



Joint depth map interpolation using minimax paths

Longquan Dai, Feihu Zhang, Mengke Yuan, Xiaopeng Zhang*

National Laboratory of Pattern Recognition, Institute of Automation Chinese Academy of Sciences, 95 Zhongguancun East Road, Beijing, China

ARTICLE INFO

Article history:

Received 29 July 2014

Received in revised form

27 March 2015

Accepted 28 March 2015

Communicated by X. Gao

Available online 7 April 2015

Keywords:

Depth map

Joint interpolation

Minimax path

ABSTRACT

We propose a minimax path-based depth interpolation method. The algorithm computes for each target pixel varying contributions from reliable depth seeds, and weighted averaging is used to interpolate missing depths. Compared with state-of-the-art joint geodesic upsampling method which selects the K nearest seeds to interpolate missing depths with $O(Kn)$ complexity, our method does not need to limit the number of seeds to K and reduces the computational complexity to $O(n)$. In addition, the minimax path chooses a path with the smallest maximum immediate pairwise pixel difference on it, so it tends to preserve sharp depth discontinuities better. In contrast to the results of previous depth upsampling algorithms, our approach provides accurate depths with fewer artifacts.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

Recently, low-cost depth sensors such as ToF Camera and Microsoft Kinect have been widely used in various applications and gradually are shaping new forms of man–machine interaction. However, limited by current depth sensing techniques, the depth map suffers from various problems and thus the quality is not as good as the color image captured by optical camera. Specifically, the resolution of a depth image is very low and some parts of the depth map are lost due to occlusion or other degradation. For instance, swiss ranger SR-4000 can only obtain depth maps with resolution 176×144 and PMD technologies camcube 2.0 only get depth maps with resolution 204×204 . Although Kinect can provide depth maps with resolution up to 640×480 , the maximum measuring distance is only 3.5 m. Recently, the second Kinect makes an improvement, but the maximum measuring distance is still limited to only 4.5 m. However, in many digital imaging applications, high resolution images are usually desired for later image processing and analysis. The appetency for high image resolution stems from two principal application areas: improvement of pictorial information for human interpretation; and helping representation for automatic machine perception, because image resolution describes the details contained in an image, the higher the resolution, the more the image details. Another reason to upscale the low resolution depth map is that in 3D video applications, coding of depth map at reduced resolution provides better rate-distortion performance than the utilization of

full-resolution depth maps in many coding scenarios. For that, the depth map can be down-sampled at the encoder before compression to save network bandwidth. However, the down-sampled depth map is required to be up-sampled at the decoder to provide accurate depth information for synthesizing the viewpoint video. In order to obtain a high-quality depth image, we need to both upscale the resolution of the depth maps and restore the missing depths. Microsoft Kinect raises another depth map processing problem. Although Kinect sensor can provide high-resolution depth map, it suffers from the missing depth information. This is because the structured light emitted from Kinect is absorbed by the black surfaces of objects in the scene. We thus need an inpainting technique to interpolate the missing depths according to the depths in the neighborhood. Briefly, the depth upsampling and inpainting tasks are special cases of depth interpolation as depth interpolation employs a set of sparse seeds to compute the missing depths of corrupted depth maps. Here, seeds stand for the pixels with observed depths and target pixels denote the pixels whose depths are lost. In this paper, we propose a novel fast joint depth interpolation algorithm for these interpolation tasks.

One possible way to produce high quality interpolation result with perceptual consistent edges is to interpolate the missing depths under the guidance of a color image I which is in tandem with the depth sensor. The joint interpolation approach is reasonable as abrupt color transition often leads to abrupt depth transition. So we calculate the depths of target pixels according to the depths of seeds under the guidance of a registered color image for a damaged depth map. Specifically, in our algorithm, we extract all pairs minimax paths from the registered guidance image I , and map all of them onto the target depth map D to indicate the geometric structure of D . After that, we interpolate the missing depths along the minimax paths.

* Corresponding author. Tel./fax: +86 10 82544696.

E-mail addresses: lqdai@nlpr.ia.ac.cn (L. Dai), hiyexu@gmail.com (F. Zhang), mengkeyuan@foxmail.com (M. Yuan), xpzhang@nlpr.ia.ac.cn (X. Zhang).

Our method is inspired by two recent proposed interpolation methods [1,2]. The joint geodesic upsampling [1] of Liu, which is state-of-the-art method in terms of both accuracy and efficiency, employs an approximate geodesic distance as the similarity metric and chooses the K nearest samples to interpolate missing depths. In the paper [1], Liu illustrated that the upsampling accuracy increases with K . However, the computational complexity of the method is $O(Kn)$ which is also proportional to K , where n denotes the number of pixels of the target depth map. Moreover, Liu proved that the accurate interpolation result needs more reliable seeds. Hence, the running time of his algorithm will vary with the quality of interpolation results. In contrast, the running time of Yang's non-local cost aggregation [2] is $O(n)$ which implies that the running cost of the method is independent of the number of aggregated points. In the seminal paper [2], Yang claimed that every node receives supports from all other nodes. According to the statement, we conclude that Yang's method cannot distinguish interpolated points (i.e. target pixels) from interpolating seeds and thus it has no idea of the reliability of the interpolating seeds. If we directly apply this method to perform interpolating, the final result will be contaminated by the undefined values of interpolated points and the depths of unreliable seeds. We argue that an ideal method should be capable of exploiting the depths of all the reliable seeds and the computational complexity should be linear time and has nothing to do with the number of interpolating seeds K . Moreover, an ideal method should also be capably applied to various interpolation tasks besides the classic upsampling task with the uniform seed distribution. In the paper, we are going to build the ideal interpolation algorithm by both utilizing the minimax distance that aggregates the edge lengths of the minimax path on the guidance image I 's graph representation and employing the reliable seeds defined by the minimal spanning tree (MST) [3].

The minimax path [4] is the curve with the smallest intermediate length of the longest edge. If there are two or more seeds on the same minimax path, it is reasonable to consider that the depth of the nearest seed is more reliable than the other seeds. Hence, in our interpolation scheme, we take the nearest seed as one of the reliable seeds of the target pixel. The remainder seeds on the minimax path are treated as unreliable seeds. In other words, we only take the depths of the target pixel's reliable seeds to compute its missing depth for accurate interpolation. Since the minimax path is the path with the smallest maximum immediate pairwise pixel difference on it, the path is sensitive to the underlying structure of the guidance image and can follow the regularity of the guidance image I . More importantly, the path does not cross the color edges of I , which are typically also the depth boundaries of D . Otherwise, the crossing boundary edge will increase the maximal length of edges in the path. Hence, the minimax path is able to detect color boundaries that do not have high contrast and it also does not go through the boundaries of thin segments which have uniform colors and therefore can preserve fine details. Interpolating along the minimax paths that link seeds and target pixels, we can keep the sharp depth edges that coincide with the color edges of the guidance image. Furthermore, the length of the minimax path (i.e. minimax distance) can be used to calculate the contribution of each reliable seed for the connected target pixel in a geometric aware manner.

Interpolating with minimax paths involves extracting all pairs minimax paths because we need to calculate the minimax distance and interpolating along these paths. Seemingly, it is a time consuming task since the computational complexity of extracting all pairs geodesic paths is $O(n^2 \log n)$ [5]. Indeed, there is an efficient way to complete the work in the literature. In 1961, Hu [6] reduced the task of extracting all pairs minimax paths to the problem of finding an MST. Additionally, it is worth mentioning

that there are lots of efficient algorithms to find an MST in the literature. Specifically speaking, the first algorithm for finding a minimum spanning tree was proposed by Czech scientist Otakar Boruvka in 1926 [7]. The two commonly used algorithms published in 1956 are Prim's algorithm [8] and Kruskal's algorithm [9]. All three are greedy algorithms that run in polynomial time. Recently, Karger et al. [10] discovered a linear time randomized algorithm for a single processor to find an MST in 1995. Six years later, Chong et al. [11] proposed a more efficient parallel algorithm with $O(\log n)$ complexity for the multicore computer.

2. Related work

The most related works are the depth upsampling methods. Roughly speaking, these upsampling methods can be categorized as global methods and local methods. Global methods [12–14] utilize the optimization framework that punishes a large cost for the coupled pixels which have similar colors but different depths. Although global methods usually produce high-quality upsampling results, the computational cost is too high to apply it in real-time processing. Instead, people usually adopt a local weighted average based scheme to design efficient interpolation algorithms. The optimization based global methods are just used as an auxiliary technology to achieve the high-quality upsampling results that cannot be produced by the more simple and efficient filtering based local methods. In the literature, Diebel and Thrums [12] proposed the first optimization based upsampling method by employing an MRF formulation, where the smoothness term is weighted according to texture derivatives. A more complex approach was proposed by Park et al. [13]. They employed a combination of different weighting terms of a least squares optimization including segmentation, image gradients, edge saliency and non-local means for depth upsampling in 2011. One year later, Yang et al. [14] formulated the depth recovery task as finding the minimizer of AR prediction errors and thus proposed an adaptive color-guided auto-regressive (AR) model that subjects to measurement consistency for high quality depth recovery from low quality measurements. In the same year, Li et al. [15] combined the intensity and depth data in a Bayesian Framework for depth upsampling. Recently, Kang et al. [16] extracted depth discontinuity and depth variance from a low-resolution depth map and a color image as the depth local features and then incorporated them into an energy function for the Markov Random Field based depth upsampling. The high-resolution depth map is obtained by optimizing the energy function using belief propagation. To prevent scene texture from being erroneously transferred to reconstructed scene surfaces, Choi et al. [17] proposed a framework for dividing the color image into different regions and applying different methods tailored to each region type for depth upsampling. Dai et al. [18] formulate the upsampling task as a sparse signal recovery problem that solves an underdetermined system. With a reference color image, the low-resolution depth map is converted into suitable sampling data (measurements). Unlike previous methods, Yang et al. [19] taken a slightly different strategy which acquires a high resolution depth map by a given pair of color images and a low-resolution depth map as inputs.

Local methods [1,20–23] are based on the weighted average scheme introduced by Kopf et al. [21] and often have fast implementations. Our approach also falls into this category. In the literature, Yang et al. [22] performed the bilateral filtering on the depth cost volume under the guidance of a RGB image in the iterative refinement process. Chan et al. [23] used a noise aware joint bilateral filter to increase the resolution and to reduce depth map errors at multiple frames per second. Dolson et al. [20] presented a flexible method for fusing information from optical

and range sensors based on an accelerated high-dimensional filtering approach. Recently, Liu et al. [1] proposed a fast joint geodesic upsampling method by utilizing geodesic distances to upscale a low resolution depth image in real time. Technically speaking, the novelty of local methods is determined by two factors: the reliable seed choosing strategy and the similarity measure of coupled nodes. In this paper, we introduce novel ideas to redesign the reliable seed choosing strategy as well as the similarity measure. All these improvements are significantly different from other local methods. In addition, we present an efficient interpolation implementation with $O(n)$ complexity. The experiments show that our novel interpolation method achieves the best results in terms of accuracy and efficiency.

Accurate interpolation results rely on the similarity metric heavily because interpolation methods employ it to measure the affinity between target pixels and seeds for calculating the contribution of seeds. In the literature, Euclidean distance [21] is the simplest similarity measure and is widely used in many applications. Nevertheless, it is not aware of the underlying geometric structure of the data points that lie on a curved manifold. So, on the graph representation of these points, people propose a set of computational intensive link-based distances such as the average-commute-time distance [24], the SoP covariance distance [25] and the geodesic distance [1] to conquer the weakness of Euclidean distance. Here, we prefer the minimax distance (i.e. the length of minimax path) [26] for reasons that there are efficient algorithms to extract all pairs minimax paths and the minimax path is a geometric-aware path. More importantly, the minimax distance allows for a fast interpolation algorithm.

2.1. Contribution

Generally speaking, the main contribution of our paper is a novel interpolation method with the following advantages:

- Our method uses the minimax distance, which is aware of the underlying regularity of the guidance image I , to measure the similarity of nodes.
- Our method exploits a novel reliable seed choosing strategy which can seize all the reliable seeds regardless of the seed distribution.
- Our method is a novel and high-efficient method which achieves $O(n)$ complexity and is independent of the number of reliable interpolating seeds.

3. Joint interpolation algorithm

Our joint interpolation scheme exploits a registered guidance image I to guide the interpolation on the target depth map D for preserving the sharp depth edges. To obtain accurate results, we calculate the depth of each target pixel by exclusively using the depths of the target pixel's reliable seeds. Specifically, in the following paragraphs, we will first investigate the difficulties of designing an ideal joint interpolation method. After that, the solutions to these difficulties will be presented. Finally, we will assemble these solutions together to composite our joint interpolation algorithm and offer an analysis of the computational complexity of the novel interpolation algorithm.

3.1. Essential ingredients of an ideal joint interpolation

We will distinguish what are the essential ingredients of an ideal joint interpolation method by bringing out the shortcomings of three well-known local upsampling methods [21,1,2]. In 2007, Kopf et al. [21] introduced the first weighted average based upsampling

formula (1) called joint bilateral filter. However, the user-specified square window Ω_p and the Euclidean distance d_e [21] are suboptimal to take advantage of the geometric structure of the guidance image I as neither Ω_p nor d_e can adapt itself to the underlying geometric structure of the guidance image I . As a remedy, Liu et al. [1] substituted the Euclidean distance d_e [21] with the geodesic distance d_g [1] and replaced the user-specified window selecting method with the K nearest seeds choosing strategy. However, the new interpolation method as formulated in Eq. (2) is still problematic for the reason that extracting all pairs geodesic paths is a time-consuming task with $\log(n^2 \log n)$ complexity [21]. In compensation for this, Liu et al. employed a iterative approximation geodesic distance which owns a fast extracting algorithm with $O(Mn)$ complexity. But accurate approximation demands M iterations. The method thus is still inefficient. In addition, Liu et al. showed that the quality of results increases with the number K of reliable seeds. Unfortunately, their $O(Kn)$ algorithm depends on K too. In contrast, Yang [2] offered a fast aggregation method with $O(n)$ complexity. Nevertheless, this method does not distinguish reliable seeds from unreliable seeds. Therefore, the depths of target pixels will receive the contributions from other pixels in the image domain Ω and the depths of seeds will be modified by other nodes. Moreover, the method employs an unnormalized weighted average scheme which is not suited to interpolate the depth map. In brief, the aggregation formula of Yang as illustrated in Eq. (3) is not appropriate for high quality interpolation too:

$$D(p) = \frac{\sum_{q \in \Omega_p \cap S} w_{p,q}^e D(q)}{\sum_{q \in \Omega_p \cap S} w_{p,q}^e}, \quad p \in \Omega \setminus S \quad (1)$$

$$D(p) = \frac{\sum_{q \in \mathcal{K}_p} w_{p,q}^g D(q)}{\sum_{q \in \mathcal{K}_p} w_{p,q}^g}, \quad p \in \Omega \setminus S \quad (2)$$

$$D(p) = \sum_{q \in \Omega} w_{p,q}^m D(q), \quad p, q \in \Omega \quad (3)$$

where $w_{p,q}^e = \exp(-\frac{d_e(p,q)}{\sigma_e})$, $w_{p,q}^g = \exp(-\frac{d_g(\pi_{p,q}^g)}{\sigma_g})$ and $w_{p,q}^m = \exp(-\frac{d_n(\pi_{p,q}^m)}{\sigma_m})$ are the weights, $d_e(p,q)$, $d_g(\pi_{p,q}^g)$ and $d_n(\pi_{p,q}^m)$ denote the Euclidean distance [21], the geodesic distance [1] and the length of MST path $\pi_{p,q}^m$ [2] respectively, Ω_p , \mathcal{K}_p and S stand for the user-specified window, the K nearest seed pixels of pixel p and the set of seeds respectively.

Overall, we argue that an ideal joint interpolation should have following properties:

- An ideal interpolation method should utilize a geometric-aware similarity metric, which is sensitive to the underlying geometric structure of guidance image, to weight the contributions from interpolating seeds. Furthermore, the affinity metric should also take on an efficiently computing algorithm.
- An ideal interpolation method ought to take advantage of a new reliable seed choosing strategy which can automatically catch all the reliable seeds for any sample distribution besides the uniform distribution used by upsampling.
- An ideal interpolation method should not just use the depths of reliable seeds to estimate the depths of target pixels, but also keep the depths of reliable seeds unchanged. Moreover, its implementation should be high-efficiency, especially for real-time applications, thus an algorithm with $O(n)$ complexity must be given the primary consideration.

3.2. The geometric aware minimax distance

Similar to the well-known geodesic distance, the minimax distance (i.e. the length of the minimax path) is another kind of

geometric-aware metric which can be used to measure the similarity between seeds and target pixels. Technically speaking, the minimax path is defined on the graph representation $G(V, E)$ (e.g. Fig. 2a) of the guidance image I , where each pixel of I corresponds to a node $p \in V$ and the first order neighborhood (v_1, v_2) of pixels v_1 and v_2 denotes an edge $e_{v_1, v_2} \in E$ with length $d(e_{v_1, v_2}) = \|I(v_1) - I(v_2)\|_1$. Let Π_{pq} denote all the possible paths that connect an initial node $p = v_{a_0}$ and a destination vertex $q = v_{a_m}$ such that $\{e_{v_{a_0}, v_{a_1}}, e_{v_{a_1}, v_{a_2}}, \dots, e_{v_{a_{m-1}}, v_{a_m}}\} \subseteq E$, for any path $\pi_{v_{a_0}, v_{a_m}} = (v_{a_0}, v_{a_1}, \dots, v_{a_m})$, $v_{a_i} \in V$ that sequentially passes through the nodes $\{v_{a_0}, v_{a_1}, \dots, v_{a_m}\}$. The length of a path on the graph $G(V, E)$ can be defined as the d_1 distance which is formulated in Eq. (4) and computed by aggregating edge lengths on the given path. Different from the d_1 distance, the d_∞ distance defined by Eq. (5) measures the maximal edge length of the path $\pi_{v_{a_0}, v_{a_m}}$:

$$d_1(\pi_{v_{a_0}, v_{a_m}}) = \sum_i d(e_{v_{a_i}, v_{a_{i+1}}}) \quad (4)$$

$$d_\infty(\pi_{v_{a_0}, v_{a_m}}) = \max_{i \in \{0, \dots, m-1\}} \{d(e_{v_{a_i}, v_{a_{i+1}}})\} \quad (5)$$

Employing the d_1 and d_∞ metrics, we can simply define the minimax path $\pi_{p,q}^m$ and the geodesic path $\pi_{p,q}^g$ that link two terminal pixels p and q in Eqs. (6) and (7) respectively:

$$\pi_{p,q}^m \in \arg \min_{\pi_{p,q} \in \Pi_{p,q}} (d_\infty(\pi_{p,q})) \quad (6)$$

$$\pi_{p,q}^g \in \arg \min_{\pi_{p,q} \in \Pi_{p,q}} (d_1(\pi_{p,q})) \quad (7)$$

From the definitions, we can easily observe that the geodesic path $\pi_{p,q}^g$ is the path that links p, q with the smallest length. Similarly, the minimax path $\pi_{p,q}^m$ is just the path connecting p to q with the smallest maximum edge. Further, we define the minimax length $d_m(\pi_{p,q}^m)$ of the minimax path $\pi_{p,q}^m$ and the geodesic length $d_g(\pi_{p,q}^g)$ of the geodesic path $\pi_{p,q}^g$ in Eqs. (8) and (9) respectively. In plain English, the minimax length and the geodesic length are just the total length of the minimax path $\pi_{p,q}^m$ and the geodesic path $\pi_{p,q}^g$ which are illustrated in Eqs. (6) and (7) respectively:

$$d_m(\pi_{p,q}^m) = d_1(\pi_{p,q}^m) \quad (8)$$

$$d_g(\pi_{p,q}^g) = d_1(\pi_{p,q}^g) \quad (9)$$

The minimax distance is more suitable than the geodesic distance, for the minimax path is more sensitive to color transition than geodesic. If a path $\pi_{v_{a_0}, v_{a_m}}$ follows the geometric structure of the guidance image I , its largest edge length $d_\infty(\pi_{v_{a_0}, v_{a_m}})$ must be very small; otherwise, the length of crossing boundary edge will increase $d_\infty(\pi_{v_{a_0}, v_{a_m}})$. The minimax path $\pi_{p,q}^m$ is just the shortest d_∞ path. In contrast, the geodesic path prefers the shortest path which tolerates large color transition on the path, as shown in Fig. 1. We can also verify the behavior from the definition (7) of the geodesic path. The definition of the geodesic path prefers the path with the shortest d_1 distance. If there is a path that achieves the shortest d_1 distance by cutting across sharp edges as illustrated in Fig. 1, the definition will choose it as the geodesic path. Obviously, the geodesic path in the situation is not geometric aware. On the contrary, the minimax path minimizes the maximal edge length of a path and therefore can guarantee that the path does not cut across any sharp edge and follow the regularity of guidance image.

Another advantage of minimax path is that we can efficiently extract all pairs minimax paths. Hu [6] proved that a path of the MST T_G which connects distinct pairs of nodes p, q is the minimax path of the G . Thus, all pairs minimax paths are recorded by the MST. In the literature, the MST can be retrieved in linear time [10] for a single processor or in $O(\log n)$ time [11] for multiprocessors. Indeed, MST as a powerful tool which can capture the geometric structure of nature image has already been successfully used in



Fig. 1. The comparison between the geodesic path and the minimax path. The figure shows that the geodesic chooses the crossing edges path which has a large jump at the color boundaries as almost the path is on the flatten area and the length on flatten area is nearly zeros whereas the minimax path is more sensitive to color edges and is able to follow the structure of guidance image.

image filtering [27], segmentation [28] and cost aggregation [2]. In our application, MST guarantees that we can adaptively choose all the reliable seeds and will not interpolate depths across the boundary.

At last, we notice that although there are lots of geometric aware distances such as the average-commute-time distance [24] and the SoP covariance distance [25] in the literature. However only the geodesic path and the minimax path are discussed here. The reason is that all these geometric aware distances are extremely time consuming and thus they are not suited for depth interpolation in real time. In the depth upsampling literature, none of them are employed to upscale a low-resolution depth map. So we only compare the geodesic distance with the minimax path.

3.3. A reliable seed choosing strategy

An MST of the graph representation $G(V, E)$ of the depth map connects all points on the graph $G(V, E)$. The path linking two nodes along the MST is just the minimax path between them [6]. We can easily find that the seeds on the graph $G(V, E)$ separate the graph into L different regions Ω_i . So these seeds form the boundaries B_i of these regions Ω_i . For instance, Fig. 2a illustrates a graph representation of a depth map, where the green nodes represent the target pixels and the yellow pixels denote the seeds. Fig. 2b shows that an MST of the graph representation of the depth map connects all nodes together. Moreover, we can observe that the yellow seeds divide the green nodes into different parts and therefore form the boundaries of these partitions. In other words, for any two nodes in different regions, the path linking them along the MST must meet a yellow seed at least (i.e. the minimax path from a non-boundary seed to any target pixel in Ω_i must pass through a boundary seed of Ω_i). Since boundary points are the nearest seeds to the target pixels in the sense of the minimax distance, for each target pixel in the region Ω_i , using the depths of the boundary points B_i to interpolate the depths of the target pixels in the region Ω_i is more reliable than using other seeds to interpolate the missing depths. We call these boundary seeds B_i as the reliable seeds of the target pixels in Ω_i and exclusively use their depths to interpolate the target pixels in Ω_i .

The reliable seed choosing strategy has three major advantages. First, it automatically catches all reliable seeds for any target pixel and thus our interpolation algorithm can employ all reliable depth information to interpolate the depth of each target pixel. Second, it is not sensitive to the seed distribution and can be used for various seed distributions other than the uniform distribution taken by upsampling. Last but not least, the strategy can be integrated into a linear time joint interpolation algorithm without extra cost.

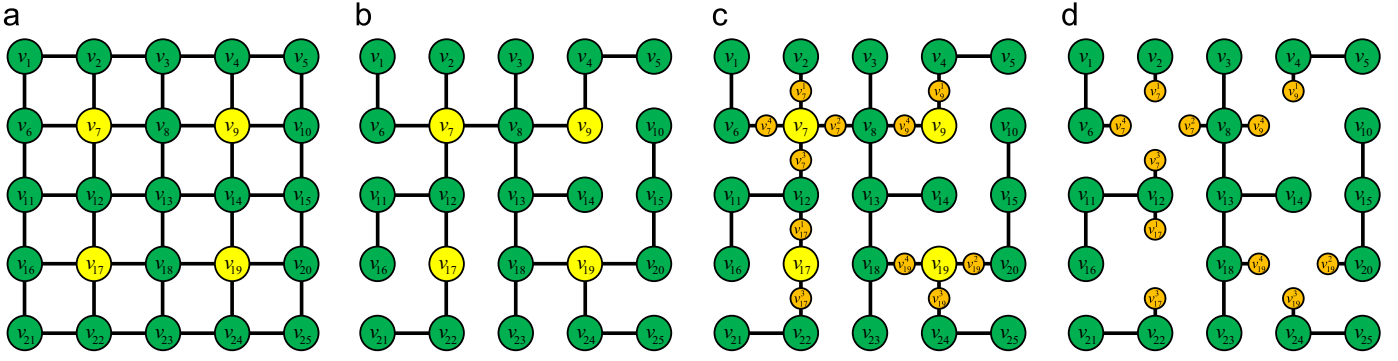


Fig. 2. The minimax path illustration. (a) The graph representation G of guidance image, where the green nodes are target pixels and yellow nodes denote seeds. (b) An MST of the graph G . The yellow seeds partition the green nodes into different parts. (c) The obtained interpolation tree by inserting orange auxiliary nodes into the MST. (d) The auxiliary nodes divide the interpolation tree into several parts by wiping off seeds.

3.4. A linear complexity joint interpolation algorithm

Our interpolation formula is expressed as Eq. (10), which not just take the minimax distance as the similarity metric, but also employs our reliable seed choosing strategy to choose the interpolating seeds for each target pixel:

$$D(p) = \frac{\sum_{q \in \tilde{B}_i} w_{p,q}^m D(q)}{\sum_{q \in \tilde{B}_i} w_{p,q}^m} \quad p \in \Omega_i \setminus S \quad (10)$$

where $w_{p,q}^m = \exp(-\frac{d(\pi_{p,q}^m)}{\sigma_m})$ and $1 \leq i \leq L$.

The formula can be computed in a linear time by transforming the MST into an interpolation tree and updating depths along the interpolation tree T . Let $N(s)$ denote the neighborhood nodes which are linked to the seed s by the MST. For each edge $e_{s,p}$, $p \in N(s)$, we insert an intermediate point s_p into the middle of s and p , as shown in Fig. 2c, and assign $d(e_{s,p}) = d(e_{s,s_p})$, $d(e_{s_p,p}) = \infty$. Here, all auxiliary points are denoted as A . We can observe from Fig. 2d that the interpolation tree T is divided into L subtrees T_i by wiping off all seeds, and the depths of auxiliary nodes $\tilde{B}_i = A \cap T_i$ determine the depths of the target pixels in the subtrees T_i . Then Eq. (10) can be rewritten as

$$D(p) = \frac{\sum_{q \in \tilde{B}_i} w_{p,q}^t D(q)}{\sum_{q \in \tilde{B}_i} w_{p,q}^t} \quad p \in T_i \setminus A, \quad 1 \leq i \leq L \quad (11)$$

where $w_{p,q} = \exp(-\frac{d(\pi_{p,q}^t)}{\sigma_m})$ and $\pi_{p,q}^t$ is the path that connects p to q along the interpolation tree T .

In order to obtain the updating formulae with linear complexity, we put $\tilde{C}(q) = D(q)$, $\tilde{W}(q) = 1$, for $q \in S$; $\tilde{C}(q) = D(s)$, $\tilde{W}(q) = 1$, for $q \in N(s)$, $s \in S$; $\tilde{C}(q) = 0$, $\tilde{W}(q) = 0$, for a target pixel q , then Eq. (11) can be further transformed into following forms:

$$C(p) = \sum_{q \in T} w_{p,q}^t \tilde{C}(q) \quad W(p) = \sum_{q \in T} w_{p,q}^t \tilde{W}(q) \quad (12)$$

$$D(p) = C(p)/W(p) \quad p \in T_i \setminus A, \quad 1 \leq i \leq L \quad (13)$$

In Eq. (12), we replace \tilde{B}_i with T . Although the value of $C(p)$ and $W(p)$ for $p \in T_i \setminus A$ receives supports from all other nodes on the interpolation tree T according to Eq. (12), the depths of target pixels in T_i only depend on the depths of auxiliary nodes \tilde{B}_i as the length of the edge between the seed and its auxiliary points is set as infinity.

Algorithm 1. Joint interpolation algorithm.

Input: Graph $G = (V, E)$; Intermediate costs $C^\dagger(v)$, $W^\dagger(v)$; Costs $C(v)$, $W(v)$;
Output: Depth map $D(v)$;
 1: **for** each $v \in V$ **do**
 2: **if** v is a seed **then**

```

3:          $C^\dagger(v) = C(v) = D(v)$ ;  $W^\dagger(v) = W(v) = 1$ ;
4:     else
5:          $C^\dagger(v) = C(v) = 0$ ;  $W^\dagger(v) = W(v) = 0$ ;
6:     end if
7: end for
8: Retrieve an MST  $T_G = (V, E')$  with  $E' \subset E$  from  $G$ ;
9: Assign each node  $v \in V$  a breadth-first traversal order according to  $T_G$ ;
10: for  $i \in [\#T, \dots, 1]$  do
11:     if  $P(v_i)$  is not a seed or  $C^\dagger(v_i)$ ,  $W^\dagger(v_i)$  are 0 then
12:         Update  $C^\dagger(P(v_i))$ ,  $W^\dagger(P(v_i))$  using Eqs. (14) (15);
13:     end if
14: end for
15: for  $i \in [1, \dots, \#T]$  do
16:     if  $v_i$  is not a seed then
17:         Update  $C(v_i)$ ,  $W(v_i)$  using Eqs. (16) (17);
18:     end if
19: end for
20:  $D(v) = C(v)/W(v)$  Eq. (13);

```

Yang [2] shows that the non-local cost aggregation form $C(p)$ and $W(p)$ which is defined on the tree can be efficiently figured out by traversing the tree structure of the interpolation tree T in two sequential passes. Initially, we visit the nodes of T_p using the breadth first traversal algorithm to assign each node v_i an order i based on the visit sequence with the property that if v_i is the father of v_j , then $i \leq j$, where $1 \leq i, j \leq \#T_p$, $\#T_p$ denotes the nodes number of T_p . In the first pass, while the tree is traced from $v_{\#T_p}$ to v_1 , $C^\dagger(v_i)$ and $W^\dagger(v_i)$ are not updated until all its children have been updated by the updating formulae:

$$C^\dagger(P(v_i)) = C^\dagger(P(v_i)) + w_{v_i, P(v_i)}^t C^\dagger(v_i) \quad (14)$$

$$W^\dagger(P(v_i)) = W^\dagger(P(v_i)) + w_{v_i, P(v_i)}^t W^\dagger(v_i) \quad (15)$$

where $C^\dagger(v)$ and $W^\dagger(v)$ are initialized as $\tilde{C}(v)$ and $\tilde{W}(v)$ respectively, $P(v)$ records the father of node v , both $C^\dagger(v)$ and $W^\dagger(v)$ are the intermediate aggravated results. In the second pass, we employ the updating formulae (16) and (17) to renew $C(v)$ and $W(v)$ when the tree is traversed from v_1 to $v_{\#T_p}$. At last, we use Eq. (13) to estimate final interpolation depths:

$$C(v_i) = w_{P(v_i), v_i}^t \tilde{C}(P(v_i)) + (1 - w_{v_i, P(v_i)}^t) C^\dagger(v_i) \quad (16)$$

$$W(v_i) = w_{P(v_i), v_i}^t \tilde{W}(P(v_i)) + (1 - w_{v_i, P(v_i)}^t) W^\dagger(v_i) \quad (17)$$

At last, removing all auxiliary points s_p from final results, we can obtain the interpolation results. Note that since $d(e_{s_p, s})$ is

infinity, no points can contribute its depth to the seed s through the edge $e_{s_p,s}$. We thus are able to keep the depths of seeds unchanged. Conversely, the depths of seeds would also not propagate its depth to target pixels. This is an undesirable situation. To solve the problem, we assign the depths of seeds to the auxiliary points s_p and employ s_p as proxies to propagate the depths of seeds to target pixels in the same regions. This is the reason why we introduce auxiliary points at first and get rid of them at last.

3.5. Refinement of the joint interpolation algorithm

The linear complexity joint interpolation algorithm stated above is not the most efficient interpolation algorithm yet as it involves the operation of interpolating nodes into a predefined MST and removing them from final results. Supposing that the redundant operation could be eliminated, the algorithm will become more compact. Through an in-depth analysis, we can discover that the auxiliary points s_p are only used to prohibit the depths diffusion between different parts B_i which are segmented by the seeds S along the MST. The function of auxiliary points can be replaced by introducing if statement to judge whether the algorithm propagates depth information across boundaries. Specifically, in the bottom to top pass, we check whether the point $P(v_i)$ is a seed. If it is not a seed, we use (Eqs. (14) and 15) to update the values of $C^1(P(v_i))$ and $W^1(P(v_i))$ respectively. Otherwise, we keep them unchanged. Similarly, in the up to down pass, we check whether the point v_i is a seed. If it is not a seed, we use (Eqs. (16) and 17) to update the values of $C(v_i)$ and $W(v_i)$. Otherwise, we also keep them unchanged. In this paper, we adopt the updating strategy to remove the unnecessary auxiliary points and report the refined interpolation algorithm in Algorithm 1.

At last, we discuss an implementation detail for people who are interested in it. As is known to all, the MST is not unique for a graph. In this situation, we can choose more than one MST to perform interpolation. People maybe wonder what would happen if there are more than only one MST and how to choose an appropriate MST. Frankly speaking, Algorithm 1 does not tell us how to determine an MST because it is not important. More clearly, our algorithm is rather robust for different MSTs. To prove this, we randomly select five different MSTs and choose the interpolation result of an MST as the reference image. After that, we compare the comparison interpolation results (i.e. interpolation results of the other MSTs) with the reference and quantify the similarity. Here, we prefer the structural similarity (SSIM) index [29] to measure the affinity between two images. For SSIM index, one is only reachable in the case of two identical sets of data, and zero means two images are completely different. The SSIM data in Table 1 demonstrates that the similarity between the comparison interpolation results and the reference image. It is easy to verify that the interpolation results for different MSTs are nearly same.

3.6. An analysis of the computational complexity

The computational complexity of our method consisted of two parts: the cost of extracting an MST from the graph representation G of the guidance image I and the cost of updating Eqs. (14)–(17). In Eqs. (14)–(17), $w_{P(v_i),v_i}^t$ only depends on the edge weights of the interpolation tree T and can be pre-computed in linear complexity. Thus, only a total of 6 addition/subtraction operations and 7 multiplication/division operations are required for each node. Hence we can complete the interpolation task with $O(n)$ complexity disregarding the number K of reliable seeds.

For extracting an MST from a graph, Karger et al. [10] presented a linear time randomized algorithm to extract an MST from a graph. With a linear number of processors, the parallel algorithm

Table 1

The similarity measurements for the interpolation results produced by different MSTs. The SSIM indices are calculated from an interpolation reference image and four comparison interpolation images, where both the reference image and four comparison images are produced by different MSTs.

Index	MST1	MST2	MST3	MST4
SSIM	0.987	0.991	0.989	0.994

[11] for the minimum spanning tree problem can solve the problem in $O(\log n)$ time. Overall, we conclude that the overall complexity of our algorithm is $O(n)$.

4. Experiments

We implement our algorithm using Matlab and compare it with state-of-the-art methods on a desktop computer with 4 GB memory. At first, we will disclose why our method can achieve the best results in terms of accuracy and efficiency by performing an in-deep investigation on the difference between our algorithm and other depth upsampling methods. After that, the parameter sensitivity and computational analysis of our algorithm are offered. Finally, we exhibit the interpolation results of real world data to test the interpolation ability of our method.

4.1. An in-deep investigation

In this section, we first employ a toy example to illustrate the implementation differences between different methods and explain why these differences contribute to a spectacular interpolation performance improvement.

Fig. 3 is exploited to disclose the implementation differences between the joint upsampling methods [1,21] and ours algorithm, where A , B and C are the foreground points, in addition, B also stands for a target pixel. We can observe that the user-specified window Ω_B [21] in Fig. 3a not just selects the foreground seeds D and E for target pixel B , but also accepts the background seeds F, G as interpolating seeds. Thus, it is rational to conclude that the user-specified window Ω_B is not able to adapt itself to the underlying geometric structure of the guidance image. In the paper [1], Liu et al. assumed that the geodesic path always follows the underlying regularity and selects the K nearest seeds \mathcal{K}_B as reliable seeds. However, Fig. 3a shows an exception: the geodesic paths $\pi_{A,B}^g$ and $\pi_{B,C}^g$ cut across the background area and select the background seeds D, E as the reliable seeds of target pixel B . In contrast, the minimax paths $\pi_{A,B}^m$ and $\pi_{B,C}^m$ in Fig. 3a do not contain any edges that cut across boundaries. Therefore, our reliable seed choosing strategy only selects the foreground reliable seeds F and G .

Fig. 3b shows the upsampled depth profiles of three methods. We select a path that goes through B and cut across the foreground and background in Fig. 3a as the profile. We can observe that our approach can keep the sharp edge without artifacts. Affected by the background seeds D and E , the edges of both joint bilateral upsampling and joint geodesic upsampling are smoothed unavoidably.

Except for the reliable seeds choosing strategy, both the similarity metric and the computational complexity of our algorithm outperform previous methods [1,21]. The Euclidean distance d_e defined on a 5D manifold [32] is not geometric-aware. Liu et al. [1] substituted it with the geodesic distance d_g . Fig. 3 shows that the ability of detecting the underlying structure of the minimax path is better than the geodesic path, thus selecting the minimax path length d_m as an affinity metric should be more reasonable

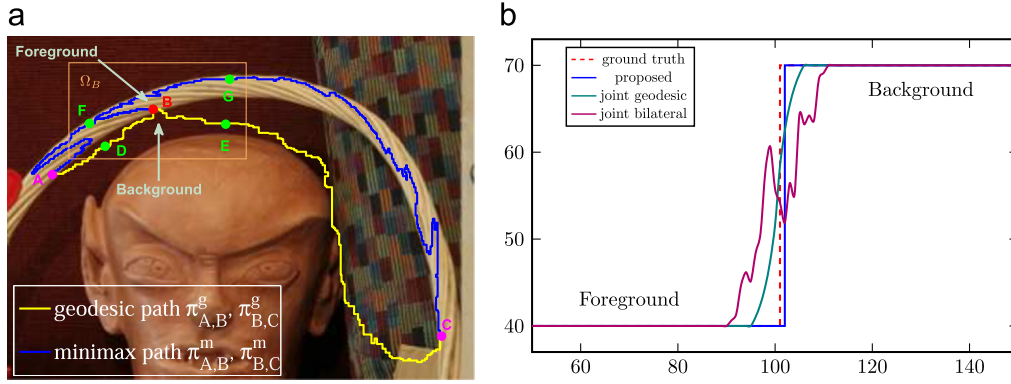


Fig. 3. Comparison among joint bilateral, joint geodesic and our upsampling methods. (a) Geodesic paths and minimax paths between A, B and B, C, where B is the target point, D, E, F and G are the seeds, D and E, F and G are on the geodesic paths and minimax paths respectively. (b) The upsampled depth profiles of joint bilateral, joint geodesic and our upsampling methods.

than employing the geodesic distance d_g . Moreover, compared with the $O(Kn)$ complexity of [1,21], the complexity of our adjoint depth upsampling method is only $O(n)$.

Although the form of Algorithm 1 is similar to the non-local cost aggregation [2] of Yang, the discrepancies between them are notable. First, Yang takes an unnormalized weighted average scheme as illustrated in Eq. (3). But, directly applying Yang's scheme to interpolate the missing values is inadvisable. Instead of using the unnormalized scheme, we exploit a normalized scheme to interpolate missing depths. Second, Yang claims that every node receives supports from all other nodes. Thus, the depth of target pixel will be contaminated by the unreliable depths of other target pixels and the depths of unreliable seeds. Even worse, these unreliable depths will also modify the correct depths of seeds according to the interpolation formula (3) of Yang. In contrast, we exclusively use the depths of reliable seeds to estimate the missing depths. In this way, we can keep the values of target pixels as accurate as possible. Last but not least, the fast algorithm of Yang performs on the entire image domain. On the contrary, our interpolation is based on segmentation area in which the depths of pixels are nearly the same and the pixels are surrounded by the boundary seeds. As these boundary seeds are the most correlative nodes for the interior points, we can accurately compute the missing depths of each segmentation according to the boundary seeds. In the sequel, we will show that these modifications contribute to a spectacular performance improvement.

4.2. Quantitative and visual evaluation

In this section, only upsampling results are compared with seven methods [1,2,14,21,22,17,16] on Middlebury's benchmark [33] to verify the interpolation ability of our approach because these methods only focus on the depth upsampling and the interpolation quality of our method is independent of the seed distribution (refer to Section 4.3). For fair comparison, we subscribe the $2 \times$, $4 \times$, $8 \times$, and $16 \times$ upsampling results of Art, Book, and Moebius from original authors. In the experiments, the low resolution depth images are obtained by downsampling and we use the percentage of bad matching pixels (PBP) [30] and the peak signal-to-noise ratio (PSNR) [31] to evaluate the performance of discontinuous regions and continuous areas, where the discontinuous regions are obtained by dilating 1-pixel wide edge of ground truth and the rest of image domain Ω are considered as continuous areas. In the literature, PBP estimates the interpolation quality via calculating the percentage of bad matching pixels and PSNR is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. The seven methods used for comparison are the joint bilateral upsampling (JBL) [21], the

joint geodesic upsampling (JG) [1], the bilateral filter upsampling with cost volume (BF) [22], the non-local aggregation (NLA) [2], the auto-regressive optimization (AR) [14], the structure and texture aware depth map upsampling (ST) [17] and the depth local features upsampling (LF) [16]. In the following experiments, we assign $\sigma_m = 0.05$ consistently for our method because the parameter setting can produce satisfactory results for most situations according to our experience.

Fig. 5 exhibits the $8 \times$ upsampling results of the seven methods for visual comparison. Our algorithm produces the sharpest edge results (Fig. 5g) with fewest artifacts. JBL employs the Euclidean distance d_e [21] to measure the affinity without regarding the intermediate values between the target pixel and the seed. Thus, the Euclidean distance is not aware of the edge between the coupled nodes and tends to blur boundaries (Fig. 5c). In contrast, JG offers better results (Fig. 5d) as it not just exploits the geodesic path to detect the geometric structure between the interpolated pixels and the interpolating pixels, but also utilizes the geodesic distance d_g to measure the similarity of the coupled nodes. However, Fig. 3a shows that the geodesic path is not as edge sensitive as the minimax path. The shortcoming causes that the upsampled edges are not as sharp as ours. Both NLA and our methods utilize MST to guide the interpolation. In the results (Fig. 5b) of NLA, each point p indiscriminately receives the contributions from other nodes in the image domain Ω and the correct depths of seeds are modified (refer to Eq. (3)). On the contrary, our method exclusively employs the boundary seed B_i to interpolate the depths of the interior point $p \in \Omega_i$ and gets rid of the negative effects of unreliable depths. The upsampled edges of NLA thus are also not as sharp as ours. Unlike our method, AR tends to smooth depth transition (Fig. 5a) as its optimization framework cannot completely prohibit the edge depth information diffusion between boundaries. Same as AR, ST and LF are all optimization based upsampling methods. The major difference is how to estimate the weights of neighborhood pixels. Therefore, they also suffer from the same problem of AR. We illustrate the results of ST and LF in Fig. 5e and f respectively. It is not hard to observe that the edges of ST and LF are also not as sharp as ours and there are nasty artifacts along depth edges.

The quantitative evaluation of PBP and PSNR indices at discontinuous regions and continuous areas are reported in Tables 2 and 3 respectively. We can find that the PBP and PSNR indices at continuous areas of different methods are almost same. Only the PBP and PSNR indices at discontinuous regions, which reflect the edge preserving ability, can distinguish different methods. Benefitting from the geometric-aware minimax path, our method achieves the lowest DISC ranks and highest PSNR scores among all methods as we anticipated at the beginning.

Table 2

PBP quantitative comparison. We compare the proposed algorithm with state-of-the-art methods on the Middlebury dataset using percentage of bad matching pixels (PBP) [30], where the disparity error tolerance is 1. DISC denotes the PBP index at discontinuous regions and CONT stands for the PBP index at continuous areas.

Methods	Art				Book				Moebius			
	2 ×	4 ×	8 ×	16 ×	2 ×	4 ×	8 ×	16 ×	2 ×	4 ×	8 ×	16 ×
AR [14]												
DISC	0.15	0.24	0.32	0.41	0.12	0.23	0.34	0.39	0.12	0.20	0.29	0.33
CONT	0.03	0.06	0.10	0.17	0.03	0.05	0.10	0.19	0.02	0.04	0.07	0.14
NLA [2]												
DISC	0.17	0.27	0.39	0.48	0.22	0.33	0.41	0.49	0.16	0.24	0.33	0.48
CONT	0.01	0.04	0.07	0.12	0.02	0.04	0.08	0.18	0.01	0.03	0.05	0.10
JBL [21]												
DISC	0.16	0.24	0.39	0.49	0.19	0.27	0.41	0.49	0.16	0.23	0.35	0.48
CONT	0.01	0.03	0.06	0.12	0.01	0.02	0.05	0.16	0.01	0.02	0.04	0.10
JG [1]												
DISC	0.11	0.19	0.30	0.36	0.06	0.18	0.35	0.43	0.10	0.20	0.24	0.34
CONT	0.03	0.05	0.08	0.12	0.03	0.05	0.09	0.17	0.02	0.04	0.06	0.12
LF [16]												
DISC	0.09	0.17	0.26	0.31	0.05	0.14	0.20	0.33	0.08	0.17	0.20	0.32
CONT	0.02	0.04	0.08	0.12	0.02	0.05	0.09	0.15	0.01	0.02	0.06	0.10
ST [17]												
DISC	0.07	0.12	0.21	0.27	0.03	0.10	0.12	0.21	0.06	0.12	0.16	0.27
CONT	0.01	0.04	0.07	0.12	0.01	0.04	0.08	0.14	0.01	0.02	0.05	0.11
Ours												
DISC	0.04	0.09	0.16	0.22	0.02	0.06	0.09	0.15	0.04	0.09	0.14	0.21
CONT	0.01	0.03	0.07	0.12	0.01	0.04	0.08	0.13	0.01	0.02	0.05	0.10

Table 3

PSNR quantitative comparison. We compare the proposed algorithm with state-of-the-art methods on the Middlebury dataset using the peak signal-to-noise ratio (PSNR) [31]. DISC denotes the PSNR index at discontinuous regions and CONT stands for the PSNR index at continuous areas.

Methods	Art				Book				Moebius			
	2 ×	4 ×	8 ×	16 ×	2 ×	4 ×	8 ×	16 ×	2 ×	4 ×	8 ×	16 ×
AR [14]												
DISC	25.94	24.08	23.42	19.54	33.93	32.19	29.11	25.43	35.15	32.78	30.06	26.74
CONT	47.88	46.28	44.98	41.38	56.27	53.96	51.04	48.31	57.03	54.27	52.76	50.12
NLA [2]												
DISC	24.72	23.22	21.41	19.47	33.14	30.94	26.61	23.04	32.02	29.96	27.48	24.54
CONT	46.94	46.01	44.69	40.38	56.03	52.97	51.04	47.31	56.81	53.92	52.98	49.47
JBL [21]												
DISC	24.30	23.03	21.06	19.12	32.74	30.11	26.05	23.16	31.92	29.02	27.48	24.05
CONT	47.54	46.01	43.30	41.42	55.56	53.11	50.89	47.21	56.18	53.97	51.64	49.71
JG [1]												
DISC	25.72	23.89	22.76	18.83	33.13	31.47	28.31	25.03	34.98	32.64	29.69	27.63
CONT	46.92	45.89	44.35	40.83	55.18	53.23	51.76	49.21	57.22	54.16	52.64	49.64
LF [16]												
DISC	24.29	22.79	22.21	18.95	32.73	30.17	27.95	26.46	33.50	31.88	29.00	27.02
CONT	46.77	45.92	44.24	40.99	55.92	53.34	51.65	48.47	56.96	54.47	51.49	50.40
ST [17]												
DISC	26.07	24.21	23.22	19.21	34.32	32.06	29.74	26.61	35.11	33.24	30.29	28.73
CONT	47.02	46.19	45.05	41.42	56.02	53.34	51.60	48.91	57.63	54.74	52.45	50.17
Ours												
DISC	26.87	25.22	24.01	20.74	35.02	32.96	30.48	27.54	36.14	33.94	31.61	29.22
CONT	47.12	46.29	45.01	41.54	56.60	53.42	51.60	49.01	57.81	54.92	52.03	50.47

4.3. Parameter sensitivity and computation analysis

The parameter setting of our method is simpler than previous depth upsampling methods [1,12–14,20–22,17,16] because our algorithm has only one parameter σ_m which adjusts the similarity

of coupled nodes. Using different σ_m , we record the PBP scores of our method to draw the performance curves of PBP and report the statistical data in Fig. 4. Since the performance of our algorithm at the discontinuous regions and continuous areas is rather robust in a wide range of σ_m as illustrated in Fig. 4, we do not need to fine

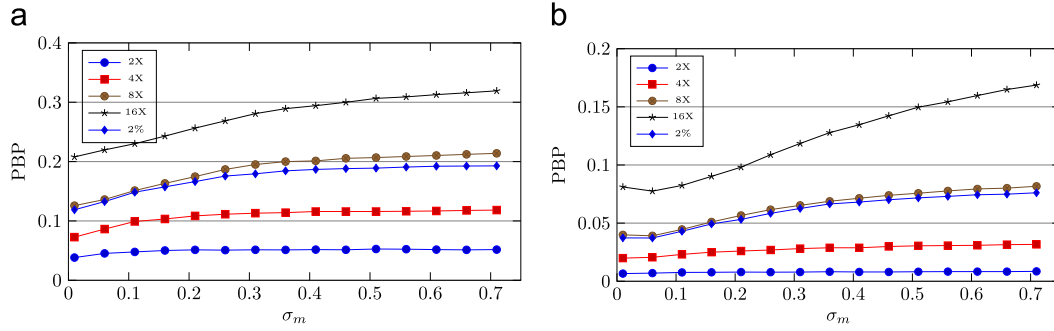


Fig. 4. Parameter sensitivity. PBP performance curves of discontinuous regions and continuous areas are shown as a function of σ_m . The proposed algorithm renders good performance for a wide range of σ_m under $2\times$, $4\times$, $8\times$, $16\times$ upsampling rates and 2% random missing inpainting: (a) Discontinuous regions and (b) continuous areas.

Table 4

Computational speed. The table shows the processing speed (frames per second) achieved for upsampling and inpainting.

Speed	$2\times$	$4\times$	$8\times$	$16\times$	5% random missing
fps	8	8	8	8	8

tweak the scale factor σ_m for pursuing stable results. In Fig. 4, the PBP curves of $8\times$ upsampling and 2% random missing inpainting are very close. We can also discover that the seed numbers of $8\times$ upsampling and 2% random missing inpainting are nearly the same by a simple calculation. The phenomenon is not casual. It implies that the interpolation quality of our method primarily depends on the seed number. Fig. 4 also illustrates the PBP performance is decreasing while σ_m increases. The major reason is that the color difference is downplayed when σ_m becomes larger, which leads to blurred boundaries.

We use a 690×560 image to test the computational efficiency of our method. Table 4 lists the running times for 2% random missing inpainting and $2\times$, $4\times$, $8\times$ upsampling. The running time of our method is stable at 8 fps and has nothing to do with applications. Compared with the fastest upsampling method [1] which achieves 0.03 fps for exact method and 3 fps for approximation approach, our algorithm outperforms it significantly. Moreover, our algorithm can achieve further acceleration. We find that nearly 90% running time are spent on extracting an MST and Vineet [34] proposed a fast GPU extracting method which obtains 30–50 times speedup over the CPU.

4.4. Real world experiments

In the last experiments, we apply our method to process the range map of Microsoft Kinect. Microsoft Kinect, which is designed for man–computer interaction, equips with a depth camera and a color camera sensor array. The 680×460 depth map acquired from Kinect usually contains lots of holes around depth discontinuities due to various degradation reasons. This kind of structure missing degradation inevitably lowers the quality of depth map and limits its applications.

We input the registered guidance images in Fig. 6a and the deteriorated depth maps in Fig. 6b to our interpolation algorithm. The restoration results are illustrated in Fig. 6c, where the first depth map in Fig. 6b loses a lot of depths for the reason that the black surfaces of laptop and wires absorb the structure-light emitted from Kinect and the second range map in Fig. 6b is synthesized to simulate the occlusion degradation. We also exhibit the results of 5% random missing inpainting and $8\times$ upsampling in Fig. 6e and g respectively. The seed distributions are shown in Fig. 6d and f. We can observe that they suffer from structure

Table 5

PBP performance. The table shows the PBP of 5% random missing inpainting and $8\times$ upsampling.

Index	5% random missing	$8\times$ upsampling
PBP	0.04	0.06
PBP	0.05	0.08

missing degradation too. Undoubtedly, this kind of hybrid degradation will significantly increase the difficulty to restore the missing depths. To evaluate the PBP scores of 5% random missing and $8\times$ upsampling, we consider the depth maps in Fig. 6b as ground truth to evaluate the performance of our method and list the PBP scores in Table 5. From the PBP indices and Fig. 6, we confidently conclude that our method can produce satisfactory results for real world data and can be used in the working environment.

4.5. Object recognition using RGB-D

We are witnessing a new revolution of sensing technologies as the RGB-D camera like the Microsoft Kinect are capable of providing high quality synchronized videos of both color and depth. With the impressive active sensing capabilities, Kinect as a cheaper consumer electronics represents an opportunity to dramatically increase the robustness of object recognition toward real-life recognition applications for common people. However, the depth map acquired by Kinect suffers from various problems. The major degradation of the depth map is that the structure light is apt to be absorbed by the black surface in the scenes and therefore the depth map in the corresponding location forms black holes. The corrupted depth map will unavoidably affect the recognition accuracy.

In the experiments, we use the benchmark provided by Lai et al. [35] for RGB-D object recognition to test the object recognition accuracy. The RGB-D object dataset contains visual and depth images of 300 physically distinct objects taken from multiple views, and the chosen objects are commonly found in home and office environments. To acquire these images, Kinect simultaneously records both color and depth images at 640×480 resolution.

The task of object recognition is difficult partly because images are in high-dimensional space and can change with viewpoint, while the objects themselves may be deformable, leading to large intra-class variation. The core of building object recognition systems is to extract meaningful representations (features) from high-dimensional observations such as images, videos, and 3D point clouds. To compare the recognition performance caused by the degradation of the depth map, we incorporate the recently proposed hierarchical kernel descriptors

Table 6

The accuracy comparisons of object recognition on the different RGB-D images, where the object recognition system is composed by the hierarchical kernel descriptor and SVM. We input the raw RGB-D images, restored RGB-D images, low-resolution RGB-D images and high-resolution RGB-D images to the object recognition system and compare the recognition accuracy.

Methods	Time (s)	Accuracy for restoration (%)				Accuracy for different resolution (%)			
		Raw RGB-D images		Restored RGB-D images		Low RGB-D images		High RGB-D images	
		Category	Instance	Category	Instance	Category	Instance	Category	Instance
AR [14]	40	84.1 \pm 2.2	82.4	85.3 \pm 1.9	83.5	60.1 \pm 2.1	59.2	81.6 \pm 2.2	80.9
NLA [2]	2	84.1 \pm 2.2	82.4	85.0 \pm 2.0	83.1	60.1 \pm 2.1	59.2	81.7 \pm 2.0	80.7
JBL [21]	20	84.1 \pm 2.2	82.4	84.2 \pm 2.0	82.6	60.1 \pm 2.1	59.2	81.0 \pm 2.0	80.9
JG [1]	0.1	84.1 \pm 2.2	82.4	85.2 \pm 2.1	83.3	60.1 \pm 2.1	59.2	81.9 \pm 2.1	81.1
LF [16]	50	84.1 \pm 2.2	82.4	84.2 \pm 2.0	82.5	60.1 \pm 2.1	59.2	81.0 \pm 2.1	80.5
ST [17]	60	84.1 \pm 2.2	82.4	85.5 \pm 1.9	83.7	60.1 \pm 2.1	59.2	81.8 \pm 2.0	81.3
Ours	0.04	84.1 \pm 2.2	82.4	85.1 \pm 2.0	83.3	60.1 \pm 2.1	59.2	81.5 \pm 2.0	81.0

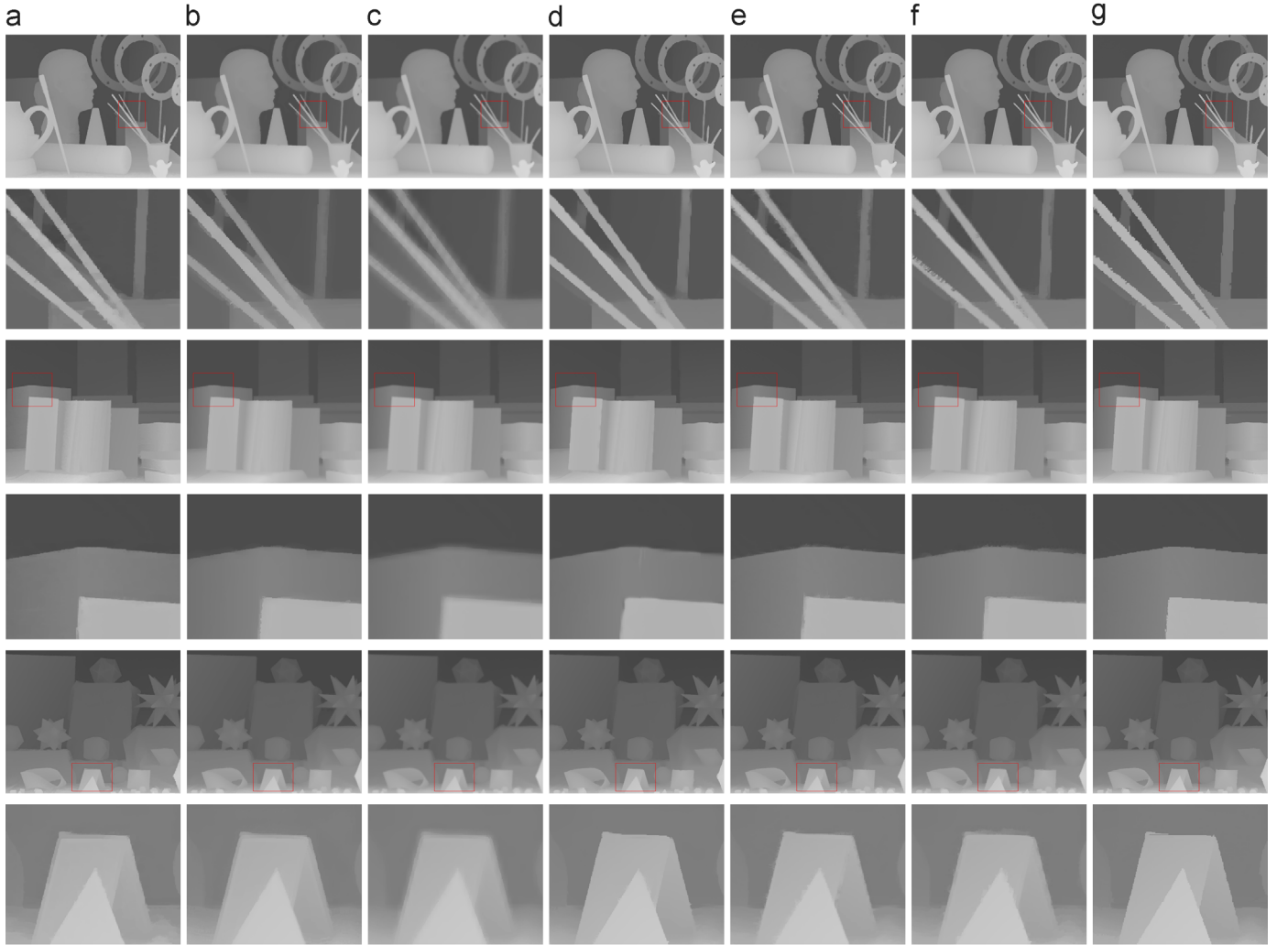


Fig. 5. Visual comparison. Visual comparison of the upsampling results at 8 \times upsampling rate. From left to right, each column corresponds to the results of AR [14], NLA [2], JBL [21], JG [1], ST [17], LF [16] and ours. Our method produces the best upsampling results with sharpest edges and fewest artifacts.

(HKDES) [36], which is able to represent meaningful features of RGB-D data, with SVM to build an object recognition. For more information of the object recognition system, refer to the original paper [36] of Bo and reference therein. The object recognition system is able to distinguish between two levels of object recognition: instance recognition and category recognition. Instance recognition is recognizing distinct objects, for example a coffee mug with a particular appearance and shape. Category recognition is determining the category name of

an object. Note that one category usually contains many different object instances.

Since almost depth map in the RGB-D object dataset [35] are corrupted, we can employ our interpolation method to restore the missing depths under the guidance of the registered color image. After that, we evaluate hierarchical kernel descriptors on the raw RGB-D object dataset [36] and the restored RGB-D object dataset. In the meanwhile, we downsample the resolution the raw RGB-D

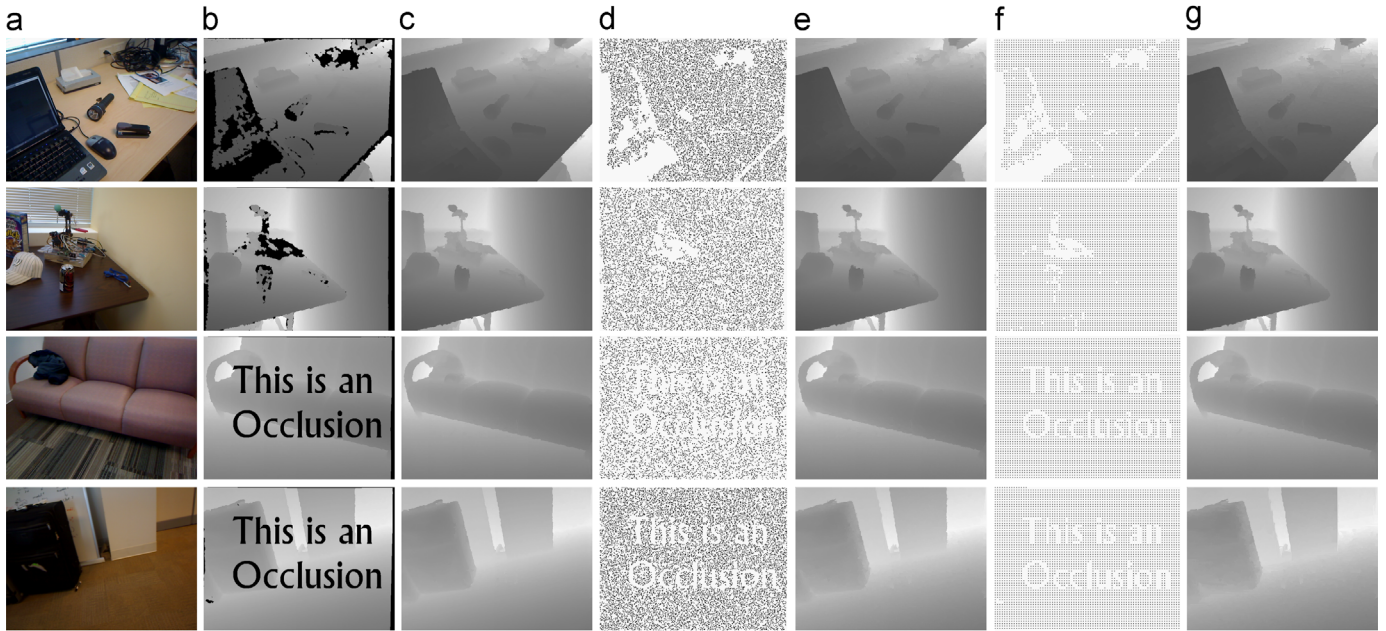


Fig. 6. Visual illustration. Visual illustration of the interpolation results for real world data. (a) The registered guidance images. (b) The depth maps which are used as seed for structure missing. (c) The restoration results for structure missing. (d) The seed distribution for simultaneous 5% random missing and structure missing. (e) The inpainted results of (d). (f) The seed distribution for simultaneous 8 × upsampling and structure missing. (g) The upsampled results of (f).

object dataset with 4× rate to obtain a low resolution version of the object dataset [36] and then upscale them to produce high resolution RGB-D images with the size 600 × 480. The low resolution RGB-D images and high-resolution RGB-D images are also fed to the organization system respectively. It is worth noting that we only perform 4× upsampling here because the object recognition system will fail for too small RGB-D images. The goal of the two experiments is to (1) verify the influence of the corrupted depth map for object recognition; (2) compare the accuracy difference of different resolution RGB-D images. We report the statistic data in Table 6, where the accuracy of category recognition is the average accuracy over 10 random train/test splits. As we expect, both the category recognition accuracy and the instance recognition accuracy of the restored RGB-D images and high-resolution RGB-D images are higher than the accuracy of the raw RGB-D images and low-resolution RGB-D images. This explains the importance of the low level interpolation algorithm for the high level object recognition application.

Finally, we notice that the discrepancies of recognition performance between different interpolation methods are not significant because the object recognition application is a high level application and thus is not very sensitive to the quality of the low level RGB-D images, which causes the recognition performance which is almost same as all interpolation methods. Although the performance difference of recognition is negligible, the processing speed is capable of distinguishing our method from previous methods. Indeed, the major advantage of our method is the speed. Actually, our method is the fastest depth interpolation algorithm according to our experiment. The running time for a 600 × 480 depth map is listed in the first column of Table 6. After a simple calculation, we can figure out that our algorithm is 1000+ times faster than the slowest method.

5. Conclusion

We presented a novel joint interpolation algorithm and developed an efficient implementation. In the algorithm, we employ

minimax paths to denote the geometric structure of the guidance image I and map them onto the target depth map D to guide the depth interpolation. The minimax distance is introduced to measure the affinity of the coupled pixels. A new reliable seed choosing strategy is proposed to select all reliable seeds for target pixels. The seeds on the MST partition image domain Ω into different areas. In each partition, we employ the boundary points (reliable seeds) to estimate the depths of interior pixels (target pixels). We also conducted extensive experiments to compare our method with other depth upsampling methods. The results show that our algorithm outperforms other methods in terms of accuracy and efficiency.

Advantages of the proposed method: Directly applying the non-local cost aggregation (3) of Liu [1] will cause serious problems: The depth of target pixel p will be contaminated by the undefined depths of other target pixels and the depths of unreliable seeds as these depth values tend to deviate from the ground true depth of p drastically. Fortunately, all these problems can be satisfactorily solved by our algorithm. The interpolating process of our method is based on the segmentation that clusters the most correlative points together. Our method also exclusively uses the depths of the most correlative boundary points to interpolate the missing depths of target pixels. We can observe that these modifications contribute to a spectacular performance improvements from Fig. 5b, g and Table 2. Compared with JBL [21], JG [1], BF [22], NLA [2] and AR [14], our method produces the sharpest edge-preserving results with the least running time. We own the achievement to the geometric-aware minimax distance and our novel reliable seed choosing strategy.

Further work: There are two types of artifacts in the interpolation results. One type of artifacts that may arise while interpolating is the edge blurring in the depth map. When two different objects on either side of a depth discontinuity have similar colors, all existing interpolation algorithms tend to produce intermediate depths and blur the depth edges between the two objects. For better results, we wish to distinguish the two similar color objects. In contrast to other geometric structure detection methods, the minimax path can detect tiny color transition and help us produce

better results. However, the solution is not the optimal one yet. The texture copying [23] (i.e. textures from the guidance image are transferred into geometric patterns that appear in the upsampled depth maps) as another kind of degradation is also not well handled by our method. To the best of our knowledge, all joint depth upsampling methods in the literature cannot suppress this artifact as all these methods use color edges to indicate depth edges. Although the texture copying is inevitable in theory, the negative effects of the artifacts are not serious as illustrated in Fig. 6. In the further work, we are going to reduce the two kinds of negative artifacts further.

Acknowledgments

This work is supported by the National High-Tech Research and Development Program of China (863 Program) with No. 2015AA016402, and by National Natural Science Foundation of China with Nos. 61332017, 61331018, 91338202, and 61271430. Authors would like to thank respected anonymous reviewers for their constructive and valuable suggestions for improving the overall quality of this paper.

References

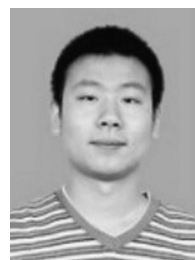
- [1] M.-Y. Liu, O. Tuzel, Y. Taguchi, Joint geodesic upsampling of depth images, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 169–176.
- [2] Q. Yang, A non-local cost aggregation method for stereo matching, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1402–1409.
- [3] R. Sedgewick, K. Wayne, *Algorithms*, 4th edition, Addison-Wesley, New York, 2011.
- [4] M. Pollack, The maximum capacity through a network, *Oper. Res.* 8 (1960) 733–736.
- [5] D.B. Johnson, Efficient algorithms for shortest paths in sparse networks, *J. ACM* 24 (1977) 1–13.
- [6] T.C. Hu, The maximum capacity route problem, *Oper. Res.* 9 (1961) 898–900.
- [7] O. Boruvka, O Jistém Problému Minimálním (About a Certain Minimal Problem) (in Czech, German summary), *Práce Mor. Přírodoved. Spol. v Brně III* 3 (1926).
- [8] R.C. Prim, Shortest connection networks and some generalizations, *Bell Syst. Tech. J.* 36 (1957) 1389–1401.
- [9] J.B. Kruskal, On the shortest spanning subtree of a graph and the traveling salesman problem, in: *Proceedings of the American Mathematical Society*, vol. 7, 1956.
- [10] R.E.T. David R. Karger, Philip N. Klein, A randomized linear-time algorithm to find minimum spanning trees, *J. ACM* 42 (1995) 321–328.
- [11] K.W. Chong, Y. Han, T.W. Lam, Concurrent threads and optimal parallel minimum spanning trees algorithm, *J. ACM* 48 (2001) 297–323.
- [12] J. Diebel, S. Thrun, An application of Markov random fields to range sensing, in: *Proceedings of Advanced Neural Information Processing Systems*, 2005, pp. 291–298.
- [13] J. Park, H. Kim, Y.-W. Tai, M. Brown, I. Kweon, High quality depth map upsampling for 3D-TOF cameras, in: IEEE International Conference on Computer Vision, 2011, pp. 1623–1630.
- [14] J. Yang, X. Ye, K. Li, C. Hou, Depth recovery using an adaptive color-guided auto-regressive model, in: *Proceedings of the 12th European Conference on Computer Vision*, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 158–171.
- [15] J. Li, G. Zeng, R. Gan, H. Zha, L. Wang, A Bayesian approach to uncertainty-based depth map super resolution, in: *Proceedings of the 11th Asian Conference on Computer Vision*, Springer-Verlag, Berlin, Heidelberg, 2013, pp. 205–216.
- [16] Y. Kang, S. Lee, Y. Ho, Depth map upsampling using depth local features, *Electron. Lett.* 50 (2014) 170–171.
- [17] O. Choi, S.-W. Jung, A consensus-driven approach for structure and texture aware depth map upsampling, *IEEE Trans. Image Process.* 23 (2014) 3321–3335.
- [18] L. Dai, H. Wang, X. Zhang, Depth map upsampling using compressive sensing based model, *Neurocomputing* 154 (2015) 325–336.
- [19] Y. Yang, M. Gao, J. Zhang, Z. Zha, Z. Wang, Depth map super-resolution using stereo-vision-assisted model, *Neurocomputing* 149 (2015) 1396–1406.
- [20] J. Dolson, J. Baek, C. Plagemann, S. Thrun, Upsampling range data in dynamic environments, in: IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 1141–1148.
- [21] J. Kopf, M.F. Cohen, D. Lischinski, M. Uyttendaele, Joint bilateral upsampling, in: *ACM SIGGRAPH 2007 Papers, SIGGRAPH '07*, ACM, New York, NY, USA, 2007.
- [22] Q. Yang, R. Yang, J. Davis, D. Nister, Spatial-depth super resolution for range images, in: IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.
- [23] D. Chan, H. Buisman, C. Theobalt, S. Thrun, A noise-aware filter for real-time depth upsampling, in: A. Cavallaro, H. Aghajan (Eds.), *ECCV Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications*, Marseille, France, 2008, pp. 1–12.
- [24] F. Fouss, A. Pirotte, J.-M. Renders, M. Saerens, Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation, *IEEE Trans. Knowl. Data Eng.* 19 (2007) 355–369.
- [25] A. Mantrach, L. Yen, The sum-over-paths covariance kernel: a novel covariance measure between nodes of a directed graph, *IEEE Trans. Pattern Anal. Mach. Intell.* 32 (2010) 1112–1126.
- [26] K.-H. Kim, S. Choi, Walking on minimax paths for k -NN search, in: *AAAI 2013*, 2013.
- [27] J. Stawiaski, F. Meyer, Minimum spanning tree adaptive image filtering, in: *IEEE International Conference on Image Processing*, 2009, pp. 2245–2248.
- [28] T.N. Janakiraman, P.V.S.S.R.C. Mouli, Image segmentation based on minimal spanning tree and cycles, in: *Proceedings of the International Conference on Computational Intelligence and Multimedia Applications*, IEEE Computer Society, Washington, DC, USA, 2007, pp. 215–219.
- [29] Z. Wang, A. Bovik, H. Sheikh, E. Simoncelli, Image quality assessment: from error visibility to structural similarity, *IEEE Trans. Image Process.* 13 (2004) 600–612.
- [30] D. Scharstein, R. Szeliski, A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, *Int. J. Comput. Vis.* 47 (2002) 7–42.
- [31] Q. Huynh-Thu, M. Ghanbari, Scope of validity of PSNR in image/video quality assessment, *Electron. Lett.* 44 (2008) 800–801.
- [32] D. Barash, Fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation, *IEEE Trans. Pattern Anal. Mach. Intell.* 24 (2002) 844–847.
- [33] H. Hirschmuller, D. Scharstein, Evaluation of cost functions for stereo matching, in: *IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.
- [34] V. Vineet, P. Harish, S. Patidar, P.J. Narayanan, Fast minimum spanning tree for large graphs on the GPU, in: *HPG 2009*, ACM, New York, NY, USA, 2009, pp. 167–171.
- [35] K. Lai, L. Bo, X. Ren, D. Fox, A large-scale hierarchical multi-view RGB-D object dataset, in: *IEEE International Conference on Robotics and Automation*, 2011, pp. 1817–1824.
- [36] L. Bo, K. Lai, X. Ren, D. Fox, Object recognition with hierarchical kernel descriptors, in: *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1729–1736.



Longquan Dai received his B.S. degree in Electronic Engineering from Henan University of Technology, China, in 2006. He received his M.S. degree in Electronic Engineering from Shantou University, China, in 2010. Currently, he is working toward the Ph.D. degree in Computer Science at Institute of Automation, Chinese Academy of Sciences, China. His research interests lie in computer graphics, computer vision and optimization-based techniques for image analysis and synthesis.



Feihu Zhang received the B.S. degree in Computer Science from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2014. He is currently a Research Assistant in the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His current research interests include computer vision, image processing and optimization-based techniques for image analysis and synthesis.



Mengke Yuan received his B.S. degree in Mathematics in Zhengzhou university in 2012, and is working towards M.Sc. degree in the same university. He is now a Guest Student in the Sino-French Laboratory (LIAMA) and National Laboratory of Pattern Recognition (NLPR) at Institute of Automation, Chinese Academy of Sciences. His research interests lie in computer aided geometry design, computer vision and optimization based techniques for image analysis and synthesis.



Xiaopeng Zhang received his B.S. degree in Mathematics from Northwest University in 1987, and the Ph.D. degree in Computer Science from Institute of Software, Chinese Academy of Sciences (CAS), in 1999. He is a Professor in the Sino-French Laboratory (LIAMA) and (National Laboratory of Pattern Recognition) at Institute of Automation, CAS. His main research interests are computer graphics and pattern recognition. He received the National Scientific and Technological Progress Prize (Second Class), in 2004. He is a Member of ACM and IEEE.