HAL
open science

Generalized Shortest Path-based Superpixels for Accurate Segmentation of Spherical Images

Rémi Giraud, Rodrigo Borba Pinheiro, Yannick Berthoumieu

To cite this version:

Rémi Giraud, Rodrigo Borba Pinheiro, Yannick Berthoumieu. Generalized Shortest Path-based Superpixels for Accurate Segmentation of Spherical Images. International Conference on Pattern Recognition (ICPR 2020), Jan 2021, Milan, Italy. hal-02539465

HAL Id: hal-02539465

https://hal.science/hal-02539465v1

Submitted on 16 Apr 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Generalized Shortest Path-based Superpixels for Accurate Segmentation of Spherical Images

Rémi Giraud, Rodrigo Borba Pinheiro, Yannick Berthoumieu
Bordeaux INP, Univ. Bordeaux, CNRS, IMS, UMR 5218
F-33400 Talence, France
remi.giraud@ims-bordeaux.fr, rodrigo.borba_pinheiro@bordeaux-inp.fr, yannick.berthoumieu@ims-bordeaux.fr

***Abstract*—Most of existing superpixel methods are designed to segment standard planar images as pre-processing for computer vision pipelines. Nevertheless, the increasing number of applications based on wide angle capture devices, mainly generating 360° spherical images, have enforced the need for dedicated superpixel approaches. In this paper, we introduce a new superpixel method for spherical images called SphSPS (for Spherical Shortest Path-based Superpixels). Our approach respects the spherical geometry and generalizes the notion of shortest path between a pixel and a superpixel center on the 3D spherical acquisition space. We show that the feature information on such path can be efficiently integrated into our clustering framework and jointly improves the respect of object contours and the shape regularity. To relevantly evaluate this last aspect in the spherical space, we also generalize a planar global regularity metric. Finally, the proposed SphSPS method obtains significantly better performances than both planar and spherical recent superpixel approaches on the reference 360° spherical panorama segmentation dataset.**

***Index Terms*—Superpixels, Spherical images, Regularity**

I. Introduction

The growing in resolution and quantity of image data has highlighted the need for efficient under representations to reduce the computational load of computer vision pipelines. In this context, superpixels were popularized with [1] to reduce the image domain to irregular regions having approximately the same size and homogeneous colors. Contrary to regular multi-resolution schemes, a result at the superpixel scale can be very close to the optimal one at the pixel scale. Superpixels have been successfully used in many applications such as: semantic segmentation [2], [3], optical flow estimation [4] or style transfer [5]. The main issue to deal with is the irregularity between all regions that may prevent from using the standard neighborhood-based tools. Nevertheless, this issue has been addressed in graph-based approaches [6], using neighborhood structure [7], or within deep learning frameworks [8].

At the same time, the use of new acquisition devices capturing wide angles, such as fish eyes, generally covering a 360° field of view has become more and more popular. These devices offer a global capture of the environment, particularly interesting for applications such as autonomous driving. With a depth-aware system, the intensity can be projected on a 3D point cloud. Otherwise, the image sphere is generally projected on a discrete 2D plane to generate an equirectangular image inducing distorsions [9]. In this context, several works, *e.g.*, [10], [11] have used standard planar superpixels although they do not consider the geometry distorsions in the equirectangular image, that may limit the segmentation accuracy and their interpretation on the spherical acquisition space [12].

Many superpixel approaches have been proposed over the years, most exclusively to segment standard planar images. These methods use watershed [13], region growing [14], eikonal-based [15], graph-based energy [16], or even coarse-to-fine algorithms [17]. A significant breakthrough was obtained with the SLIC method [1], locally adapting a K-means algorithm on a trade-off between distances in the spatial and CIELab color space to generate superpixels. The method has few parameters and a low processing time, but is limited to adjust to different image content and to accurately capture object borders. Many improvements of SLIC have been proposed using boundary constraint [18], advanced feature space [19], non-iterative clustering [20], a shortest path approach [21], or even deep learning processes [8] although these last methods present the usual limitations, in terms of resources, training time, large dataset needed, and applicability to other images.

For spherical images, The unsupervised segmentation approach of [22] has been extended in [23], but generates very irregular regions, not considered as superpixels. More recently, the SLIC method was extended to produce spherically regular superpixels [12]. Pixels are projected on the unit sphere for computing the spatial constraints and produce regular superpixels in the spherical space. Besides the display interest, the respect of the acquisition space geometry enables to more accurately segment the image objects [12]. Nevertheless, this approach comes with the same limitations as SLIC, *i.e.*, limited adaptability to different contexts with severe non robustness to textures or noise due to the use of a standard color feature space, and no explicit integration of contour information. These limitations are addressed in [21], for which authors obtain significantly higher accuracy for standard planar image segmentation by considering the color and contour features along the shortest path between the pixel and the superpixel.

Contributions: In this paper, we address the limitations of the spherical approach of [12], by proposing in Section II a new superpixel method called SphSPS (Spherical Shortest Path-based Superpixels). SphSPS is based on the same spherical K-means approach of [12] but exploites more advanced features [19] and generalizes the notion of shortest path [21], to the acquisition space, here the spherical one. To this end, a dedicated fast shortest path algorithm is defined to integrate the information of this large number of pixels into the method.

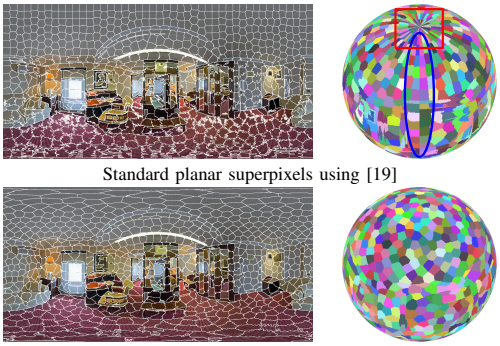


Fig. 1. Example of superpixel segmentation with a planar [19] and the proposed spherical SphSPS method on a 360° panorama image. SphSPS provides accurate superpixels that are regular in the spherical acquisition space (red square) and connected at horizontal boundaries (blue ellipse).

SphSPS generates in very limited processing time accurate and regular spherical superpixels (see Figure 1). To relevantly evaluate the regularity aspect in the spherical space, we also propose a generalization of the global regularity measure [24] (Section III). SphSPS obtains higher segmentation performance than the state-of-the-art methods on the reference 360° spherical panorama segmentation dataset [25] (Section IV).

II. Spherical Shortest Path-based Superpixels

To introduce SphSPS, we first present the K-means method [1] (Section II-A) and its spherical adaptation [12] (Section II-B). Then, we present the feature extraction method on a planar shortest path [21] (Section II-C1) and our generalization to the spherical space (Sections II-C2 and II-C3).

A. Planar K-means Iterative Clustering

SphSPS is based on the SLIC algorithm [1] using an iteratively constrained K-means clustering of pixels. Superpixels S_i are first initialized over the image as blocks of size $s \times s$, described by the average CIELab colors C_{S_i} and barycenter position $X_{S_i} = [x_i, y_i]$ of pixels in S_i. The clustering for each pixel p relies on a color $d_c(p, S_i) = \|C_p - C_{S_i}\|_2^2$, and a spatial distance $d_s(p, S_i) = \|X_p - X_{S_i}\|_2^2$. At each iteration, each superpixel S_i is compared to all pixels $p = [C_p, X_p]$, of color C_p at position X_p, within a $(2s+1)\times(2s+1)$ area A_i around its barycenter X_{S_i}. A pixel p is associated to the superpixel minimizing the distance D defined as:

$$D(p, S_i) = d_c(p, S_i) + d_s(p, S_i)\frac{m^2}{s^2}, \quad (1)$$

with m, the trade-off parameter setting the shape regularity. Finally, a post-processing step ensures the region connectivity.

B. Spherical Geometry

In the spherical acquisition space, vertical and horizontal coordinates are respectively projected to the meridians and circles of latitude, so the spherical image has a width twice superior to its height. SphSPS is based on the same adaptation of the planar K-means method to the spherical geometry as [12], that requires three steps. The first one is the initialization of the K superpixels. To spread the barycenters along the sphere, we also use the Hammersley sampling [26]. The second step is the search area that must consider the proximity of pixels in the spherical space. For instance, superpixels on the image top and bottom have larger search areas. This area A_i, is defined for each superpixel S_i of barycenter $X_{S_i} = [x_i, y_i]$ as:

$$A_i = \{[x, y] | x_i - \frac{S}{\sin\phi} \leq x \leq x_i + \frac{S}{\sin\phi}, y_i - S \leq y \leq y + S\}, \quad (2)$$

with $\phi = y\pi/h$ the polar angle corresponding to the y-th row for an image of height h and width w, and the average superpixel size $S = w/\sqrt{K\pi}$. The 360° geometry aspect must also be handled to horizontally connect the pixels. This is done with a mirror effect when the search region falls outside the image boundaries [12]. The third aspect is the computation of the spatial distance, that must also be done in the spherical space. For each image pixel $X = [x, y]$ the projection on the 3D acquisition space $X^a = [x^a, y^a, z^a]$ is computed as:

$$\begin{cases} x^a = \sin(\frac{y\pi}{h})\cos(\frac{2x\pi}{w}) \\ y^a = \sin(\frac{y\pi}{h})\sin(\frac{2x\pi}{w}) \\ z^a = \cos(\frac{y\pi}{h}) \end{cases} \leftrightarrow \begin{cases} x = \lfloor \frac{\arctan2(y^a, x^a)w}{2\pi} \rfloor \\ y = \lfloor \frac{\arccos(z^a)h}{\pi} \rfloor \end{cases}. \quad (3)$$

Note that $x \in [-\frac{w}{2}, \frac{w}{2}]$, when computed from X^a, so we map x on the image domain with $x \leftarrow x + w$, if $x \leq 0$.

The most straightforward 3D spatial distance is the Euclidean one $d_s(X_p^a, X_{S_i}^a) = \|X_p^a - X_{S_i}^a\|_2^2$. SphSPS uses the spherical and computationally costless cosine dissimilarity distance proposed in [12] as $d_s(X_p^a, X_{S_i}^a) = 1 - \langle X_p^a, X_{S_i}^a \rangle$. Note that with adjusted parameter m (1), both distances can achieve almost similar performances for [12] (see Section IV).

C. Generalized Shortest Path Method

1) Feature extraction on a shortest path: In [21], color and contour information of pixels q on the planar shortest path $\mathbf{P}_{p,S_i}$ between a pixel p and a superpixel S_i are used to improve segmentation accuracy and regularity. SphSPS also integrates these features and has the same clustering distance D than [21]. Nevertheless, in the following, the shortest path $\mathbf{P}_{p,S_i}$ differs since we compute it in the spherical space.

First, to relevantly increase the regularity and prevent non convex shapes to appear, the color distance of the pixels on the path is added to the color distance d_c such that:

$$d_c(p, S_i, \mathbf{P}_{p,S_i}) = \lambda d_c(p, S_i) + \frac{1-\lambda}{|\mathbf{P}_{p,S_i}|}\sum_{q \in \mathbf{P}_{p,S_i}} d_c(q, S_i), \quad (4)$$

with λ a trade-off parameter usually set to 0.5.

The contour information can also be considered to increase the respect of objects borders using a contour map C, with values between 0 and 1. A contour term d_C is defined as:

$$d_C(\mathbf{P}_{p,S_i}) = 1 + \gamma \max_{q \in \mathbf{P}_{p,S_i}} C(q), \quad (5)$$

with γ the parameter penalizing the crossing of a contour.

The final clustering distance of SphSPS is defined as:

$$D(p, S_i) = \left(d_c(p, S_i, \mathbf{P}_{p,S_i}) + d_s(p, S_i)\frac{m^2}{s^2}\right) d_C(\mathbf{P}_{p,S_i}), \quad (6)$$

with the spherical spatial distance d_s using the cosine dissimilarity as $d_s(X_p^a, X_{S_i}^a) = 1 - \langle X_p^a, X_{S_i}^a \rangle$ [12], and $\mathbf{P}_{p,S_i}$ the proposed spherical shortest path computed as follows.

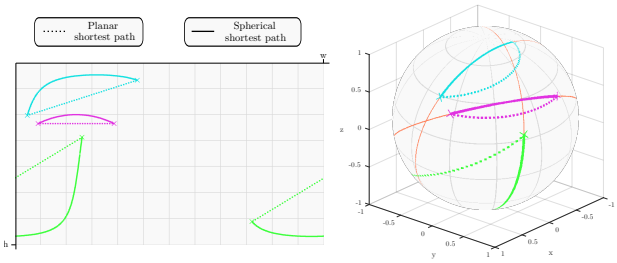


Fig. 2. Examples of planar (dotted lines) and spherical shortest path (full lines) between points in the 2D image space (left) and 3D acquisition space (right). The spherical path follows the shortest geodesic path on the sphere.

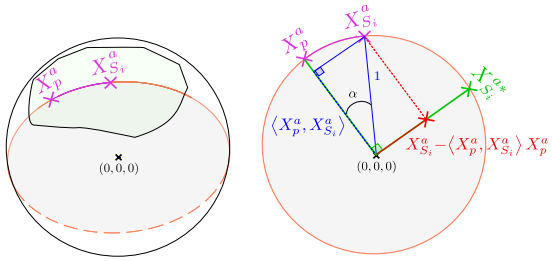


Fig. 3. Computation of the spherical shortest path. The orthogonal coordinate system $[\vec{X_p^a}, \vec{X_{S_i}^{a*}}]$ is computed from projection of $X_{S_i}^a$ on X_p^a (8). The angle α between the positions is then is used to sample 3D points on the path (9).

2) Generalized shortest path: In Figure 2, we compare shortest paths in the planar space, as in [21], and in the spherical one as in SphSPS. With planar images, since no distorsions are introduced between the acquisition and the image space ($\mathbb{N}^2$), they are considered equivalent. Hence, the shortest path reduces to a linear path and can be easily computed with a discrete algorithm [27]. Nevertheless, in general, the shortest path should be computed in the acquisition space, than can be spherical or even circular using fisheyes with different capture angles. Hence, the generalized formulation of the shortest path problem computes it in the acquisition space ($\mathbf{P}_{p,S_i}^a$) and projects it back to the planar image space:

$$\mathbf{P}_{p,S_i} = \mathbf{P}_{p,S_i}^a \overrightarrow{\text{proj}} \{\mathbb{N}^2\}. \tag{7}$$

3) Shortest path in the spherical space: The spherical shortest path consists in following the geodesic along the sphere [28], lying on a *great circle* (in orange color in Figures 2 and 3), containing the two points and the sphere center. Tangential methods to extract way-points on the great circle have been formalized for instance in [29]. Nevertheless, such theoretical approaches use many trigonometric computations that impact the performance. In the following we propose a simpler reformulation of the spherical geodesic path problem.

Fast geodesic path implementation: For each comparison of a pixel at X_p^a to a superpixel of barycenter $X_{S_i}^a$, we propose to first compute an orthogonal coordinate system $[\vec{X_p^a}, \vec{X_{S_i}^{a*}}]$ within their great circle. To build such system, we perform an orthogonalization process to get the position $X_{S_i}^{a*}$, creating an orthogonal vector to X_p^a within the great circle such as:

$$X_{S_i}^{a}{}^{*} = \frac{X_{S_i}^a - \left\langle X_p^a, X_{S_i}^a \right\rangle X_p^a}{\left\| X_{S_i}^a - \left\langle X_p^a, X_{S_i}^a \right\rangle X_p^a \right\|_2}, \tag{8}$$

with the scalar product $\left\langle X_p^a, X_{S_i}^a \right\rangle$ already computed for the spatial distance d_s. Then, the angle between the two points is simply obtained with $\alpha = \arccos\left(\left\langle X_p^a, X_{S_i}^a \right\rangle\right)$. Finally, the geodesic path $\mathbf{P}_{p,S_i}^a$ is defined within $[\vec{X_p^a}, \vec{X_{S_i}^{a*}}]$, by starting from the pixel position, and linearly increasing the angle shift from 0 to α, to reach the superpixel barycenter such as:

$$\mathbf{P}_{p,S_i}^a = \cos(\alpha_{\mathbf{N}}) X_p^a + \sin(\alpha_{\mathbf{N}}) X_{S_i}^{a*}, \tag{9}$$

with $\alpha_{\mathbf{N}} = \frac{[0,N-1]}{N-1}\alpha \in \mathbb{R}^N$, intermediate angles to linearly sample N points between the two positions. The geodesic path is finally projected in the planar space (3) to get $\mathbf{P}_{p,S_i}$ (7). By this way, we obtain the shortest spherical path coordinates with simple calculations, dividing the processing time by a factor 2 compared to tangential approaches. An example of spherical shortest path on a great circle with the the computation of the corresponding coordinate system is illustrated in Figure 3.

Optimization: First, for each superpixel, we can store the color distance computed to each tested pixel, reducing the processing time by 50%. Then, contrary to the planar linear path algorithm [27] we can exploit path redundancy. If the path of a pixel to a superpixel crosses a previously computed path to the same superpixel, the rest of the path should be the same since they lie on the same great circle. So we can also store the average color and contour information on the path for each crossed pixel. This is done efficiently using recursive implementation. By this way, for many pixels we are able to directly access the large quantity of information contained in the shortest path, again reducing the processing time by 50%.

III. Generalized Global Regularity Measure

Superpixels tend to optimize a color and spatial trade-off, so metrics should mainly evaluate object segmentation and regularity performances. This last aspect has been sparsely evaluated although most methods have a regularity parameter that may significantly impact superpixel-based pipelines. Moreover, the standard compactness metric [30], which is the only one extended to the spherical space [12] was proven very limited [24]. In this section, we propose a new way to relevantly evaluate the regularity in the acquisition space.

A. Limitation of the Compactness Measure

In [12], the compactness measure C [30] is extended to the spherical case. The regularity of a segmentation $\mathcal{S} = \{S_i\}$ is only seen as a notion of circularity, computed as:

$$\text{C} = \frac{1}{\sum_{S_i \in \mathcal{S}} |S_i|} \sum_{S_i \in \mathcal{S}} Q(S_i)|S_i|, \tag{10}$$

with $Q(S_i)$=$(4\pi|S_i| - |S_i|^2)/|P(S_i)|^2$ the spherical isoperimetric quotient [31]. Hence, each superpixel is independently compared to a circular shape, such that for instance, ellipses can have higher C measures than squares. In [24], this metric has been proven highly sensitive to boundary noise and inconsistent with the superpixel size. Moreover, in [12] it even fails to differentiate spherical and planar-based methods.

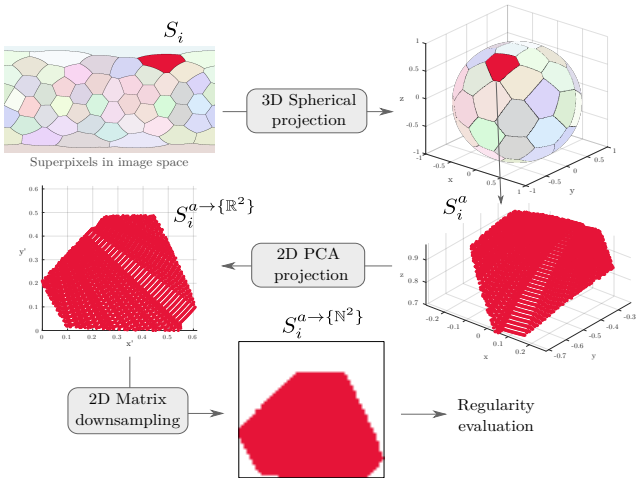


Fig. 4. Illustration of the projection process of the proposed Generalized Global Regularity (G-GR) metric (12). A superpixel shape S_i is projected in the acquisition space (S_i^a), then on a two dimensional one using a PCA analysis ($S_i^{a\to\{\mathbb{R}^2\}}$), then downsampled to generate a 2D matrix ($S_i^{a\to\{\mathbb{N}^2\}}$), allowing for instance to compute a convex hull to measure its regularity.

B. Generalized Global Regularity Metric

1) Global regularity metric: In [24], a global regularity metric (GR) is introduced, to address the issues of the compactness. First, the Shape Regularity Criteria (SRC) is defined to robustly evaluate the convexity, the contour smoothness, and the 2D balanced repartition of each superpixel. Convexity and smoothness properties are computed with respect to the discrete convex hull containing the shape.

As for the compactness C (10), SRC is independently computed for each superpixel, so [24] also introduces a Smooth Matching Factor (SMF) to evaluate the consistency of superpixel shapes. Each superpixel is compared, after registration on its barycenter, to the average superpixel shape, created from the superposition of all registered superpixels.

Finally, the notion of regularity is defined by the GR (Global Regularity) metric combining these two metrics such that:

$$\mathrm{GR}(\mathcal{S}) = \frac{1}{\sum\limits_{S_i\in\mathcal{S}} |S_i|} \sum_{S_i\in\mathcal{S}} |S_i| \mathrm{SRC}(S_i)\mathrm{SMF}(S_i). \tag{11}$$

2) Generalization in the acquisition space: Ideally the regularity should be evaluated in the acquisition space. In our context, S_i in the spherical acquisition space gives S_i^a, a set of 3D positions on the unit sphere (3). GR being based on the computation of convex hull, and barycenter registration, it cannot be directly applied to such point clouds in $\mathbb{R}^3$.

To generalize the metric, we propose to simply project the 3D points of S_i^a on a discrete 2D plan, and then apply the initial GR. The whole process is illustrated in Figure 4. To do so, we first project a superpixel S_i in the discrete image space to its acquisition one, here to get a spherical point cloud S_i^a. Then, we apply a Principal Component Analysis (PCA) on S_i^a, and project the points on its two most significant eigenvectors to reduce to a 2D point cloud $S_i^{a\to\{\mathbb{R}^2\}}$. Finally, a downsampling is performed to obtain a discrete 2D shape $S_i^{a\to\{\mathbb{N}^2\}}$. By this way, each superpixel shape has a relevant discrete projection in the acquisition space. The proposed Generalized Global Regularity (G-GR) metric is defined as:

$$\text{G-GR}(\mathcal{S}) = \frac{\sum\limits_{S_i\in\mathcal{S}} \left|S_i^{a\to\{\mathbb{N}^2\}}\right| \mathrm{SRC}(S_i^{a\to\{\mathbb{N}^2\}})\mathrm{SMF}(S_i^{a\to\{\mathbb{N}^2\}})}{\sum\limits_{S_i\in\mathcal{S}} \left|S_i^{a\to\{\mathbb{N}^2\}}\right|}. \tag{12}$$

With the proposed G-GR metric, a gap is now visible such that no planar methods have higher regularity than spherical ones for a given number of superpixels (Section IV-C).

IV. Results

A. Validation Framework

1) Dataset: We consider the Panorama Segmentation Dataset (PSD) [25], containing 75 360° equirectangular images of 512×1024 pixels, having between 115 and 1085 segmented objects with an average size of 1334 pixels. These images are taken from the standard spherical dataset SUN360 [32], and accurate ground-truth segmentations are provided by [25].

2) Metrics: To relevantly evaluate SphSPS performances and compare to state-of-the-art methods, we use the superpixel metrics recommended in [24], for several superpixel numbers. The main aspects to evaluate are the object segmentation and spatial regularity performances, which is robustly evaluated in the acquisition space with the proposed G-GR metric (12).

For the segmentation aspect, the standard measure is the Achievable Segmentation Accuracy (ASA) [16], highly correlated to the Undersegmentation Error [33] as shown in [24]. The ASA measures the overlap of a superpixel segmentation $\mathcal{S}$ with the ground truth objects, denoted $\mathcal{G}$, such as:

$$\mathrm{ASA}(\mathcal{S},\mathcal{G}) = \frac{1}{\sum\limits_{S_i\in\mathcal{S}} |S_i|} \sum_{S_i} \max_{G_j\in\mathcal{G}} |S_i \cap G_j|. \tag{13}$$

The Boundary-Recall (BR) is the commonly employed metric to evaluate the detection of the ground truth contours $\mathcal{B}(\mathcal{G})$ by the boundaries of the superpixels $\mathcal{B}(\mathcal{S})$ such that:

$$\mathrm{BR}(\mathcal{S},\mathcal{G}) = \frac{1}{|\mathcal{B}(\mathcal{G})|} \sum_{p\in\mathcal{B}(\mathcal{G})} \delta[\min_{q\in\mathcal{B}(\mathcal{S})} \|p-q\| < \epsilon], \tag{14}$$

with ϵ a distance threshold set to 2 pixels [24], and $\delta[a] = 1$ when a is true and 0 otherwise. To prevent methods generating superpixels with fuzzy borders to get high performances [24], BR results are compared to the Contour Density (CD), *i.e.*, the number of pixels of superpixel borders.

The standard Precision-Recall curves can also be represented to illustrate the overall object contour detection performances. These are computed on a contour probability map $\in [0,1]$ generated by averaging the superpixel borders obtained at different scales $K \in [50, 3000]$. This map is thresholded by several intensities to get a binary contour map. For each threshold, the Precision (PR), the percentage of accurate detection among the superpixel borders, is computed with the BR measure. For all PR curves, to synthesize the contour detection performance, we also report the maximum on all thresholds of the F-measure defined as:

$$\mathrm{F} = \frac{2.\mathrm{PR}.\mathrm{BR}}{\mathrm{PR}+\mathrm{BR}}. \tag{15}$$

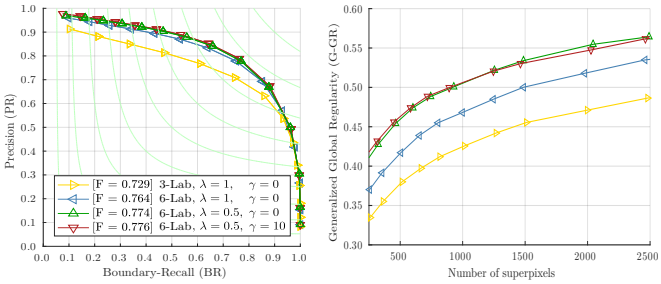
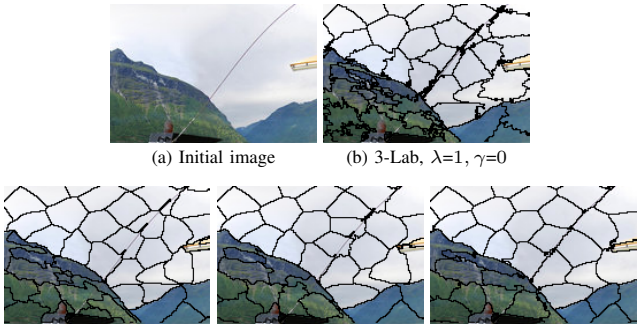


Fig. 5. Impact of the SphSPS distance parameters. The contributions enable to significantly improve the accuracy and regularity performances.



(a) Initial image (b) 3-Lab, λ=1, γ=0

6-Lab, λ=1, γ=0 6-Lab, λ=**0.5**, γ=0 6-Lab, λ=0.5, γ=**10**

Fig. 6. Visual impact of SphSPS parameters. Each contribution relevantly increases the regularity and $\gamma = 10$ integrates the contour prior information.

3) Parameter settings: SphSPS was implemented with MATLAB using C-MEX code, on a standard Linux computer with 12 cores at 2.6 GHz with 64GB of RAM. Contrary to [12], using the 3 average color features of the CIELab space, we use the 6 CIELab dimension space of [19], also including the features of neighboring pixels [21]. In the shortest path, $N = 15$ pixels are considered (9). The number of iterations is set to 5, and the parameter λ (4), setting the trade-off between the central pixel and the ones on the shortest path, is set to 0.5 as in [21]. When used, the contour prior is computed from [34] and γ set to 10 (5). Finally, the parameter m (6) is empirically set to 0.15 to provide a visually satisfying trade-off between the respect of object contours and spatial regularity.

B. Impact of Contributions

In this section, we show the impact of contributions within SphSPS. We report for different distance settings the contour detection PR/BR curves, with the maximum F-measure (15), and the regularity G-GR (12) curves in Figure 5, and a zoom on a segmentation example in Figure 6. With a 3 feature dimension space, SphSPS reduces to the spherical SLIC algorithm [12]. With the 6 dimension space, SphSPS uses the CIELab features of [19], and the neighboring pixels information as in [21], with $\lambda = 0.5$ the color distance (4), and $\gamma = 10$, the contour information on the shortest path (5).

We demonstrate that each contribution enables to improve the segmentation performances. We can especially observe that the color distance on the shortest path, that strengthens the superpixel convexity and homogeneity, indeed provides much more regular superpixels while also increasing the accuracy.

C. Comparison with the State-of-the-Art Methods

We compare the performances of the proposed SphSPS approach to the ones of the state-of-the-art methods. We consider the planar methods SLIC [1], LSC [19], SNIC [20] and SCALP [21], and the spherical approach SphSLIC [12] in 2 different settings, *i.e.*, considering the Euclidean (SphSLIC-Euc) and the Cosine (SphSLIC-Cos) distances (see Section II-B). To ensure fair comparison, planar and SphSLIC-Euc methods are used with their default settings, since they provide a good trade-off between accuracy and regularity. Note that for the SphSLIC-Cos method [12], results are reported for the regularity setting optimizing the segmentation accuracy, since low performances were obtained with default settings.

In Figure 7, we report the contour detection results measured by PR/BR curves with F-measure (15), and BR/CD (14), the segmentation of objects with ASA (13), and regularity with the proposed G-GR metric (12), obtained for several numbers of superpixels. SphSPS overall obtains the best segmentation results, with for instance the higher F-measure (0.776), and significantly outperforms the other spherical method SphSLIC, in both distance modes, while producing very regular superpixels. Note that even without contour prior ($\gamma = 0$), *i.e.*, only using color information on the shortest path, SphSPS still significantly outperforms the other state-of-the-art methods. We also observe that by using a linear path approach, SCALP [21] degrades the segmentation accuracy of LSC [19]. This result highlights the need for considering our spherical shortest path instead of the linear one.

The regularity measured with the proposed G-GR (12) appears to be very relevant and able to differentiate planar and spherical methods. It evaluates the convexity and contour smoothness of each superpixel along with their consistency, while C (10) is only based on a non robust and independent circularity assumption. Hence, with G-GR, the regularity in the spherical space is accurately measured such that no planar methods have higher regularity than spherical ones for a given number of superpixels, contrary to C [12].

In Figure 8, we show segmentation examples of SphSPS compared to the state-of-the-art methods, on 360° equirectangular images and projected on the unit sphere. SphSPS produces regular superpixels in the spherical space and accurately captures the object contours compared to the other methods.

Finally, in terms of processing time, the relevance of our features enables SphSPS to fastly converge in a low number of iterations. For instance, only using the 6 dimensional feature space [19], SphSPS generates superpixels in 0.85s per image of size 512×1024 pixels and already obtains higher accuracy ($F = 0.764$) than the state-of-the-art methods (see Figure 5). With the significant optimizations proposed in Section II-C3, SphSPS can use the information on the shortest path to obtain significantly higher accuracy in only 2.30s, *i.e.*, faster than existing spherical approaches [12]. Moreover, with basic multi-threading, we easily reduce the processing time of our implementation to 0.7s to further facilitate the use of SphSPS[1].

[1]Available code at: https://github.com/rgiraud/sphsps

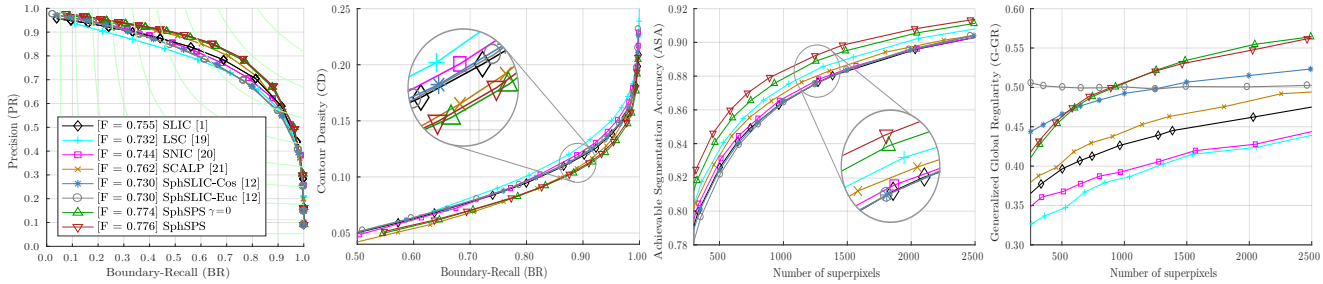


Fig. 7. Quantitative comparison on PR/BR, BR/CD, ASA and G-GR of the proposed SphSPS method to the state-of-the-art ones on the PSD [12].

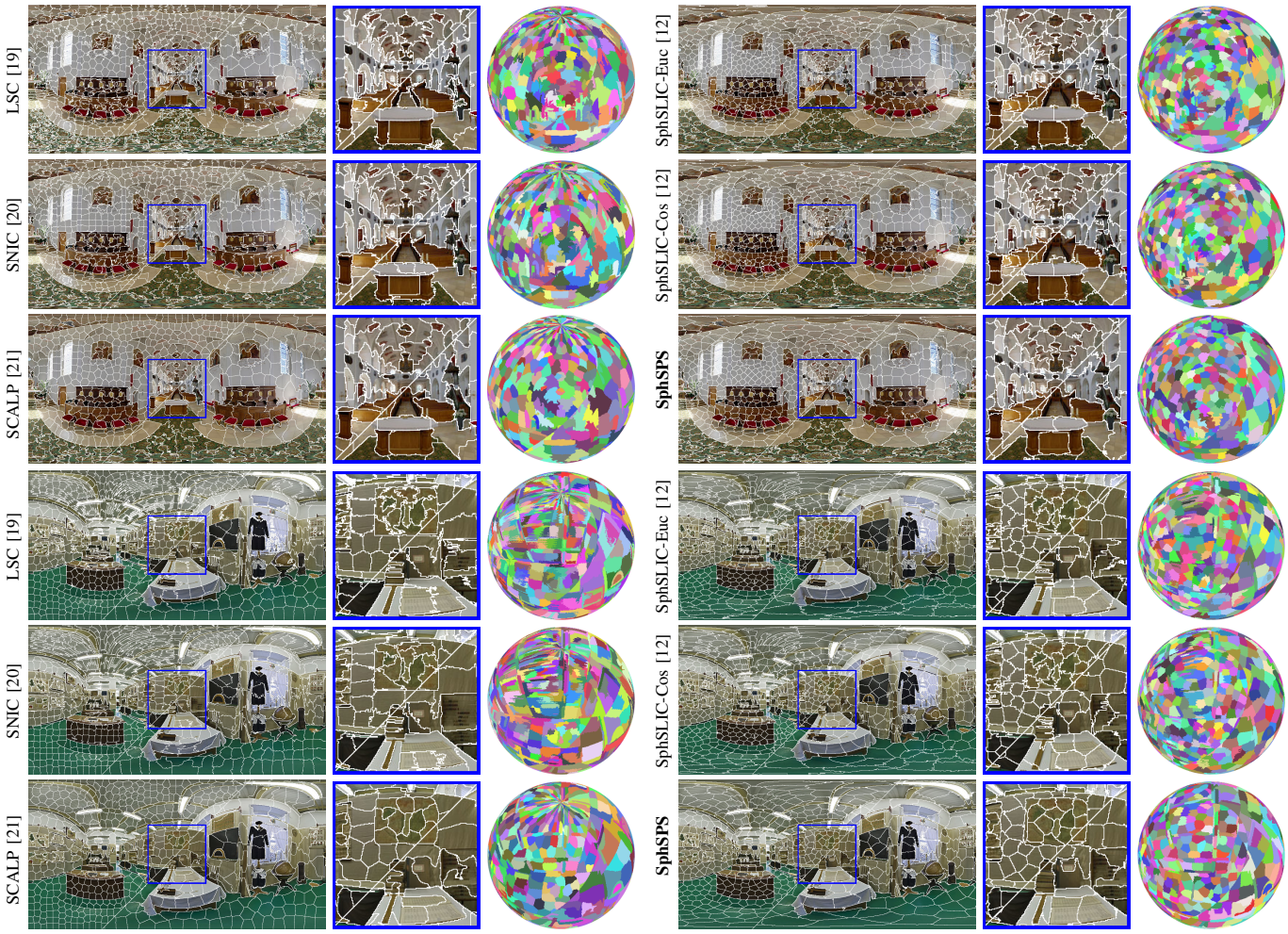


Fig. 8. Visual comparison between SphSPS and the best planar (left) and spherical (right) state-of-the-art methods on PSD images, for two superpixel numbers $K = 1200$ (top-left) and $K = 400$ (bottom right). The compared methods may generate inaccurate superpixels, while SphSPS produces regular spherical superpixels with smooth boundaries that adhere well to the image contours.

V. Conclusion

In this work, we generalize the shortest path approach between a pixel and a superpixel barycenter [24] to the case of spherical images. We show that the complexity resulting of the large number of pixels to process can be extremely reduced using the path redundancy on the 3D sphere. Color features on this path enable to generate both very accurate and regular superpixels. Moreover, SphSPS can consider a contour prior information to further improve its performances.

To ensure a relevant evaluation of regularity, we introduce a generalized metric measuring the spatial convexity and consistency in the 3D spherical space. While providing regular results in the acquisition space, SphSPS significantly outperforms both planar and spherical state-of-the-art methods.

Accuracy and regularity in the acquisition space are crucial for relevant display and for computer vision pre-processing. Future works will extend our method to spherical videos and other acquisition spaces, *e.g.*, circular or polarimetric.

REFERENCES

[1] R. Achanta, A. Shaji, and K. Smith et al., "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, pp. 2274–2282, 2012.

[2] J. Tighe and S. Lazebnik, "SuperParsing: scalable nonparametric image parsing with superpixels," in *European Conference on Computer Vision*, 2010, pp. 352–365.

[3] H. Wang and P. A. Yushkevich, "Multi-atlas segmentation without registration: a supervoxel-based approach," *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 535–542, 2013.

[4] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3061–3070.

[5] J. Liu, W. Yang, X. Sun, and W. Zeng, "Photo stylistic brush: robust style transfer via superpixel-based bipartite graph," *IEEE Trans. on Multimedia*, vol. 20, no. 7, pp. 1724–1737, 2017.

[6] S. Gould, J. Zhao, X. He, and Y. Zhang, "Superpixel graph label transfer with learned distance metric," in *European Conference on Computer Vision*, 2014, pp. 632–647.

[7] R. Giraud, V.-T. Ta, A. Bugeau, P. Coupé, and N. Papadakis, "SuperPatchMatch: an algorithm for robust correspondences using superpixel patches," *IEEE Transactions on Image Processing*, vol. 26, no. 8, pp. 4068–4078, 2017.

[8] W.-C. Tu, M.-Y. Liu, V. Jampani, D. Sun, S.-Y. Chien, M.-H. Yang, and K. Jan, "Learning superpixels with segmentation-aware affinity loss," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[9] D. Zorin and A. H. Barr, "Correction of geometric perceptual distortions in pictures," in *International Conf. on Computer Graphics and Interactive Techniques*, 1995, pp. 257–264.

[10] R. Cabral and Y. Furukawa, "Piecewise planar and compact floorplan reconstruction from images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 628–635.

[11] K. Sakurada and T. Okatani, "Change detection from a street image pair using cnn features and superpixel segmentation." in *British Machine Vision Conference*, 2015, pp. 61–1.

[12] Q. Zhao, F. Dai, Y. Ma, L. Wan, J. Zhang, and Y. Zhang, "Spherical superpixel segmentation," *IEEE Trans. on Multimedia*, vol. 20, no. 6, pp. 1406–1417, 2018.

[13] V. Machairas, M. Faessel, D. Cárdenas-Peña, T. Chabardes, T. Walter, and E. Decencière, "Waterpixels," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3707–3716, 2015.

[14] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi, "Turbopixels: fast superpixels using geometric flows," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2290–2297, 2009.

[15] P. Buyssens, I. Gardin, S. Ruan, and A. Elmoataz, "Eikonal-based region growing for efficient clustering," *Image and Vision Computing (IVC)*, vol. 32, no. 12, pp. 1045–1054, 2014.

[16] M.-Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa, "Entropy rate superpixel segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 2097–2104.

[17] J. Yao, M. Boben, S. Fidler, and R. Urtasun, "Real-time coarse-to-fine topologically preserving segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2947–2955.

[18] Y. Zhang, X. Li, X. Gao, and C. Zhang, "A simple algorithm of superpixel segmentation with boundary constraint," *IEEE Transactions on Circuits and Systems for Video Technology*, no. 99, 2016.

[19] J. Chen, Z. Li, and B. Huang, "Linear spectral clustering superpixel," *IEEE Transactions on Image Processing*, vol. 26, pp. 3317–3330, 2017.

[20] R. Achanta and S. Süsstrunk, "Superpixels and polygons using simple non-iterative clustering," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4895–4904.

[21] R. Giraud, V.-T. Ta, and N. Papadakis, "Robust superpixels using color and contour features along linear path," *Computer Vision and Image Understanding*, vol. 170, pp. 1–13, 2018.

[22] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167–181, 2004.

[23] H. Yang and H. Zhang, "Efficient 3d room shape recovery from a single panorama," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5422–5430.

[24] R. Giraud, V.-T. Ta, and N. Papadakis, "Evaluation framework of superpixel methods with a global regularity measure," *Journal of Electronic Imaging*, vol. 26, no. 6, 2017.

[25] L. Wan, X. Xu, Q. Zhao, and W. Feng, "Spherical superpixels: benchmark and evaluation," in *Asian Conference on Computer Vision*, 2018, pp. 703–717.

[26] T.-T. Wong, W.-S. Luk, and P.-A. Heng, "Sampling with hammersley and halton points," *Journal of Graphics Tools*, vol. 2, no. 2, pp. 9–24, 1997.

[27] J. E. Bresenham, "Algorithm for computer control of a digital plotter," *IBM Systems Journal*, vol. 4, no. 1, pp. 25–30, 1965.

[28] M. Gromov *et al.*, "Filling riemannian manifolds," *Journal of Differential Geometry*, vol. 18, no. 1, pp. 1–147, 1983.

[29] C. F. Karney, "Algorithms for geodesics," *Journal of Geodesy*, vol. 87, no. 1, pp. 43–55, 2013.

[30] A. Schick, M. Fischer, and R. Stiefelhagen, "Measuring and evaluating the compactness of superpixels," in *International Conference on Pattern Recognition*, 2012, pp. 930–934.

[31] R. Osserman *et al.*, "The isoperimetric inequality," *Bulletin of the American Mathematical Society*, vol. 84, no. 6, pp. 1182–1238, 1978.

[32] J. Xiao, K. A. Ehinger, A. Oliva, and A. Torralba, "Recognizing scene viewpoint using panoramic place representation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2695–2702.

[33] P. Neubert and P. Protzel, "Superpixel benchmark and comparison," in *Forum Bildverarbeitung*, 2012, pp. 1–12.

[34] S. Xie and Z. Tu, "Holistically-nested edge detection," in *IEEE International Conference on Computer Vision*, 2015, pp. 1395–1403.