IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 23, 2020, accepted October 13, 2020, date of publication October 23, 2020, date of current version November 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3033307

Superpixel Segmentation by Forgetting Geodesic Distance

BING LUO[1], JUNKAI XIONG[1], AND LI XU[2]
[1]Center for Radio Administration Technology Development, School of Computer and Software Engineering, Xihua University, Chengdu 610039, China
[2]School of Science, Xihua University, Chengdu 610039, China

Corresponding author: Bing Luo (mathild1987@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61801398 and Grant 61806028, and in part by the Young Scholars Program of Xihua University.

ABSTRACT Superpixel segmentation could be of benefit to computer vision tasks due to its perceptually meaningful results with similar appearance and location. To obtain the accurate superpixel segmentation, existing methods introduce geodesic distance to fit the object boundaries. However, conventional geodesic distance easily suffers from error accumulation and excessive time consumption. This paper proposes a fast superpixel segmentation method based on a new geodesic distance, called forgetting geodesic distance. In contrast to the conventional geodesic distance, the forgetting geodesic distance utilizes a forgetting factor to gradually reduce the influence of previous path cost and focuses on the latest pixels' difference. Intuitively, a pixel with lower difference with respect to the latest path contextual distance will be more similar with the corresponding region. In the new path, the path cost devotes much greater attention to the latest pixels' difference and could significantly relieve error accumulation. The pixels are also aggregated with less dependence on seeds as the path extends, which avoids the seed updating. The experimental results validate that the proposed method obtains 2 percent and 1 percent gain on average compared with most of the state-of-the-art methods in terms of BSD500 and VOC2012 datasets, respectively.

INDEX TERMS Superpixel segmentation, forgetting geodesic distance, error accumulation.

I. INTRODUCTION

Superpixel segmentation aims to obtain local regions with appearance and location consistency. It is used to extract perceptually meaningful element regions, which significantly reduces the computation complexity for other computer vision applications, such as saliency detection [1], [2], object segmentation [3]–[7], object detection [8] and recognition [9], and biomedical image analysis [10].

Existing superpixel segmentation methods can be mainly classified into two categories: graph- and clustering-based methods. Graph-based methods construct a node adjacency graph based on the similarity of adjacent pixels. Then, they treat the superpixel as a graph split task, which is modeled as a variety of energy minimization problems such as those in [6], [11]–[13]. Because Ncut [11] utilizes the eigen-decomposition to approximate the superpixel partition as a nondeterministic polynomial-time (NP)-hard problem, this will be time consuming and result in poor boundary adherence when increasing the image scale and superpixel number. Other graph-based methods mainly utilize combinational optimization to solve this type of energy minimization problem and could efficiently reduce the computational complexity. However, these methods model a superpixel only via adjacent matrices, without considering any seed point locations, and generate superpixels with irregularity [12] or with low boundary adherence [11], [13]. Clustering-based methods evenly assign initial centers in images and update cluster centers until convergence; examples are TurboPixels (TP) [14], simple linear iterative clustering (SLIC) [15], lazy random walks [16], higher-order superpixels [17], distance-metric-based superpixel segmentation [18], linear spectral clustering (LSC) [19] and simple noniterative clustering (SNIC) [20]. These methods employ initial seed points as the anchor points to estimate the position of superpixel centers and label neighboring pixels in local regions. However, these methods consider the Euclidean distance between pixels and seeds in feature space, which only fits the spherical clustering and results in increased undersegmentation error in the case of irregular shapes and blurred object boundaries.

The associate editor coordinating the review of this manuscript and approving it for publication was Madhu S Nair.

In contrast, geodesic distance searches for the shortest path among pixel and seed in an adjacent graph to calculate path cost as a similarity evaluation. It connects the object similarity regions with lowest gradients, which ameliorates the issue of inaccurate similarity measurement in irregular object regions. For example, Wang *et al.* [21] introduced the structure-sensitive superpixels (SSS) algorithm, which considers the structural information of images. It computes the geodesic distance by the geometric flows instead of the simple Euclidean distance. However, geodesic distance is sensitive to color, with gradual change due to significant error accumulation. Meanwhile, computational efficiency caused by measuring the geodesic distance and iterative updating becomes its bottleneck. Hence, gradually forgetting previous errors along with increasing the shortest path is an efficient similarity measurement method, which could devote much greater attention to the latest pixels' difference. We refer to this as forgetting geodesic distance. Then, the new pixel would be added into a cluster if the tail pixels of the path have smaller pixels' difference.

This paper proposes a new geodesic distance based on forgetting strategy to address these issues. Intuitively, a pixel will be more likely to belong to a region when it has less geodesic distance from its local neighbor pixels, which could be determined by a forgetting factor to reduce the influence of the previous pixels on the shortest path. On the one hand, forgetting geodesic distance relieves the error accumulation by summing the errors of the starting pixel and ending pixel in the conventional geodesic distance. On the other hand, the performance of superpixel segmentation will not depend on seed updating. This is because the influences of seed points are faded away, which leads to more rapid superpixel segmentation.

Some subjective results are shown in Fig. 1. We can see that SLIC [15] and SNIC [20] cannot segment the left branch of the first image. The reason for this is that the performances of these two methods heavily depend on seed locations; seeds which cannot locate the fine object structure result in inferior segmentation performance, especially for small superpixel numbers. However, our method relieves the seed influence by forgetting geodesic distance, which enables devoting much more attention to the latest path contextual distance and more easily fits the irregular object structure. Similarly, our method obtains superior segmentation performance in the case of blurred object boundaries in comparison with other methods. This also validates that our method could achieve better boundary extraction through focusing on the latest path contextual distance.

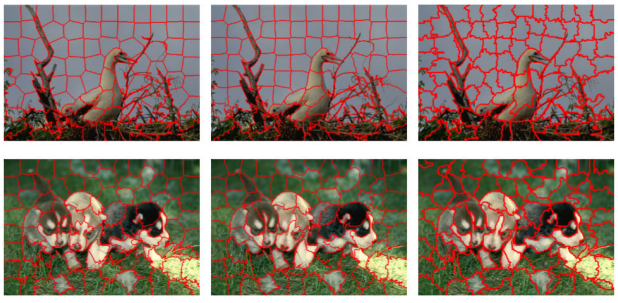

FIGURE 1. Example for compared subjective results. From left to right: SLIC [15], SNIC [20] and our method.

Our main contributions are concluded as follows:

- We propose a new superpixel segmentation method based on forgetting geodesic distance.
- We propose a nonrecursive optimization method to rapidly calculate the new geodesic distance.

II. RELATED WORK

Superpixel segmentation could be roughly divided into two classes: graph-based and clustering-based. The first class constructs image pixels as a graph structure and models the segmentation problem as cut edge searching with minimum cost. For example, Shi and Malik [11] introduce a normalized cut to balance the size of different partitions and relax the cost function as a Rayleigh quotient to obtain the analytic solution. However, this method heavily constrains the size balance for cluster groups to maintain the region regularity, which reduces segmentation performance for the weak boundaries of irregularly shaped objects. Meanwhile, the eigenfactor-based optimization framework results in a low computation efficiency. Felzenszwalb and Huttenlocher [12] proposes a minimum spanning tree (MST)-based superpixel segmentation based on regional differences. This method not only results in irregularity of superpixel regions but also reduces the weak boundary adherence rate due to considering the minimum internal difference and maximum intradifference, especially for images with complex noise backgrounds. Moore *et al.* then propose two regular superpixel segmentation methods based on vertical and horizon seam-like paths [13] and binary region graph cut overlap [22], respectively. To avoid crossing of multiple paths, they add some regularity constraints into their models, which results in not only many narrow superpixel regions but also higher undersegmentation error. Most existing graph-based methods calculate pixel similarity based on adjacent pairwise pixel distance, which poses difficulty in maintaining the regional regularity. Hence, Fu *et al.* [23] proposes a regular superpixel segmentation by finding geodesic path links of two junctions lying on the object boundary. Although this method could more directly obtain superpixel borders to fit with the object boundary, it limits the pairwise link for vertical and horizon junctions to preserve the superpixel regularity, which results in a lower object boundary adherence rate. Hence, Munoz *et al.* propose iterative spanning forest (ISF) to extract superpixels to maintain object boundary adherence [24]. It introduces a mixture seed sampling strategy to adaptively place seed numbers according to region contents, which solves the superpixel segmentation regularity issue via graph-tree theory.

The clustering-based methods initialize seeds with regular grids and grow pixel membership gradually. For instance,

TP [14] locates initial seeds and expands the superpixel border towards gradient flows until reaching a much larger image gradient. Due to introducing a curvature cost to maintain regular superpixels, TP obtains poor segmentation performance under the condition of irregular object and complex background. The most popular superpixel segmentation method, i.e., SLIC [15], uses a fast k-means algorithm to efficiently formulate the superpixel. However, pairwise distance-based k-means based on color and position space do not always perform well for images with weak boundaries. Meanwhile, the k-means optimization framework requires multiple iterations to achieve convergence, which reduces the computation efficiency. Wang *et al.* propose an autonomous attachment superpixel segmentation method, called ALIC [25], via natural continuity between pixels, which not only solves the pixel isolation issue but also improves the operation speed. Xie *et al.* propose a high-precision superpixel segmentation method by adaptively initializing seed number according to image contents and splitting undersegmentation superpixels [26]. Then, Achanta and Słzsstrunk [20] propose a linear time superpixel segmentation method to improve the time cost without any iteration. However, those methods utilize Euclidean distance between pixels and seeds in color and location space, which leads to difficulty in obtaining an efficient balance, especially in the case of complex background and weak boundary. Wang *et al.* [21] introduce the structure-sensitive superpixel (SSS), which considers the structural information of images. It uses geodesic distance computed by the geometric flows instead of simple Euclidean distance. However, the geodesic distance often causes significant error accumulation, which leads to unsatisfactory superpixel segmentation for irregularly blurred object boundaries. Meanwhile, computational efficiency becomes its bottleneck due to measuring the geodesic distance and iterative updating. Recently, Zhao *et al.* and Hu *et al.* propose two superpixel segmentation methods based on raster scanning, i.e., MBS [27] and FLIC [28], to improve computation efficiency. The former uses a coarse-to-fine hierarchical scheme to estimate superpixel centers. However, it introduces minimum barrier distance (MBD) to evaluate pixels' similarity, which results in lower boundary adherence in the case of weak boundary regions. The latter introduces an active search strategy to label neighbor pixels via natural continuity, which results in irregular superpixel regions.

III. PROPOSED METHOD

In this section, we propose a new superpixel segmentation method by forgetting geodesic distance. First, we model the superpixel segmentation as a geodesic label assignment problem. Then, we introduce the forgetting geodesic distance to address the error accumulation in conventional geodesic calculation. Finally, optimized methods via recursive and nonrecursive versions are introduced to efficiently obtain the exact solution.

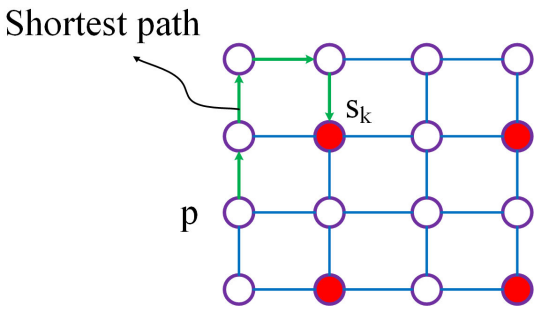

FIGURE 2. Explanation for superpixel generation based on shortest path.

Algorithm 1 Superpixel Algorithm: Recursive Version

Input: image I, initial seed set $S = \{s_1, s_2, \ldots, s_K\}$, $MAX = 10^6$
Output: Superpixel label L

```
for each p in I do
    D(p) = MAX
end for
for each p in I do
    (D(p), L(p)) = Disfun(p, D)
end for
```

A. PROBLEM FORMULATION

Given an image I, we build an undirected weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ based on the four neighborhoods of the pixel. Define vertices as image pixel set $\{p_i\}$, i.e., $\mathcal{V} = \{p_i\}$. Let $S = \{s_1, \ldots, s_K\}$ be the superpixel seeds set, which is a subset of $\mathcal{V}$. K denotes the number of seeds. The edges set consists of adjacent edges between image pixels, i.e., $\mathcal{E} = \{(p_i, p_j)|p_i$ is a neighborhood of $p_j\}$. In this paper, we define $e_*(p_i, p_{i+1})$ as edge weight with a scalar, and $d(p, s_k)$ is denoted as path cost along with $\{p_1 = p, p_2, \ldots, p_n = s_k\}$. We model superpixel segmentation as a geodesic label assignment problem [4], [29]. As shown in Fig. 2, the superpixel label of pixel p is assigned as the seed s_k's indication which has the shortest path from p on graph $\mathcal{V}$:

$$d(p, s_k) = \min_{p_1=p, p_2, \ldots, p_n=s_k} \sum_{i=1}^{n-1} e(p_i, p_{i+1}) \quad s.t.(p_i, p_{i+1}) \in \mathcal{E} \tag{1}$$

where $e(p_i, p_{i+1}) = e_c(p_i, p_{i+1})e_b(p_i, p_{i+1})$ consists of color distance e_c and boundary distance e_b between image adjacent pixels, in which e_b is calculated by structure edge [30].

However, Eq. 1 accumulates the error of all pixels on the geodesic path between pixel p and seed s_k. This will result in inaccurate segmentation. For example, in Fig. 3, the conventional geodesic distance in (b) will accumulate the error from seed to target pixel, which results in higher undersegmentation errors. As shown in (d), the horizontal coordinate denotes the position relationship between target pixel p and seed s. The geodesic distance accumulates the area under the gradient magnitude curve. It can be determined

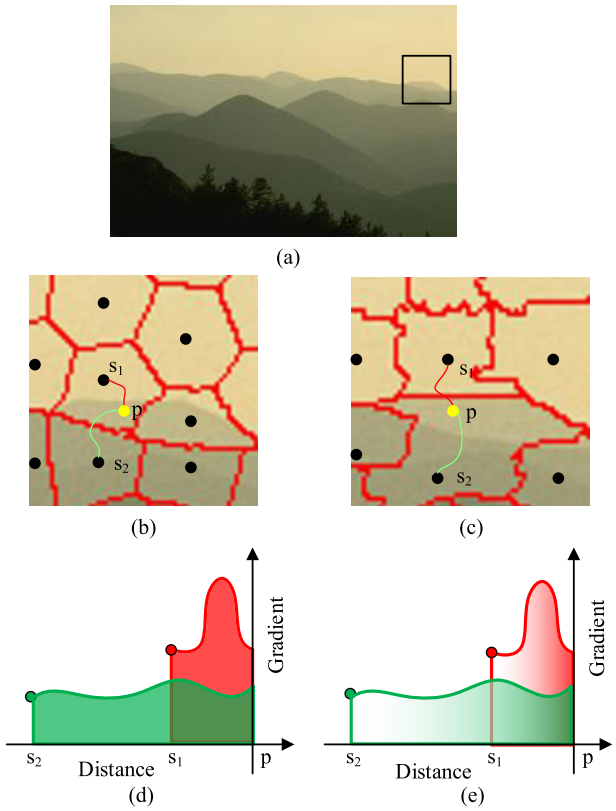

FIGURE 3. Explanation of the conventional geodesic distance and the forgetting geodesic distance. Black points: seeds. Yellow point: target pixel. (a) Original image. (b) Conventional geodesic distance. (c) Forgetting geodesic distance. (d) Error accumulation for (b). (e) Error accumulation for (c).

that pixel p has a shorter geodesic distance to seed s_1 than to seed s_2, although there is a higher gradient magnitude in the object boundary. The gradient accumulation will amplify the magnitude of the geodesic path between pixel p and seed s_2, although it has more smooth gradient values along the path.

Hence, to ameliorate the issue of the error accumulation, we introduce a forgetting factor to weaken the influence of previous error. Specifically, we redefine geodesic distance as follows:

$$d(p_{i+1}, s_k) = \min_{p_1=p,p_2,\ldots,p_n=s_k} \lambda\, d(p_i, s_k) + e(p_i, p_{i+1}) \quad s.t.(p_i, p_{i+1}) \in \mathcal{E},\ \lambda \in (0, 1) \tag{2}$$

The distance devotes much more attention to the similarity of the latest several pixels on the shortest path. Intuitively, the latest pixels on the geodesic path from seed s_k to pixel p could show the adjacent similarity between p and its global context. In other words, the distance weightily aggregates the similarity of pixel p's most similar contextual pixels which lie on the shortest path. Hence, p will be more easily assigned into the similar local region dominated by seed s_2, as shown in Fig. 3 (c). In Fig. 3 (e), the new geodesic distance between p and seed s_2 could be computed as the fading shaded area under the gradient magnitude curve, which is smaller than that between p and seed s_1. Compared with conventional geodesic distance in Fig. 3 (d), the forgetting geodesic distance in Fig. 3 (e) focuses on near gradient magnitude and reduces the error accumulation on the long distance path. This leads to more reasonable label assignents in Fig. 3 (c).

Algorithm 2 *Disfun*(p, D)

```
1:  min = MAX
2:  if p ∈ {s_1, s_2, ..., s_K} then
3:      return (0, L(p))
4:  end if
5:  T = neighbour(p)
6:  for each t_i ∈ T do
7:      (Val, L(t_i)) = Disfun(t_i, D)
8:      D(p) = min{Val + e(p, t_i) * d'(p, s_k), D(p)}
9:      if D(p) < min then
10:         min = D(p)
11:         t* = t_i
12:     end if
13: end for
14: L(p) = L(t*)
15: return (min, L(p))
```

Furthermore, pixel p_{i+1} with larger gradient and larger spatial distance $d'(p_{i+1}, s_k)$ to adjacent seed s_k has a reduced probability to be touched by s_k. We thus modify the recursive formula for geodesic distance as follows:

$$d(p_{i+1}, s_k) = \min_{p_1=p,p_2,\ldots,p_n=s_k} \lambda\, d(p_i, s_k) + e(p_i, p_{i+1})\, d'(p_{i+1}, s_k) \quad s.t.(p_i, p_{i+1}) \in \mathcal{E},\ \lambda \in (0, 1) \tag{3}$$

The formula considers not only the spatial relationship between seed and pixel, but also the local contexture on the geodesic path in terms of the similarity. Hence, it is more robust to blurred boundaries and texture regions. In addition, the pixel tends to be attached by the closer seeds, which could maintain segmentation regularity implicity.

B. OPTIMIZATION

To find the pixel labels, our method computes the shortest distance between pixel p and seed s_k and assigns pixel p with the corresponding label. The direct solution is to use recursive structure for this problem. The distance starting from p to the seed s_k could be calculated as the minimum distance from p's neighbor t to seed s_k by adding edge $e(p, t)$, which falls into the recursive process until computing $D(s_k) = 0$. Hence, each pixel has a different minimum distance to different seed s_k. However, we need not save the distance for the pixel to all seeds s_k: we only maintain the smallest one, which inspires us to follow the recursive algorithms in Alg. 1 and Alg. 2. For each pixel p, we recursively call the *Disfun* function to obtain the smallest distance $D(p)$. Given distance map D, the *Disfun* function could recursively calculate a shortest path between pixel p and seed set $S = \{s_1, s_2, \ldots, s_K\}$. In Alg. 2 line 11, t^* is p's optimal adjacent pixel index with smallest

Algorithm 3 Nonrecursive Version for Superpixel Segmentation Algorithm

Input: image I, initial seed set S, edge map E
Output: Superpixel label L

prior queue Q, distance D, temp label set L', state set M
initialize $M(s_k) = Labeled$, $L(s_k) = k$, other pixels $M(p) = Unlabeled$, $L(p) = 0$
for $k \in \{1, 2, \ldots, K\}$ **do**
  for each pixel p in the four neighborhoods of s_k **do**
    set $L(p) = k$
    use Eq. 3 to obtain $D(p)$
    push $D(p)$ on Q, set $M(p) = waiting$
  end for
end for
while Q is not empty **do**
  Pop Q to obtain $D(p)$
  if $M(p) = Labeled$ **then**
    continue
  end if
  $M(p) = Labeled$
  $L(p) = L'(p)$
  for each pixel t in the four neighborhoods of p **do**
    if $M(t) = Unlabeled$ **then**
      $L(t) = L(p)$
      obtain $D(t) = \lambda * D(p) + e(p, t) * d'(t, s_k)$
      push $D(t)$ on Q, set $M(t) = waiting$
    else $\{M(t) = waiting$ and $D(t) > D(p)\}$
      if $L_p == L_t$; continue; end
      obtain $D'(t)$ from p via $D'(t) = \lambda * D(p) + e(p, t) * d'(t, s_k)$
      if $D'(t) < D(t)$ **then**
        $L(t) = L(p)$
        push $D'(t)$ on Q
      end if
    end if
  end for
end while

distance. Finally, it returns the minimum value $D(p)$ and corresponding label $L(p)$. In other words, besides computing the smallest distance from p to seed s_k, we also return the smallest distance from p to all seed set S, which could be performed by recording the label $L(p)$ and minimum value *min*. In fact, we only calculate the distances to the four nearest seeds to reduce the computational cost.

In recursive processing, the algorithm will call to itself repeatedly, which is achieved by executing on working stack. The program first saves the operating data into the register, then opens up new space to call the recursive function. These operations will need to perform N calls and assign N local variables and N function spaces. It is a substantial burden which is impractical for image pixel-level applications. Hence, we propose a new nonrecursive version to solve the storage burden issues. The algorithm is concluded in Alg. 3.

Intuitively, the recursive version would solve some subproblem repeatedly. For example, to obtain $D(p)$, we should calculate its neighbor $D(t_i)$, $t_i \in \mathbb{T}(p)$. However, $D(t_i)$ would have been solved by another subproblem $D(q)$, $t_i \in \mathbb{T}(q)$ in advance. However, the recursive version cannot avoid making calls to itself again, which results in excessive redundant computation. The proposed nonrecursive version introduces a distance matrix D to record the minimum distance from pixel p to seed set S and introduces logical variable M to record the visited situation of each pixel. On the one hand, we set $M(p) = Labeled$ for the visited pixel, and $M(p) = Unlabeled$ for the unvisited pixel. On the other hand, pixels on the superpixel boundary might be achieved by two adjacent seeds s_i, s_j. Then, we set $M(p) = waiting$ to denote that p is a waiting pixel and needs to be further assigned a new label with smaller seed distance. Specifically, if t is a waiting pixel, we should compare $D(t)$ with $D(p)$ to judge whether the new distance for t from p, i.e., $D'(t) = \lambda\, D(p) + e(p, t)\, d'(t, s_k)$, is smaller than the old one $D(t)$, as shown in Fig. 4. If the distance from p to t is smaller than $D(t)$, we should change t's label to p's, i.e., $L(t) = L(p)$. The new distance computation will only occur for the pixels of superpixel boundaries. Most of the smooth regions have $L(p) == L(t)$ and do not require updating of t's label, which significantly reduces the computational cost.

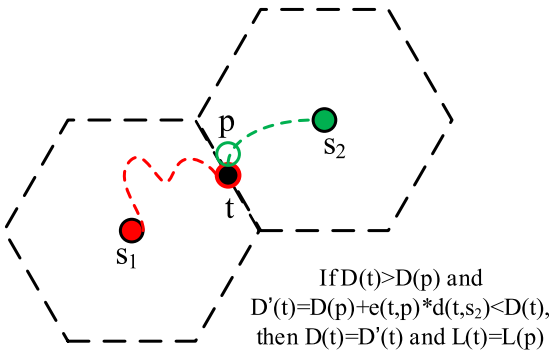

FIGURE 4. Distance calculation for the boundary pixel.

The nonrecursive version uses the bottom-up framework to calculate and record $D(p)$ to reduce computational cost. Specifically, we build a prior queue Q to identify the pixel visiting order. Meanwhile, temporal variable $M(t)$ is set to record whether the pixel is visited or not. The temporary cost $D'(t)$ to record the current path cost is updated when it is visited by some pixel from another path with less cost value. On the other hand, there exists large redundancy in the superpixel interior with regional consistency. In other words, adjacent pixels in the superpixel interior have the same labels, while those in superpixel borders have different labels. We could only compare the boundary pixels' new and old distances to improve computational efficiency. Intuitively, a pixel could be visited by other pixels from four paths, which results in four cost comparison and is obviously

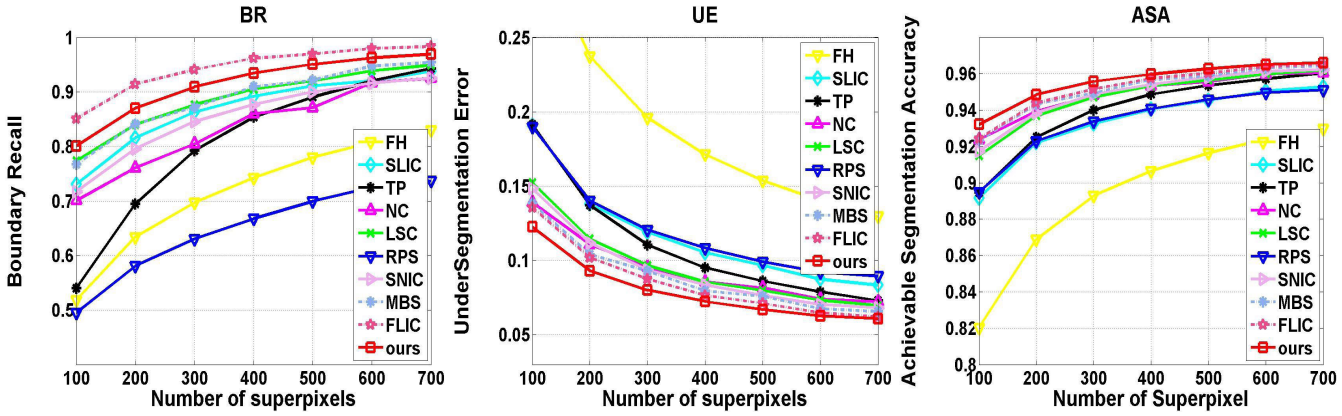

FIGURE 5. Objective results on BSD500 for our method and compared methods in terms of BR, UE and ASA.

time-consuming. Fortunately, we find that pixels in the superpixel interior tend to have the same label. We could then compute and compare one pixel candidate path cost twice or more only when the labels of two neighbor pixels are inconsistent.

SNIC [20] proposes a fast optimization for seed expansion-based superpixel segmentation. It builds a prior queue to maintain the visiting order by corresponding distance value $d_{i,k}$, which represents the distance from pixel i to the k-th centroid. It could obtain the minimum distance of the visited pixel in Euclidean space by continuing to pop the first element of the prior queue. However, calculation of geodesic distance should select the best one from the numerous candidate paths. Visiting each pixel only once in SNIC [20] could not obtain the shortest path for geodesic distance calculation in our situations. Hence, the proposed recursive and nonrecursive algorithms could calculate the shortest geodesic path via candidate pixels' comparison. Intuitively, if a path B between pixel p and s_k is shorter than a previous path A, then we can update the path cost of p and its corresponding label, which is much more probable for boundary pixels. This update is not performed via the SNIC [20] optimization framework because of visiting only once.

IV. EXPERIMENT AND ANALYSIS

We evaluate the proposed method on two datasets. The first one is the BSD500 segmentation dataset, which contains five hundred images of annotated boundary maps as ground truth. The second one is the PASCAL VOC 2012 segmentation challenge (SegVOC12), which contains 2,913 object segmentation maps as ground truth. To obtain an object and comprehensive comparison, we test our method by three different metric measures, which are the common quality indications in existing works. Meanwhile, we compare our method with nine state-of-the-art methods: FH [12], SLIC [15], TP [14], NC [11], LSC [19], RPS [23], SNIC [20], MBS [27] and FLIC [28]. We execute segmentation based on publicly available codes provided by authors. We set the forgetting factor $\lambda = 0.5$.

A. OBJECTIVE RESULTS

We use boundary recall (**BR**), undersegmentation error (**UE**) and achievable segmentation accuracy (**ASA**) [31] to measure the segmentation performance. We use $\{G_i | i = 1, 2, \cdots, M\}$ to denote the segmentation of ground truth, and use $\{S_j | j = 1, 2, \cdots, K\}$ to represent results by superpixel algorithms. Three indications are defined as follows:

$$BR = \frac{\sum_{i \in GT_b} logical(\min_{j \in SG_b} \parallel x_i - x_j \parallel_2 < 2)}{GT_b} \tag{4}$$

where GT_b and SG_b represent the boundary results of ground truth and superpixels. $\parallel x_i - x_j \parallel_2$ is defined as 2-norm to denote the location difference between pixel i and pixel j.

$$UE(S) = \frac{[\sum_{\{G_i | S_j \cap G_i \neq \emptyset\}} - Area(G_i)] - Area(G_i)}{Area(G_i)} \tag{5}$$

$$ASA = \frac{\sum_j arg\max_i |S_j \cap G_i|}{\sum_i |G_i|} \tag{6}$$

where $|G_i|$ is defined as the pixel number contained in region G_i.

Specifically, **BR** measures the overlap of superpixel boundaries with ground truth boundaries. Effective segmentation performance leads to a higher **BR**. **UE** measures the percentage of leakage regions for superpixel cross ground truth boundaries. If fewer superpixels cross over different ground truth objects, **UE** is lower. **ASA** indicates the ratio of superpixels which are not leaked out of ground truth boundaries by labeling each superpixel with the ground truth's largest overlap area. The higher the value is, the better the superpixels comply with the object.

We first evaluate our method and compared methods on the BSD500 dataset. The objective results are shown in Fig. 5. In terms of **BR**, our method outperforms other state-of-the-art methods except FLIC [28]. It can be seen that our method performs better than those competing methods, such as LSC [19] and SNIC [20]. The reason is that our method pays much more attention to the latest pixels' similarity by forgetting early path influence, which benefits object boundary segmentation. LSC [19] and SNIC [20] also follow

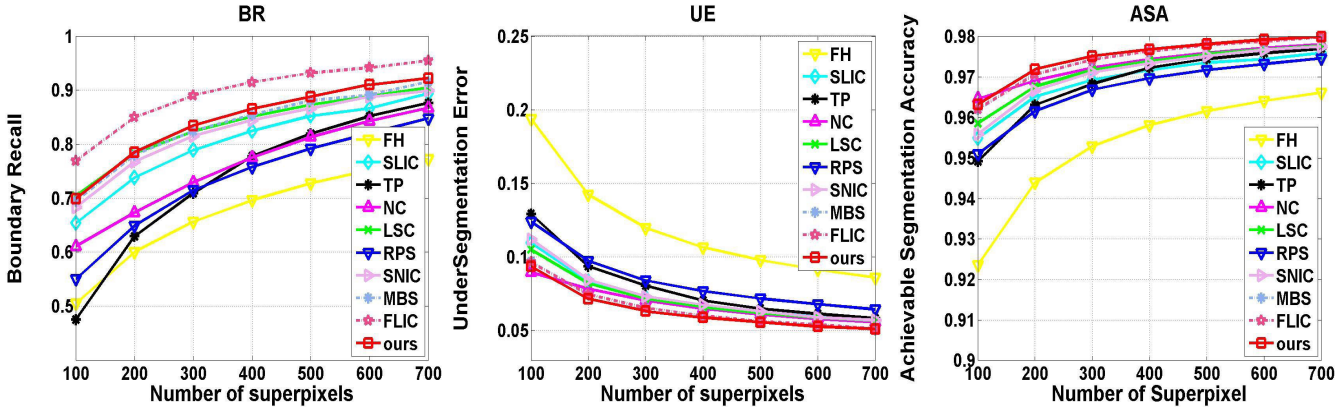

FIGURE 6. Objective results on VOC2012 for our method and compared methods in terms of BR, UE and ASA.

SLIC's framework to balance the color and space distance between cluster centers and pixels for their local neighbors. MBS [27] introduces minimum barrier distance (MBD) to evaluate the similarity between pixel and seeds. Although this method utilizes the difference between the pixel's maximum intensity and minimum intensity to evaluate similarity, it tends to suffer from undersegmentation in the case of weak boundary regions. Meanwhile, it does not explicitly enforce the pixel connectivity, which also results in inferior segmentation performance. FLIC [28] shows the best boundary adherence rate. Each pixel actively searches its neighbor pixels' labels via natural continuity, which avoids regional disjointness and limited searching range. These two factors improve the boundary adherence rate and allow first place to be achieved. However, pixel continuity without any constraint also results in the worst performance in shape uniformity, which is sensitive to maintaining compact local regions. This results in inferior evaluation in terms of **UE** and **ASA**.

For **UE**, the lower the value is, the more objects could be correctly recognized. It can be seen that our method also produces better performance compared with LSC [19] and SNIC [20]. In this comparison, NC [11] achieves the second highest error rate. The reason can be explained as follows: **UE** measures the superpixel region leaking from the ground truth boundary. Although NC [11], LSC [19] and SNIC [20] could generate more regularity superpixels, they have much lower boundary recall, which limits them to obtaining lower **UE**. This aspect also validates the advantages of our method with respect to extracting superpixel boundaries. It can be determined that RPS [23] also obtains regularity superpixel regions. However, the authors consider only local path connection for superpixel junctions to preserve superpixel regularity, but ignore the junction connection with diagonal crossover, which results in higher regional leaking. Moreover, although FLIC [28] achieves the highest boundary recall, the significant irregular regions result in higher **UE**. This is because even if a superpixel with irregular shape has reduced underfitting boundaries, it will yield high regional leaking. In the same **BR**, the less the regularity is, the greater the regional leaking.

In terms of **ASA**, it labels superpixels as the ground truth with the largest overlap area. On the one hand, regular superpixels tend to produce larger overlap ratios and lead to higher performance. On the other hand, higher boundary adherence could also achieve higher **ASA**. Our method obtains the highest performance compared with all state-of-the-art methods. It also demonstrates that the proposed method could coordinate the balance between region regularity and boundary adherence.

In addition, we test the performances of our method and the state-of-the-art approaches on SegVOC12. The objective results are shown in Fig. 6. The results suggest that our method obtains similar performance to the results for BSD500. FLIC [28] also obtains the highest score in terms of **BR**. In this experience, NC [11] obtains the best **ASA** and **UE** with relatively small numbers of superpixels. It considers intraclass similarity and interclass dissimilarity to generate regularly shaped superpixel regions, which could result in decreased regional leaking in regions with complex texture. Along with the increasing number of superpixels, the proposed method more easily fits the object boundary by the forgetting mechanism to focus on latest pixels' similarity of the geodesic path, i.e., fitting for multiple seeds could eliminate the ambiguity of pixels' assignment in blurred boundaries. The same phenomenon as for the results on BSD500 is also shown in Fig. 6 (a).

B. SUBJECTIVE RESULTS

In this section, we list some subjective results in Fig. 7 to compare our method with other state-of-the-art approaches. It can be determined that our method could not only extract the object boundaries but also obtain the regular segmentation regions. For example, the case of the dog in the highly similar background is very challenging. Our method could obtain the accurate segmentation boundary, especially for the right leg. Other methods obtain lower boundary adherence because of heavy regular constraints, such as methods [11], [14], [23].

FIGURE 7. Some subjective results of compared methods. From top to bottom: our method, NC, RPS, TP, SLIC, SNIC, LSC, MBS, FLIC.

Although LSC [19] obtains an accurate segmentation boundary, it yields highly irregular segmentation regions. In the second image, the roof of the building is difficult to segment because of the highly similar appearance with

TABLE 1. Average running time (seconds per image) for 100 superpixels on the BSD500 dataset.

Method	Ours	LSC	SLIC	SNIC	NC	TP	RPS	MBS	FLIC
Complexity	$O(n)$	$O(n)$	$O(nKT)$	$O(n)$	$O(n^{\frac{3}{2}})$	$O(n)$	$O(Knlogn)$	$O(n)$	$O(n)$
Time (s)	0.21	0.33	0.31	0.10	44.73	5.46	13.6	0.12	0.07
Code	MATLAB + C	C++	C++	C++	MATLAB + C	MATLAB + C	MATLAB + C	C++	C++

the sky. It is challenging for existing methods to capture the weak boundary. The regularity constraint of NC [11] results in low boundary adherence. RPS [23] finds the segmentation boundary via strong gradient edge, which easily follows the noise edges. SLIC and SNIC update their seed locations and balance color and location to obtain uniform segmentation regions. However, due to the Euclidean distance computation, they cannot accurately segment object parts, such as pillars. Meanwhile, the weak roof boundary results in low boundary adherence for MBS [27] and FLIC [28]. Our method calculates the geodesic distance to focus on the object boundary and extract the pillar boundaries. LSC [19] is the most competitive method, which obtains high boundary adherence but yields rough boundary results for the inner texture regions. For the bear image, the head and back boundaries of the middle bear are difficult tasks for all state-of-the-art methods. However, our method successfully extracts the complete boundaries for the bears. For the most difficult example, the rear of the right sheep in the last image has highly complex background noise and a weak region edge. All of the state-of-the-art methods failed in segmentation of the rear boundary. However, our method extracts complete object boundaries, which validates our method's effectiveness and robustness to the complex texture and weak boundary condition.

C. TIME COMPLEXITY

Our method is a linear time $O(n)$ superpixel segmentation method. We run all compared methods and our method on the same PC platform with an Intel Core i5 1.6 GHz CPU with a single-core implementation. The comparison algorithms are implemented from the publicly available source codes provided by the author web pages with default parameters. For a fair comparison, all algorithms are executed on the BSD500 benchmark to obtain the corresponding average time cost in the case of 100 superpixels. Our method requires an average of 0.21 s in total for individual images with size of 481 $\times$ 321 on the BSD500 benchmark. The most time consuming aspect is generation of the structure edge [30], which requires 110 ms. The segmentation step for all image pixels requires 100 ms, which validates our method's high efficiency.

Tab. 1 shows the average runtime for 100 superpixels with images of size 481 $\times$321 for compared methods. Most methods are subject to a linear time $O(n)$ comparison. NC [11] and TP [14] are time consuming with respect to their polynomial time complexity and multiple iterations. MBS [27], FLIC [28] and SNIC [20] achieve the top three results. The first two both use back-and-forth scanning to traverse each pixel twice, which makes them both $O(n)$ linear complexity algorithms. They require image and cost average times of 0.12 s and 0.07 s, respectively. Although MBS [27] introduces MBD distance to reduce computation cost, it uses a hierarchical scheme to update and propagate centers, which results in external time cost. Our method obtains the fourth order due to the calculation of structure edge [30]. Although LSC [19] is also a linear time algorithm, it inherits the limitations of SLIC [15], i.e., multiple iterations and centers updating after member assignments, which reduce its efficiency. In contrast, our method uses a prior queue for visited pixels. Meanwhile, the forgetting geodesic path exhibits reduced dependence on seeds as the path extends, which does not require any seed updating or iterations.

V. CONCLUSION

This paper proposes a new geodesic distance, called forgetting geodesic distance, to rapidly achieve superpixel segmentation. In contrast with the conventional geodesic distance, the forgetting geodesic distance reduces the influence of previous path cost but focuses on latest pixel difference. Intuitively, a pixel's cost will be updated when it is visited by some pixel from another path with less cost value. In the new path, the contribution of the gradient difference of previous pixels becomes smaller, and the new geodesic path could relieve the significant error accumulation. Meanwhile, the pixels are aggregated with less dependence on seeds as the path extends, which avoids the seed updating. Experimental results validate that the proposed method obtains 2 percent and 1 percent average gains compared with most of the state-of-the-art methods in terms of BSD500 and VOC2012 datasets, respectively. However, focusing on latest pixel difference, our method readily results in superpixel area imbalance. In the future, we plan to improve our results by exploring more efficient area constraints, and further extend our method to adaptive image content.

REFERENCES

[1] C. Ge, K. Fu, F. Liu, L. Bai, and J. Yang, "Co-saliency detection via inter and intra saliency propagation," *Signal Process., Image Commun.*, vol. 44, pp. 69–83, May 2016.
[2] W. Qi, J. Han, Y. Zhang, and L.-F. Bai, "Graph-Boolean map for salient object detection," *Signal Process., Image Commun.*, vol. 49, pp. 9–16, Nov. 2016.
[3] S. Manen, M. Guillaumin, and L. V. Gool, "Prime object proposals with randomized prim's algorithm," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2536–2543.
[4] B. Luo, H. Li, T. Song, and C. Huang, "Object segmentation from long video sequences," in *Proc. 23rd ACM Int. Conf. Multimedia (MM)*, 2015, pp. 1187–1190.
[5] B. Fulkerson, A. Vedaldi, and S. Soatto, "Class segmentation and object localization with superpixel neighborhoods," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 670–677.

[6] F. Meng, H. Li, Q. Wu, B. Luo, C. Huang, and K. N. Ngan, "Globally measuring the similarity of superpixels by binary edge maps for superpixel clustering," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 4, pp. 906–919, Apr. 2018.
[7] F. Meng, H. Li, Q. Wu, K. N. Ngan, and J. Cai, "Seeds-based part segmentation by seeds propagation and region convexity decomposition," *IEEE Trans. Multimedia*, vol. 20, no. 2, pp. 310–322, Feb. 2018.
[8] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 154–171, Sep. 2013.
[9] G. Mori, X. Ren, A. A. Efros, and J. Malik, "Recovering human body configurations: Combining segmentation and recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun./Jul. 2004, pp. II-326–II-333.
[10] L. Yang, G. Yang, X. Xi, X. Meng, C. Zhang, and Y. Yin, "Tri-branch vein structure assisted finger vein recognition," *IEEE Access*, vol. 5, pp. 21020–21028, 2017.
[11] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.
[12] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181, Sep. 2004.
[13] A. P. Moore, S. J. D. Prince, J. Warrell, U. Mohammed, and G. Jones, "Superpixel lattices," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.
[14] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi, "TurboPixels: Fast superpixels using geometric flows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2290–2297, Dec. 2009.
[15] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.
[16] J. Shen, Y. Du, W. Wang, and X. Li, "Lazy random walks for superpixel segmentation," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1451–1462, Apr. 2014.
[17] J. Peng, J. Shen, A. Yao, and X. Li, "Superpixel optimization using higher order energy," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 5, pp. 917–927, May 2016.
[18] Y. Zhang, X. Li, X. Gao, and C. Zhang, "A simple algorithm of superpixel segmentation with boundary constraint," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 7, pp. 1502–1514, Jul. 2017.
[19] Z. Li and J. Chen, "Superpixel segmentation using linear spectral clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1356–1363.
[20] R. Achanta and S. Susstrunk, "Superpixels and polygons using simple non-iterative clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4895–4904.
[21] P. Wang, G. Zeng, R. Gan, J. Wang, and H. Zha, "Structure-sensitive superpixels via geodesic distance," *Int. J. Comput. Vis.*, vol. 103, no. 1, pp. 1–21, May 2013.
[22] A. P. Moore, S. J. Prince, and J. Warrell, "Lattice cut constructing superpixels using layer constraints," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 2117–2124.
[23] H. Fu, X. Cao, D. Tang, Y. Han, and D. Xu, "Regularity preserved superpixels and supervoxels," *IEEE Trans. Multimedia*, vol. 16, no. 4, pp. 1165–1175, Jun. 2014.
[24] J. E. Vargas-Munoz, A. S. Chowdhury, E. B. Alexandre, F. L. Galvao, P. A. V. Miranda, and A. X. Falcao, "An iterative spanning forest framework for superpixel segmentation," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3477–3489, Jul. 2019.
[25] J. Wang, H. Shi, M. Liu, L. Cai, C. Ke, and B. Deng, "Alic: A superpixel segmentation algorithm based on autonomous attachment," in *Proc. 3rd Int. Conf. Comput. Sci. Appl. Eng.*, 2019, pp. 1–6.
[26] X. Xie, G. Xie, X. Xu, and L. Cui, "Adaptive high-precision superpixel segmentation," *Multimedia Tools Appl.*, vol. 78, no. 9, pp. 12353–12371, May 2019.
[27] Y. Hu, Y. Li, R. Song, P. Rao, and Y. Wang, "Minimum barrier superpixel segmentation," *Image Vis. Comput.*, vol. 70, pp. 1–10, Feb. 2018.
[28] J. Zhao, B. Ren, Q. Hou, and M.-M. Cheng, "Flic: Fast linear iterative clustering with active search," in *Proc. AAAI*, 2018, pp. 1–8.
[29] B. Luo, H. Li, F. Meng, Q. Wu, and C. Huang, "Video object segmentation via global consistency aware query strategy," *IEEE Trans. Multimedia*, vol. 19, no. 7, pp. 1482–1493, Jul. 2017.
[30] P. Dollár and C. L. Zitnick, "Fast edge detection using structured forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 8, pp. 1558–1570, Aug. 2015.
[31] M. Wang, X. Liu, Y. Gao, X. Ma, and N. Q. Soomro, "Superpixel segmentation: A benchmark," *Signal Process., Image Commun.*, vol. 56, pp. 28–39, Aug. 2017.

BING LUO received the Ph.D. degree in signal and information processing from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2017. He is currently a Lecturer with the School of Computer and Software Engineering, Xihua University, Chengdu, Sichuan. His research interests include machine learning, image segmentation, and object detection.

JUNKAI XIONG is currently pursuing the master's degree in computer science and technology with the School of Computer and Software Engineering, Xihua University, Sichuan, China. His research interests include computer vision, image processing, and so on.

LI XU received the Ph.D. degree from Southwest Petroleum University, Chengdu, China, in 2015. From 2016 to 2018, she was a Postdoctoral Fellow with Southwest Jiaotong University, Chengdu. She is currently a Lecturer with the School of Science, Xihua University, Chengdu. Her research interests include image segmentation and machine learning.