cost

1: Compute PPEP for each $p_r$ in $I_{left}$ using Equation (7)
2: Compute PPEP for each $p_r$ in $I_{right}$ using Equation (7)
3: Compute the adaptive weights of the isoline-like sets’ relative orders
4: Compute the matching cost for each pixel $p_r$ in $I_{left}$
5: for each pixel $p_r$ in $I_{left}$ do
   6: for $d = 0$ to $d_{max}$ do
      7: Compute $C(n, j, d)$ using Equation (8)
   8: end for
9: end for

where $I_{nooc}$ is the set of all non-occluded pixels, $|I_{nooc}|$ is the number of pixels in $I_{nooc}$ and $E(p_r)$ is defined as

$$ E(p_r) = \begin{cases} 
  0, & \text{if } |D_E(p_r) - D_C(p_r)| \leq \zeta \\
  1, & \text{otherwise}
\end{cases} \quad (10) $$

where $D_C(p_r)$ and $D_E(p_r)$ are the ground truth and estimated disparity at $p_r$, respectively. The parameter $\zeta$ is the error threshold, which is set as 1 and 4 for the Middlebury 2006 and 2014 data set [16], respectively, and 3 for the KITTI data set [17].

### 3.1 Data set selection

The Middlebury 2006, Middlebury 2014 and KITTI 2015 data sets are used to evaluate the performance of the proposed method. Typically, the Middlebury 2006 data set, which consists of the indoor stereo images collected under controlled conditions with different illuminations and exposures, is used to establish image pairs with various radiometric conditions. These image pairs are used to investigate the performance of the proposed method under radiometric distortions. In details, the illuminations are indexed as 1, 2, and 3, and the exposures are indexed as 0, 1, and 2. The 15 data subsets, including Aloe, Art, Baby1, Baby2, Baby3, Cloth1, Cloth2, Cloth3, Cloth4, Dolls, Moebius, Rocks1, Rocks2, Wood1, and Wood2, are used. In the experiments, the left image with illumination 1 and exposure 0 and the right image with illumination 2 and exposure 1 are represented as 10_21. For each of the data subsets, 18 configurations, which the stereo images have different exposures (10_11, 10_12, 20_21, 20_22, 30_31, and 30_32), different illuminations (10_20, 10_30, 11_21, 11_22, 11_31, 12_22, and 12_32), and both different exposures and illuminations (10_21, 10_32, 20_11, 20_32, 30_11, and 30_22), are used. For each data subset with 18 configurations, the average of bad pixel error rates would be evaluated. Also, the latest 15 images of the Middlebury 2014 data set, which include Adirondack, ArtL, Jadeplant, Motorcycle, MotorcycleE, Piano, PianoL, Pipes, Playroom, Playtable, PlaytableP, Recycle, Shelves, Teddy and Vintage, are used.

In addition, the KITTI 2015 data set, which is consisted of the real road-driving images, is different from the indoor stereo images in the Middlebury data sets. It is necessary to evaluate the proposed method under the KITTI 2015 data set. Since the noise interference is existed in outdoor shooting, it should be not easy to get robust disparity results under the outdoor circumstances. To evaluate the proposed method, 200 stereo image pairs (indexed from 0 to 199) in training set in which the ground truth images were provided, are used for our experiments.

### 3.2 The methods selected in the experiments

Obviously, matching cost plays an important role in stereo matching. When the matching cost is decided, local and global algorithms can be used to obtain more accurate disparities. Local algorithms perform local aggregation and compute the disparities by winner-taken-all strategy. Global algorithms optimize the cost functions and directly get the disparities. In our experiments, the investigation of PPEP’s parameter setting and adaptively weighted cost will not use the local algorithms and global algorithms. Otherwise, in order to make solid comparisons, it is necessary to conduct experiments that consist of the complete steps. The local algorithm and global algorithm will be used, respectively. In our experiments, a local algorithm named Guide-Filter (GF) [29] or a global algorithm named Graph-Cut (GC) [5] is combined with matching cost algorithms. GF and GC are the most widely used algorithms. When the matching cost algorithms are combined with GF or GC, the parameters for each method are fine-tuned to get the best results. By the way, all of the experimental results do not involve any post-processing steps.

### 3.3 Investigations of PPEP’s parameter setting and adaptively weighted cost

The Middlebury 2006 data set with various radiometric conditions mentioned above is used to investigate PPEP’s parameter setting and adaptively weighted cost. In order to avoid the effects of local algorithms or global algorithms, PPEP does not combine any stereo matching algorithms and gets the disparity results directly in this section.

1. **Investigation of PPEP’s parameter setting:** The parameter $m$ in (2) is the only parameter of PPEP. It affects the size of the interval of function (2). Experiments with different $m$ settings are conducted. As shown in Table 1, PPEP with $m = 25$ gets the best results. The value of $m$ should not be too big or too small experimentally, $m$ with small value could get small size of the interval. Then, intensity difference of two pixels may be represented more accurate. However, very small value of $m$ can lead the method to be influenced by noise easily since slight changes of matching pixels’ intensities can let the relative orders be different values. Then, $m$ with big value can let the function be robust to noise, but it may affect the accuracy. The value of $m$ can balance the accuracy and robustness. Note that in the subsequent experiments, $m$ is set as 25.

2. **Investigation of the adaptively weighted cost:** The adaptive weights are used in function (8). To verify (8)’s...
3.4 Longitudinal comparative analysis of PPEP

In this section, the effects of PPEP’s every part are discussed. PPEP could be summarised into two parts: encoding pattern and encoding function. In order to analyse the improvement of every part of PPEP, Census [7] is selected as the basic algorithm for comparison. On the basis of Census, the two parts of PPEP are gradually introduced. Table 3 shows the matching results on the Middlebury 2006 data set with different configurations of illuminations and exposures when these methods are combined with GC. As listed in Table 3, the error rates are 13.70%, 11.38% and 10.57%, respectively. The two parts of PPEP could improve the performance effectively. The encoding pattern in the proposed algorithm is the most important part. It gets 2.32% improvement while the encoding function gets 0.81% improvement. We consider that the encoding pattern is the core of PPEP, while the encoding function gets improvement on the basis of encoding pattern. To some extent, the encoding function could improve matching accuracy.

3.5 Comparisons between AD, Census, FEP and PPEP

The authors in paper [12] compared the performances of fuzzy encoding pattern (FEP) and other conventional stereo matching cost algorithms such as Census Transform (Census) [7], AD [6], ANCC [9], mutual information-based cost (MI) [10], MI and sift descriptor (MIS) [30], and support LBP (SLBP) [11]. FEP [12] had showed its improvements. Then, PPEP in our experiments will directly compare with FEP. Also, since AD and Census are classical algorithms and are widely used in many applications, PPEP is also compared with AD and Census.

(1) Middlebury data set: We evaluate above matching cost algorithms using the local and global methods on the Middlebury 2006 and Middlebury 2014 data set. Tables 4 and 5 show the evaluation results of the above algorithms using GF and GC, respectively. AD, which directly uses pixels’ intensity values, is the simplest algorithm. As listed in Tables 4 and 5, AD gets worst results, especially on the Middlebury 2006 data set, which includes image pairs with various radiometric conditions. Comparing with Census using GF and GC respectively, FEP gets 1.86% and 2.27% improvements on the Middlebury 2006 data set. However, FEP performs worse than Census on the

---

**TABLE 1** Average bad pixel error rates of PPEP with different m settings in non-occluded areas using the Middlebury 2006 data set with different configurations of illuminations and exposures

| Data set | m = 5 | m = 15 | m = 25 | m = 35 | m = 45 |
|----------|-------|--------|--------|--------|--------|
| Aloe     | 38.48%| 25.14% | 18.10% | 19.54% | 20.74% |
| Art      | 45.68%| 39.46% | 34.54% | 36.21% | 37.54% |
| Baby1    | 40.0% | 25.27% | 18.95% | 20.10% | 20.76% |
| Baby2    | 46.93%| 33.33% | 28.76% | 29.96% | 30.67% |
| Baby3    | 52.25%| 39.50% | 34.70% | 36.08% | 37.05% |
| Cloth1   | 17.70%| 9.82%  | 6.83%  | 7.87%  | 8.12%  |
| Cloth2   | 22.78%| 14.20% | 10.97% | 12.25% | 12.92% |
| Cloth3   | 28.45%| 17.70% | 10.84% | 12.24% | 13.62% |
| Cloth4   | 18.46%| 13.81% | 11.64% | 12.77% | 13.20% |
| Dolls    | 39.52%| 34.17% | 28.90% | 30.51% | 31.89% |
| Moebius  | 37.60%| 33.16% | 30.43% | 31.76% | 32.45% |
| Rocks1   | 40.42%| 22.98% | 17.06% | 18.44% | 19.61% |
| Rocks2   | 38.56%| 20.24% | 15.16% | 16.50% | 17.50% |
| Wood1    | 42.45%| 32.72% | 30.96% | 32.11% | 32.48% |
| Wood2    | 39.83%| 36.18% | 35.44% | 36.50% | 36.68% |
| Rate     | 36.61%| 26.38% | 22.22% | 23.32% | 24.35% |

The all bold values appearing in tables are used to highlighted the data with the lowest error rates.

**TABLE 2** Average bad pixel error rates of PPEP using adaptively weighted cost and using cost without weight in non-occluded areas on the 15 Middlebury 2006 data set with different configurations of illuminations and exposures

| Method | Adaptively weighted cost | Cost without weight |
|--------|--------------------------|---------------------|
| Rate   | 22.22%                   | 27.41%              |

The all bold values appearing in tables are used to highlighted the data with the lowest error rates.

---
TABLE 5  Average bad pixel error rates of AD, Census, FEP, PPEP with GC in non-occluded areas using Middlebury 2006 data set and Middlebury 2014 data set

| Data set     | AD    | Census | FEP    | PPEP  |
|--------------|-------|--------|--------|-------|
| Middlebury 2006 | 77.52% | 13.70% | 11.43% | 10.08%|
| Middlebury 2014 | 22.54% | 15.9%  | 18.5%  | 14.7% |

The all bold values appearing in tables are used to highlighted the data with the lowest error rates.

Middlebury 2014 data set. We can see that FEP gets 1.0% and 2.6% reductions than Census using GF and GC on the Middlebury 2014 data set, respectively. PPEP performs better than other matching cost algorithms used in the experiments. On the Middlebury 2006 data set, PPEP gets 2.1% and 1.35% improvements than FEP using GF and GC, respectively. On the Middlebury 2014 data set, PPEP gets 1.4% and 1.2% improvements than Census using GF and GC, respectively. PPEP performs robustly on the different versions of the Middlebury data sets. On the basis of Census, FEP considers the encoding information between local pixels on the matching windows. Also, it uses FEF to obtain relative orders. It gets better results on the Middlebury 2006 data set with severe radiometric distortions. However, FEP is not robust enough and gets worse results than Census and PPEP on the Middlebury 2014 data set. Comparing with FEP, PPEP gets more robust and accurate results. In FEP, the two non-adjacent pixels, which are far away on coordinates, are used to get encoding values. The encoding pattern of PPEP is different. It only focuses on the adjacent pixels in each small local area. We consider that adjacent pixels are more likely to get similar changes under radiometric distortions. Then, the encoding values between adjacent pixels would have better invariance. The encoding features of PPEP would be robust relatively. The encoding pattern could be one of the reasons that PPEP gets more robust results than FEP. In addition, the matching cost computation of PPEP considers different weights of the relative orders with different radii while FEP does not consider this point. We consider that the adaptive weights are helpful to improve accuracy of matching results. By the way, Figure 5 shows the test results of the Motorcycle stereo images with different exposures and illuminations by GC. It can see that our proposed method PPEP gets better matching result (has less black areas).

(2) KITTI data set: we evaluate above matching cost algorithms on the stereo pairs on the KITTI 2015 data set. Tables 6 and 7 show the results of comparisons in non-occluded areas using GF and GC, respectively. The average error rate of the AD is still much higher than other algorithms. Obviously, AD is not suitable for complex situations. On the KITTI 2015 data set, FEP performs better than AD and Census. The error rates of FEP using GF and GC are 10.92% and 8.01%, respectively. It gets 0.72% and 0.67% improvements than Census using GF.

FIGURE 5  Results of Motorcycle stereo images with different exposures and illuminations by GC. (a) Left image. (b) Right image. (c) Ground truth. (d) Disparity map of Census ($\varepsilon = 7.34\%$). (e) Disparity map of FEP ($\varepsilon = 9.85\%$). (f) Disparity map of PPEP ($\varepsilon = 6.36\%$). (The grey areas mean the occluded areas, and the black areas mean the bad pixels of the estimated disparity map)

TABLE 6  Average bad pixel error rates of AD, Census, FEP, PPEP with GF in non-occluded areas using the KITTI data set

| Method | AD    | Census | FEP    | PPEP  |
|--------|-------|--------|--------|-------|
| Rate   | 34.82% | 11.64% | 10.92% | 10.43%|

The all bold values appearing in tables are used to highlighted the data with the lowest error rates.
and GC, respectively. We can also see that PPEP has the best performances of 10.43% with GF and 7.54% with GC. It gets 0.49% and 0.47% improvements than FEP using GF and GC, respectively. Overall, PPEP has the best effect on all data sets tested in the experiments.

3.6 | Comparisons between PPEP and CNN-based methods

In stereo matching, deep learning techniques are widely used and get state-of-the-art results. Deep learning based methods require sufficient training set to train for the specific scenario. The neural networks could extract informative and representative features by using typical network structures. CNN-based methods could get excellent results on the testing set under the condition of having suitable and sufficient training data set in the same domain. The accurate model of MC-CNN [15] is trained by the Middlebury data set except the images with different configurations of illuminations and exposures and is named as MCMB. It is combined with GC-like PPEP and is tested by the Middlebury images with different configurations of illuminations and exposures. MCMB has the same type of training set and testing set. The total average error rate of MCMB in non-occluded areas is 8.38%, while PPEP is 10.08% in our experiment. MCMB performs better than PPEP averagely. However, CNN-based methods rely heavily on the specific data set, they would be over-specialising in training domain and are not easy to transfer to unfamiliar scenes. When the training set and the testing set belong to different types, the performances of CNN-based methods might deteriorate heavily, whereas traditional methods do not need to train and may be suitable for a wide range situations.

In order to further explore the generalisation ability of CNN-based methods when the training set and the testing set belong to different types, the related experiments are conducted between the proposed method and the CNN-based methods. In the comparison, all of the CNN-based methods would be trained by the KITTI 2015 data set and would be tested by the Middlebury images with different configurations of illuminations and exposures. The comparison includes three CNN-based methods: MCKI [15], EDLSM [22] and CNNF [25]. The CNN-based method MCKI also uses the accurate model of MC-CNN like MCMB and the difference between them is the setting of training data set. EDLSM and CNNF are the CNN-based methods trained for feature point matching and could be used to get the matching cost like MCKI. For MCKI and CNNF, the source code and the pre-trained models on the KITTI 2015 data set are used. Using Tensorflow, EDLSM is implemented based on the framework and parameters as in [22].

As in Section 3.5, the global method GC [5] is used with CNN-based methods directly. Table 8 shows the results of PPEP, MCKI, CNNF and EDLSM. The results of the CNN-based methods are worse than PPEP when the training set and the testing set belong to different types. We consider that these CNN-based methods depend on the training set severely and occur the problem of generalisation. In the cases of unsuitable and insufficient training sets, the performances of CNN-based methods might deteriorate heavily. Conversely, PPEP could be a better option in the absence of sufficient samples. PPEP could be applied to any real scenarios directly without training.

3.7 | Tests of the running time

In this section, the running time of all of the methods in the comparisons are tested. As shown in Table 9, Census needed only 0.51 s to construct the disparity image space on a CPU with one core. It is the simplest method and performs faster than other methods. FEP and PPEP needed 5.42 s and 3.41 s respectively. PPEP performs faster than FEP. The reason could

### Table 7

| Method | AD | Census | FEP | PPEP |
|--------|----|--------|-----|------|
| Rate   | 31.00% | 8.68% | 8.01% | 7.54% |

The all bold values appearing in tables are used to highlighted the data with the lowest error rates.

### Table 8

| Data set | PPEP | MCKI | CNNF | EDLSM |
|----------|------|------|------|-------|
| Aloe     | 10.39% | 13.03% | 26.89% | 33.33% |
| Art      | 21.66% | 23.38% | 32.90% | 55.25% |
| Baby1    | 4.54%  | 5.47%  | 10.55% | 20.11% |
| Baby2    | 9.88%  | 9.85%  | 12.99% | 27.74% |
| Baby3    | 15.86% | 20.12% | 22.84% | 33.15% |
| Cloth1   | 1.97%  | 5.36%  | 17.42% | 10.73% |
| Cloth2   | 5.81%  | 11.78% | 12.95% | 20.82% |
| Cloth3   | 4.02%  | 7.00%  | 13.24% | 31.18% |
| Cloth4   | 5.24%  | 8.70%  | 13.46% | 33.30% |
| Dolls    | 15.78% | 20.38% | 25.41% | 46.72% |
| Moebius  | 18.11% | 20.01% | 29.18% | 47.33% |
| Rocks1   | 5.75%  | 9.43%  | 15.97% | 36.49% |
| Rocks2   | 4.74%  | 9.83%  | 15.36% | 37.76% |
| Wood1    | 12.63% | 11.35% | 18.76% | 46.80% |
| Wood2    | 14.86% | 15.66% | 18.54% | 36.74% |
| Rate     | 10.08% | 12.96% | 19.10% | 34.50% |

The all bold values appearing in tables are used to highlighted the data with the lowest error rates.

### Table 9

| Method | Census | FEP | PPEP | MCKI | CNNF | EDLSM |
|--------|--------|-----|------|------|------|-------|
| Runtime (s) | 0.51 | 5.42 | 3.41 | 83.54 | 3.07 | 3.64 |

The all bold values appearing in tables are used to highlighted the data with the lowest error rates.
be that for 9 × 9 matching window case, PPEP uses 240 relative orders, but FEP uses 408 relative orders. For the CNN-based methods, MCKI, EDLSM and CNNF needed 83.54 s, 3.07 s and 3.64 s, respectively. The PPEP has running time closed to that of the CNNF and EDLSM. The MCKI has the longest running time among all the methods used in the experiments. In the Table 8, comparing with other CNN-based methods, the error rate of MCKI is closest to PPEP. However, MCKI has low running efficiency. Consider both accuracy as showed in Table 8 and the running time in this section, PPEP could be better applied to stereo matching tasks.

4 | CONCLUSION

PPEP is proposed for stereo matching. Based on the pixel pairs, it performs robustly under radiometric distortions. Also, the new encoding function and the adaptively weighted cost are used to improve accuracy. The experimental results show that the proposed method is superior to some widely used methods in the average of bad pixel error rate. Also, PPEP performs faster than FEP in our experiments. In our future work, we will focus on the accuracy and computational complexity of the stereo matching caused by the encoding.

ACKNOWLEDGMENTS

This work is partially supported by Natural Science Foundation of Guangdong Province (2019A1515010861, 2018A030 313869), Guangzhou Technical Project (Grant: 201902020008), and NSFC (Grant: 61471174).

REFERENCES

1. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J Comput. Vision 47(1), 7–42 (2002)
2. Zhan, Y., et al.: Accurate image-guided stereo matching with efficient matching cost and disparity refinement. IEEE Trans. Circuits Syst. Video Technol. 26(9), 1632–1645 (2016)
3. Yang, Q.: Stereo matching using tree filtering. IEEE Trans. Pattern Anal. Mach. Intell. 37(4), 834–846 (2015)
4. Mei, X., et al.: Segment-tree based cost aggregation for stereo matching. IEEE Conference on Acoust Speech Signal Process, New Orleans, LA, pp. 313–320. IEEE, Piscataway, NJ (2017)
5. Boykov, Y., et al.: Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. 23(11), 1222–1239 (2001)
6. Hirschmuller, H., Scharstein, D.: Evaluation of stereo matching costs on images with radiometric differences. IEEE Trans. Pattern Anal. Mach. Intell. 31(9), 1582–1599 (2009)
7. Zabih, R., Woodfill, J.: Non-Parametric Local Transforms for Computing Visual Correspondence. In: Ecklundh, J.O. (ed.) European Conference on Computer Vision, pp. 151–158. Stockholm, Sweden (1994)
8. Kanade, T.: Development of a video-rate stereo machine. Proceedings 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots, August 1995, pp. 549–557. IEEE, Piscataway, NJ (1995)
9. Heo, Y.S., et al.: Robust stereo matching using adaptive normalized cross-correlation. IEEE Trans Pattern Anal. Mach. Intell. 33(4), 807–822 (2011)
10. Hirschmuller, H.: Stereo processing by semiglobal matching and mutual information. IEEE Trans Pattern Anal. Mach. Intell. 30(2), 328–341 (2008)
11. Nguyen, V.D., et al.: Support local pattern and its application to disparity improvement and texture classification. IEEE Trans. Circuits Syst. Video Technol. 24(2), 263–276 (2014)
12. Dinh, V.Q., et al.: Fuzzy encoding pattern for stereo matching cost. IEEE Trans. Circuits Syst. Video Technol. 26(7), 1215–1228 (2016)
13. Quang, V., et al.: Matching cost function using robust soft rank transformations. IET Image Process. 10(7), 561–569 (2016)
14. Jiao, J., et al.: Joint image denoising and disparity estimation via stereo structure PCA and noise-tolerant cost. Int. J Comput. Vision 123(4), 1–19 (2017)
15. Bontar, J., Lecun, Y.: Stereo matching by training a convolutional neural network to compare image patches. J. Mach. Learn. Res. 17(1), 2287–2318 (2018)
16. Middlebury data set, http://vision.middlebury.edu/stereo/data/. Accessed September 2014
17. Kitti data set. http://www.cvlibs.net/datasets/kitti/. Accessed November 2015
18. Lincheng, Li., et al.: PMSC: Patchmatch-based superpixel cut for accurate stereo matching. IEEE Trans. Circuits Syst. Video Technol. 28(3), 679–692 (2018)
19. Taniai, T., et al.: Continuous 3D label stereo matching using local expansion moves. IEEE Trans. Pattern Anal. Mach. Intell. 40(11), 2725–2739 (2018)
20. Lim, J., Lee, S.: Patchmatch-based robust stereo matching under radiometric changes. IEEE Trans. Pattern Anal. Mach. Intell. 41(5), 1203–1212 (2019)
21. Yan, T., et al.: Segment-based disparity refinement with occlusion Handling for stereo matching. IEEE Trans. Image Process. 28(8), 3885–3897 (2019)
22. Luo, W., Schwing, A.G., Utsasun, R.: Efficient deep learning for stereo matching. IEEE Conference on Computer Vision and Pattern Recognition, June 2016, Las Vegas, NV, pp. 5695–5703. IEEE, Piscataway, NJ (2016)
23. Park, H., Lee, K.M.: Look wider to match image patches with convolutional neural networks. IEEE Signal Process. Lett. 24(2), 1788–1792 (2017)
24. Yang, M., Lv, X.: Learning both matching cost and smoothness constraint for stereo matching. Neurocomputing 314(7), 234–241 (2018)
25. Zhang, F., Wah, B.W.: Fundamental principles on learning new features for effective dense matching. IEEE Trans. Image Process. 27(2), 822–836 (2018)
26. Dong, H., Wang, T.: Stereo matching via dual fusion. IEEE Signal Process. Lett. 25(5), 615–619 (2018)
27. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. IEEE Conference on Computer Vision and Pattern Recognition, June 2005, San Diego, CA, pp. 1063–6919. IEEE, Piscataway, NJ (2005)
28. Yoon, K.-J., Kweon, I.S.: Adaptive support-weight approach for correspondence search. IEEE Trans. Pattern Anal. Mach. Intell. 28(4), 650–656 (2006)
29. Hosni, A., et al.: Fast cost-volume filtering for visual correspondence and beyond. IEEE Trans. Pattern Anal. Mach. Intell. 35(2), 504–511 (2013)
30. Heo, Y.S., Lee, K.M., Lee, S.U.: Joint depth map and color consistency estimation for stereo images with different illuminations and cameras. IEEE Trans. Pattern Anal. Mach. Intell. 35(5), 1094–1106 (2013)

How to cite this article: Fu Y, Lai K, Chen W, Xiang Y. A pixel pair–based encoding pattern for stereo matching via an adaptively weighted cost. IET Image Process. 2021;15:908–917. https://doi.org/10.1049/ipt2.12071